IJACSA
COMPUTER SCIENCE JOURNAL
International Journal of Advanced Computer Science and Applications

Volume 17 Issue 2
February 2026



ISSN 2156 - 5570 (Online)
ISSN 2158 - 107X (Print)

THE SCIENCE AND INFORMATION ORGANIZATION

www.ijacsa.thesai.org

Table of Contents

Paper 1: A Comparison of Metaheuristic Methods for the Vehicle Routing Problem

Authors: Manal El Jaouhari, Ghita Bencheikh, Ghizlane Bencheikh

PAGES 1 – 7

Paper 2: A Confidence-Aware Multi-Layer Framework for Drone Forensic Inference Using Heterogeneous Digital Evidence Sources and Flight Logs Analysis

Authors: Nidhiba Parmar, Naveen Kumar Chaudhary

PAGES 8 – 18

Paper 3: Real-Time Data-Driven Decision Support in Retail: A Hybrid GraphSAGE+XGBoost Model for Predicting Reorder Behavior and Unraveling Consumer Communities

Authors: Balayet Hossain, Md Deluar Hossen, Md Nuruzzaman Pranto, Belal Hossain, Sabrina Shamim Moushi, Nusrat Ameri, Khandakar Rabbi Ahmed

PAGES 19 – 25

Paper 4: Evaluating the Efficiency of LLM-Generated Software in Resisting Malicious Attacks

Authors: Dominic Niceforo, Haydar Cukurtepe

PAGES 26 – 34

Paper 5: Beyond the Interface: AI Integration Through Input Stream Mediation and Intelligent Output Simulation

Authors: Divij H. Patel

PAGES 35 – 47

Paper 6: Deep Learning Models: A General Methodology

Authors: Alberto Nogales, Ana M. Maitín, Álvaro J. García-Tejedor

PAGES 48 – 55

Paper 7: A Constraint-Driven Conversational Architecture for Staged Task Interaction Using Large Language Models

Authors: Joseph Benjamin Ilagan, Jose Ramon Ilagan

PAGES 56 – 63

Paper 8: Measurement of Dataset Quality and Computational Quality of Graph Neural Networks on Analog Integrated Circuit Recognition System

Authors: Arif Abdul Mannan, Koichi Tanno

PAGES 64 – 73

Paper 9: LayCoder: UI Layout Completion with an Encoder-Only Transformer and Layout Tokenizer

Authors: Iskandar Salama, Luiz Henrique Mormille, Masayasu Atsumi

PAGES 74 – 87

Paper 10: Reducing Interference in Human–Robot Collaboration: A Distance-Based Policy for Disassembly Collaborative Task

Authors: Wendy Cahya Kurniawan, Wen Liang Yeoh, Osamu Fukuda

PAGES 88 – 94

Paper 11: Understanding Authentication and Authorization: A Comparative Analysis of Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Relationship-Based Access Control (ReBAC) Authorization Models

Authors: Madhuri Margam

PAGES 95 – 99

Paper 12: Construction of an International Trade Financial Risk Assessment and Prediction Model Based on Big Data Analysis

Authors: Zeyu Liu

PAGES 100 – 112

Paper 13: Attention and Representation Learning in Byte-Level Digital Forensics: A Survey of Methods, Challenges, and Applications

Authors: Teena Mary, Sreeja CS

PAGES 113 – 124

Paper 14: Estimation of Landslide Hazard Zones Using Deep Learning Based on Diverse Geospatial Data

Authors: Kohei Arai, Kengo Oiwane, Hiroshi Okumura

PAGES 125 – 134

Paper 15: A Hybrid Framework Integrating GNN-LSTM-CNN to Map the Impact of MSME User Behavior on Digital Transformation

Authors: Jani Kusanti, Erni Widiastuti, Bintara Sura Priambada, Ramadhian Agus Triono Sudalyo, Masdava Aviv Masyayissa, Rizki Adhi Pratama

PAGES 135 – 140

Paper 16: A Computer Vision–Based Method for Determining the Vaccine Injection Position in Pangasius Fingerlings

Authors: Nguyen Phuc Truong, Luong Vinh Quoc Danh, Nguyen Chanh Nghiem

PAGES 141 – 150

Paper 17: Exposure-Based Media Mix Modeling Using Machine Learning and Genetic Algorithms

Authors: Thejan Dulara, Indra Mahakalanda, Prasanga Jayathunga

PAGES 151 – 160

Paper 18: An Ensemble Learning Framework with Metaheuristic Optimization for Credit Card Fraud Detection

Authors: Agung Nugroho, Muhtajuddin Danny, Ismasari Nawangsih

PAGES 161 – 167

Paper 19: A Lossless Medical Image Compression Framework Using Logic Minimization

Authors: Swathi Pai M, Jacob Augustine, Pamela Vinitha Eric

PAGES 168 – 178

Paper 20: An Empirical Evaluation of Multivariate Temporal Convolutional Networks with Global Market Indicators for Forecasting the Indonesian LQ45 Index

Authors: Yohanes Marakub, Muhammad Zarlis

PAGES 179 – 190

Paper 21: Integrating ABM and GIS for Flood Evacuation Planning: A Systematic Review and Future Direction

Authors: Kabir Musa Ibrahim, Abubakar Ahmad, Noor Akma Abu Bakar,
Mazlina Abdul Majid, Azamuddin Rahman

PAGES 191 – 203

Paper 22: A Novel Lightweight Explainable Multilayer Adaptive RNN-Based Intrusion Detection Framework

Authors: Nidhi Srivastav, Rajiv Singh

PAGES 204 – 214

Paper 23: Empirical Validation of Learnability Factors in Web-Based AR: Insights from the LEMARK–Hafsa Model Grounded in Kolb's Experiential Learning Theory

Authors: Sayera Hafsa, Mazlina Abdul Majid, Shafiq Ur Rehman

PAGES 215 – 224

Paper 24: Automating Computation Independent Model Elicitation in MDA using Task-Oriented Dialogue with In-Context Learning

Authors: Mohamed EL Ayadi, Yassine Rhazali, Mohammed Lahmer

PAGES 225 – 231

Paper 25: Mapping Research on Artificial Intelligence in Customer Experience: A Bibliometric Analysis

Authors: Firdaws Hayoun, Brahim Ouabouch, Youssef Aatif, Taoufiq Yahyaoui, Fatima Zahra El Arbaoui

PAGES 232 – 239

Paper 26: AI-Driven Robotic Waste Sorting for Techno-Economic Assessment in Urban Indonesia

Authors: Ida Nurhaida, Mohammad Nasucha, Hari Nugraha

PAGES 240 – 249

Paper 27: A Unified Benchmarking Framework for Offline Handwritten Signature Verification Using Deep Learning Architectures

Authors: Eissa Alreshidi

PAGES 250 – 264

Paper 28: Deep Learning Based Detection of Prostate Cancer in MRI Using Biopsy-Confirmed Ground Truth

Authors: Samana Jafri, Gajanan Birajdar

PAGES 265 – 272

Paper 29: A Composite Approach Extracting Software Quality Attributes Assessing Operational Profiles

Authors: Sudhakar Kambhampati, G Vamsi Krishna

PAGES 273 – 283

Paper 30: Robust Medical Image Reconstruction Using a Self-Evolving Encoder–Decoder and Adaptive Convolutional Power Scaling

Authors: Dhanusha P B, J. Bennilo Fernandes, A. Muthukumar, A. Lakshmi

PAGES 284 – 292

Paper 31: Cybersecurity Awareness Among Undergraduate Students in Saudi Arabia: A Quantitative Study

Authors: Turky A. Saderaldin, Ali Abuabid

PAGES 293 – 298

Paper 32: An Analytical Study of Data Augmentation Across Audio Representations for Infant Cry Classification

Authors: Meriyem Ghanjaoui, Abdelaziz Daaif, Abdelmajid Bousselham, Sajid Rahim,
Ahmed Bouatmane, Mohamed Elyoussfi

PAGES 299 – 304

Paper 33: Hybrid BERT–BiLSTM Architecture for Enhanced Cyber Threat Intelligence Classification

Authors: Syarif Hidayatulloh, Salman Topiq, Ifani Hariyanti, Dwi Sandini

PAGES 305 – 315

Paper 34: An Offline Estimation Method for Hip and Knee Joint Angles of Lower Limbs Based on Quaternion and DTW Time Alignment

Authors: Yuling Zhang, Yuejian Hua, Shengli Luo, Xiaolong Shu, Hongyan Tang, Hongliu Yu

PAGES 316 – 323

Paper 35: Radiomics Feature Profiling of Brodmann Regions in Structural MRI: A Machine Learning Study of Intensive Verbal Memorisation (Huffaz vs Controls)

Authors: Mohd Zulfaezal Che Azemin, Iqbal Jamaludin, Abdul Halim Sapuan, Mohd Izzuddin Mohd Tamrin

PAGES 324 – 330

Paper 36: Cognitive Assistance for Prosopagnosia Patients Using Landmark-Based Identity and Emotion Recognition

Authors: Bhavana Nagaraj, Rajanna Muniswamy

PAGES 331 – 340

Paper 37: Comparative Analysis of Neural Network Architectures for Classifying Depressive Content in Social Networks

Authors: Yntymak Abdrazakh, Rita Ismailova, Nurseit Zhunissov, Arypzhan Aben,
Anuarbek Amanov, Aigerim Baimakhanova

PAGES 341 – 351

Paper 38: Attention-Based Capsule Network with Vision Transformer for Underwater Hyperspectral Image Classification

Authors: Thiyagarajan B, Thenmozhi M

PAGES 352 – 365

Paper 39: Effectiveness of Capsule Networks in Detecting Deepfakes Instead of Traditional CNNs

Authors: M. C. Weerawardana, T. G. I. Fernando

PAGES 366 – 374

Paper 40: Machine-Learning–Assisted Probabilistic Wind Assessment at Sechura, Peru

Authors: Ubaldo Yancachajlla Tito, Celso Antonio Sanga Quiroz, Edilberto Velarde Coaquira, Germán Belizario Quispe

PAGES 375 – 383

Paper 41: Ziraai SSI: A Blockchain-Based Self-Sovereign Identity Model for Agricultural Supply Chains

Authors: Tariq Alar, Gnana Bharathy, Usha Batra, Mukesh Prasad

PAGES 384 – 393

Paper 42: Rubric-Relational Discourse Modeling with Counterfactual Explainability for Multi-Trait Automated Essay Scoring

Authors: N. Sreedevi, M. Madhusudhan Rao, Sridevi Dasam, Roopa Traisa, Jasgurpreet Singh Chohan, V. Saranya, Ahmed I. Taloba

PAGES 394 – 407

Paper 43: Face Sketch Recognition: Ethnic Groups Classification and Recognition Via a VGG16 Model Approach

Authors: Khalid OUNACHAD, Mohamed EL GHMARY

PAGES 408 – 419

Paper 44: Integrating Japanese Festival Traditions into Computational Thinking: A Culturally Responsive Approach to Micro:Bit-based Physical Computing

Authors: Daiki Sugiyama, M. Fahim Ferdous Khan, Ken Sakamura

PAGES 420 – 427

Paper 45: Machine Learning-Based Effort Prediction and Early Risk Detection in Software Development Projects: A Case Study

Authors: Andreea-Elena Catana, Adriana Florescu

PAGES 428 – 439

Paper 46: Self-Supervised and Explainable Transformer-Based Architectures for Robust End-to-End Speech and Language Understanding

Authors: Mahfuzul Huda

PAGES 440 – 446

Paper 47: Secure Ultra-Low Latency Data Paths: A Hybrid Architecture for High Speed Networking under Adversarial Conditions

Authors: Abdulbasid Banga

PAGES 447 – 456

Paper 48: Hybrid Deep Learning for Academic Achievement Prediction Using Spatio-Temporal and Behavioral Data in Higher Education

Authors: Asim Seedahmed Ali Osman

PAGES 457 – 470

Paper 49: Farm and Learn: An Offline Mobile Learning System Integrating AR, AI, and Game-Based Learning for Agricultural Education Among Children

Authors: Pubuditha De Silva, Chamodya Prabodhani, Daminda Herath

PAGES 471 – 480

Paper 50: An Enhanced Approach for Workmen's Compensation Insurance Fraud Detection Based on Fuzzy Rule-Based System

Authors: Reham M. Essa

PAGES 481 – 493

Paper 51: Design of a Vision Transformer-Based Architecture for Automatic Facial Emotion Monitoring in Workplace Environments

Authors: Renzo Sebastian Gonzalez Caceres, Jeramel Melissa Avila Saldaña, Patricia Gissela Pereyra Salvador

PAGES 494 – 501

Paper 52: Integrating Acoustic and Image Data Features for Melon Ripeness Classification Using Convolutional Neural Network

Authors: Endang Purnama Giri, Agus Buono, Karlisa Priandana, Dwi Guntoro

PAGES 502 – 511

Paper 53: Comparative Analysis of Machine Learning Based Algorithms for Predicting Injury Severity in Road Accidents

Authors: Soumaya AMRI, Mohammed AL ACHHAB, Mohamed LAZAAR

PAGES 512 – 523

Paper 54: Machine Learning and Deep Learning for Detecting Fake News in a Low-Resource Language

Authors: Elton Tata, Jaumin Ajdarim, Nuhi Besimi

PAGES 524 – 531

Paper 55: AMCS: Adaptive Multi-Controller SDN Security with Stateful Traffic Intelligence for Fast and Accurate Multi-Vector Attack Detection

Authors: Ameer El-Sayed, Mohamed Nosseir Hemdan, Noha Abdelkarim, Ehab Rushdy, Hanaa M. Hamza

PAGES 532 – 546

Paper 56: Wavelet-Based Dual-Domain Phase Alignment for Predicting Stock Index Trends from Investor Sentiment Cycles

Authors: Monika Gorkhe, Diksha Tripathi, Pravin D Sawant, Elangovan Muniyandy, Veera Ankalu Vuyyuru, Pratik Gite, Raman Kumar

PAGES 547 – 557

Paper 57: Machine Learning-Based Autism Spectrum Disorder Classification Using an Enhanced Convolutional Neural Network Algorithm

Authors: P. Yugander, M. Jagannath

PAGES 558 – 565

Paper 58: Privacy-Preserving Federated Learning for Multi-Institutional Lung Cancer Severity Detection

Authors: Ch. Srividya, K. Ramasubramanian, Myneni.Madhu Bala

PAGES 566 – 576

Paper 59: RiskMIS: A Web-Based Risk Management Information System

Authors: Alya Thiab Almutairi, Kassem Saleh

PAGES 577 – 592

Paper 60: Uncertainty-Aware Volumetric Transformer with Dual Spatial-Channel Attention for Lung Nodule Classification

Authors: B. N.Patil, TK Rama Krishna Rao, Nurilla Mahamatov, Elangovan Muniyandy, Arun Prasad.VK, Chamandeep Kaur, Aaquil Bunglowala, Ahmed I. Taloba

PAGES 593 – 604

Paper 61: An Articulatory-Aware CNN-BiGRU-Attention Framework for Explainable Phoneme-Level Pronunciation Assessment in ESL Speech

Authors: P. Bindhu, Jasgurpreet Singh Chohan, M. Durairaj, Megha Sawangikar, N. Neelima, Elangovan Muniyandy, G. Sanjiv Ra, Loay F. Hussien

PAGES 605 – 617

Paper 62: Cloud-Continuum-Based Deep Learning Optimization Framework for Next-Generation Healthcare Data Performance on IoT Platform

Authors: G. Aravindh, K. P. Sridhar

PAGES 618 – 629

Paper 63: Adaptive Network Security Framework for Distributed Quantum-Assisted Cloud Continuum Architectures

Authors: P. Suseendhar, K. P. Sridhar

PAGES 630 – 641

Paper 64: Deep Hybrid Learning for Sustainable Industrial Forecasting: Integrating CNN–LSTM Models to Enhance Economic Efficiency and Carbon Performance

Authors: Mohamed Amine Frikha, Mariem Mrad, Younes Boujelben, Soufiene Ben Othman

PAGES 642 – 648

Paper 65: A System-Oriented Machine Learning Approach for Planning and Execution of Decisions in Software Project Management

Authors: Foziah Gazzawe

PAGES 649 – 659

Paper 66: Cluster Domain-Aware Client Selection for Federated Learning in The Healthcare Field (CDCSF)

Authors: Sanaa Lakrouni, Marouane Sebgui, Slimane Bah

PAGES 660 – 673

Paper 67: Temporal Attention Networks for Real-Time Multimodal Emotion Recognition from EEG and fNIRS Signals

Authors: Vinod Waiker, Anne Marie D. Pahiwon, Ankush Mehta, Gadug Sudhamsu, Pavithra M, K. Kiran Kumar, B Kiran Bala, Osama R. Shahin

PAGES 674 – 685

Paper 68: Explainable AI with DRL for Smart Home Energy Management and Residential Cost Savings in IoT-Based Autonomic Systems

Authors: Prabagaran A, Srinivasan Sriramulu, S. Premkumar, Sheelesh Kumar Sharma, Karthi Govindharaju, Angelin Blessy J

PAGES 686 – 699

Paper 69: Securing Blue Carbon Accounting: A Cryptographic Framework for Coastal Ecosystem Monitoring

Authors: Heider A. M. Wahsheh

PAGES 700 – 707

Paper 70: Privacy-Aware Customer Segmentation Using a Distributed Graph-Based Attribute Projection Framework

Authors: Pentareddy Ashalatha, G. Krishna Mohan

PAGES 708 – 720

Paper 71: IoT-Driven Sensor Selection for Smart Water and Electricity Systems: A TOPSIS and VIKOR Approach

Authors: Oumaima Rhallab, Rachid Dehbi, Zouhair Ibn Batouta, Amine Dehbi

PAGES 721 – 728

Paper 72: A Feasibility Study on Synthetic RGB-NIR Image Generation for Oil Palm Fresh Fruit Bunch Grading

Authors: Nor Surayahani Suriani, Norzali Hj Mohd, Shaharil Mohd Shah, Siti Zarina Muji, Fadilla Atyka Nor Rashid

PAGES 729 – 738

Paper 73: The Acquiring Optimal Models of Random Forest and Support Vector Machine Through Tuning Hyperparameters in Classifying the Imbalanced Data

Authors: Dwija Wisnu Brata, Arif Djunaidy, Daniel Oranova Siahaan, Samingun Handoyo

PAGES 739 – 749

Paper 74: K-Nearest Neighbors Algorithm for Short-to-Medium Term Directional Stock Price Forecasting: An Analysis of Thailand's Banking Sector

Authors: Passawan Noppakaew, Parit Wanitchatchawan, Kanchana Phuhoy, Natthasorn Seubwong

PAGES 750 – 754

Paper 75: Multi-Criteria Methodology for Selecting Communication Protocols in M2M Environments

Authors: Oleg Iliev

PAGES 755 – 761

Paper 76: Generative AI as a Catalyst for Interoperability and Data-Driven Decision Support in Healthcare Systems of Developing Countries

Authors: YNSUFU Ali, MOSKOLAI NGOSSAHA Justin, AYISSI ETEME Adolphe, BOWONG TSAKOU Samuel

PAGES 762 – 775

Paper 77: Development of Lightweight Residual Convolutional Neural Network for Efficient Facial Emotion Recognition

Authors: Yelnur Mutaliyev, Zhuldyz Kalpeyeva

PAGES 776 – 781

Paper 78: Integrating Deep Reinforcement Learning for Initialization and Adaptive Pheromone Updates in Ant Colony Optimization for UAV Pathing

Authors: Mohamed A. Damos, Wenbo Xu, Abdolraheem Khader, Ali Ahmed, Mohammed Al-Mahbashi, Almuhannad S.Alorfi

PAGES 782 – 791

Paper 79: A Context-Aware Hybrid Recommendation Framework for E-Learning Platforms

Authors: Kaoutar Errakha, Amina Samih, Sanaa Dfouf, Abderrahim Marzouk

PAGES 792 – 802

Paper 80: Efficient CNN-Based Time-Domain Denoising of Impulsive Noise in NB-PLC Systems

Authors: Wided Belhaj Sghaier, Fatma Rouissi, Héla Gassara, Fethi Tlili

PAGES 803 – 812

Paper 81: Leveraging Attention Mechanism and Class Weighting for Legal Event Detection in Chinese Text

Authors: Jinhong Hu, Shaidah Jusoh

PAGES 813 – 819

Paper 82: SIFChain: A Decentralized Framework for Secure Storage Sharing and Dynamic Access Control in Virtual Power Plants

Authors: Xiaochuan Xu, Xiao Xin, Jie Liu, Ruiqi Fang, Dekai Liu, Zhixin Li

PAGES 820 – 826

Paper 83: AI-Driven Refactoring: Semantic Reconstruction of Domain Models Using LLM Reasoning

Authors: Mohamed El BOUKHARI, Nassim KHARMOUM, Soumia ZITI

PAGES 827 – 834

Paper 84: Enhancing Telecom Churn Prediction Using Emotion-Driven and Behavioral Engagement Features

Authors: Huthaifa Aljawazneh

PAGES 835 – 842

Paper 85: Coronary Heart Disease Prediction Using Machine Learning Algorithms

Authors: Inooc Rubio Paucar, Cesar Yactayo-Arias, Laberiano Andrade-Arenas

PAGES 843 – 857

Paper 86: Integrating Processing-In-Memory into HW/SW Co-Design for Automotive Embedded Systems

Authors: Zineb El Kacimi, Safae Dahmani, Oussama Elissati, Mouhcine Chami

PAGES 858 – 865

Paper 87: A Feasibility Study of Explainable Machine Learning on Small-Scale Postoperative Voice Data

Authors: Noura Haddou, Najlae Idrissi, Sofia Ben Jebara

PAGES 866 – 876

Paper 88: A Dual-Chain and Differential Privacy-Based Solution for Medical Data Privacy Protection and Access Control

Authors: Cen Gu, Luping Wang, Hongjie Wu

PAGES 877 – 887

Paper 89: Leveraging Statistical Invariants to Fortify CNN-Based Medical Diagnostics Against Adversarial Perturbations

Authors: Yassine Chahid, Anas Chahid, Ismail Chahid, Aissa Kerkour Elmiad

PAGES 888 – 895

Paper 90: A Transformer-Based Approach for Multimodal Arabic Sentiment Analysis

Authors: Ayoub BEN CHEIKHI, EL Habib NFAOUI

PAGES 896 – 906

Paper 91: HQ-RTVF: High-Quality Real-Time Virtual Try-On Fitting for Diverse Clothing and Body Morphologies

Authors: Ilham KACHBAL, Khadija Arhid, Said El Abdellaoui

PAGES 907 – 915

Paper 92: Density-Guided Adaptive Patch Learning for Robust Crowd Counting

Authors: Abdullah N Alhawsawi

PAGES 916 – 923

Paper 93: Forward Selection for Time Series-Based Qubit Generation via Parameterized Quantum Gates

Authors: Singaraju Srinivasulu, Nagarajan G

PAGES 924 – 935

Paper 94: A Hybrid DMD–TCN Framework for Interpretable Short-Horizon Prediction of 6-DOF Ship Motions

Authors: Enock Tafadzwa Chekure, Kumeshan Reddy, John Fernandes

PAGES 936 – 951

Paper 95: SecureDML:An Intelligent Framework for Preventing Poisoning Attacks in Distributed Machine Learning Systems

Authors: Archa A. T, Kartheeban K

PAGES 952 – 960

Paper 96: Post-Quantum Module Learning with Rounding-Based Public Key Encryption Using Incomplete Number Theoretic Transform

Authors: Anupama Arjun Pandit, Arun Mishra

PAGES 961 – 975

Paper 97: Abnormal State Detection of Industrial Tools Based on the MGC-YOLOv8 Algorithm

Authors: Guan Yang, Xiang Cheng, Miao Wang, Ziyue Huang, Hao Tang, Yujun Chen

PAGES 976 – 986

Paper 98: Tomato Maturity Analysis: A Comparative Study of Detection and Instance Segmentation Using YOLOv8

Authors: Salma Ait Oussous, Rachid El Bouayadi, Driss Zejli, Aouatif Amine

PAGES 987 – 993

Paper 99: Time Series Anomaly Detection Based on Entropy-Sparsified Time-Frequency Fusion and MsRwGWO Meta-Optimization

Authors: Xiaogang Yuan, Jiaxi Chen, Dezhi An, Jianxin Wan

PAGES 994 – 1009

Paper 100: Blockchain-Based Multi-Chain Data Supervision Mechanism for Traditional Chinese Medicine Traceability System

Authors: Rongjun Chen, Yun Sun, Feng Xue, Yongzhi Ma, Xinyu Wu, Xianxian Zeng, Jiawen Li, Jinchang Ren

PAGES 1010 – 1018

Paper 101: Attention-Guided Fusion of EfficientNet-B0 and Swin Transformer for Cervical Cancer Classification

Authors: Twisibile Mwalughali, Emmanuel C. OGU, Evason Karanja

PAGES 1019 – 1034

Paper 102: Optimizing Resource Allocation for Crisis-Resilient Healthcare Robotics: An Integrated MLR, MDP, and Petri Net Approach

Authors: Ikram Dahamou, Ayoub Elbazzazi, Cherki Daoui

PAGES 1035 – 1044

Paper 103: Enhanced Ant Colony Optimization for Capacitated Vehicle Routing Problem with Time Windows in Franchise Distribution

Authors: Dian Rachmawati, Tommy Lohil, Jos Timanta Tarigan

PAGES 1045 – 1058

Paper 104: Designing an Attack-Vector-Based Taxonomy for IoT Malware

Authors: Huda Aldawghan, Mounir Frikha

PAGES 1059 – 1073

Paper 105: Enhancing Malware Detection Using Machine Learning Models on Static Features

Authors: Ashwag Alotaibi, Mounir Frikha

PAGES 1074 – 1083

Paper 106: Stability-Aware QUBO Feature Selection for Tabular Classification Under Repeated Nested Cross-Validation

Authors: Marco Fidel Mayta Quispe, Leonid Alem´an Gonzales, Charles Ignacio Mendoza Mollocondo, Nayer Tumi Figueroa, Juan Carlos Juarez Vargas, Godofredo Quispe Mamani

PAGES 1084 – 1090

Paper 107: Attention-Guided Bidirectional Temporal Modelling with Graph-Based Regional Spatial Context for Bajra Crop Yield Prediction

Authors: Mamta Kumari, Suman, Devendra Prasad

PAGES 1091 – 1103

Paper 108: Scalability of Predictive Models on Multi-Core CPUs and GPUs: An Empirical Analysis

Authors: Atif Mahmood, Wan Joe Dean, P. Ganesh Kumar, Adnan N. Qureshi

PAGES 1104 – 1112

A Comparison of Metaheuristic Methods for the Vehicle Routing Problem

Manal El Jaouhari[1], Ghita Bencheikh[2], Ghizlane Bencheikh[3]
Moulay Ismaïl University of Meknes, Morocco[1, 3]
LINEACT, Cesi Engineering School, Assat, France[2]

***Abstract*—The Capacitated Vehicle Routing Problem (CVRP) is a fundamental NP-hard combinatorial optimization problem with important applications in logistics and distribution systems. Although numerous advanced approaches have been proposed in recent years, systematic benchmarking of classical metaheuristic algorithms under a unified experimental framework remains limited. This study evaluates the performance and trade-offs of four well-known metaheuristics: Hill Climbing (HC), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Genetic Algorithms (GA). All methods are implemented within the same computational environment and assessed on benchmark CVRP instances, using the CPLEX exact solver as a reference for global optimality. The results indicate that ACO achieves the smallest optimality gaps and often approaches optimal solutions, at the cost of higher computational effort. PSO strikes a favorable balance between solution quality and runtime across the tested instances, whereas HC delivers very fast solutions but degrades as problem complexity increases. GA exhibits higher variability and less competitive performance under the selected parameter settings. Overall, this comparative analysis highlights the strengths and limitations of classical metaheuristics and establishes a reproducible baseline for future research, including hybrid and learning-assisted approaches for scalable vehicle routing optimization.**

Keywords*—*Metaheuristics; Ant Colony Optimization; Hill Climbing; Genetic Algorithm; Particle Swarm Optimization; exact algorithm; Capacitated Vehicle Routing Problem

I. Introduction

The Vehicle Routing Problem (VRP), first introduced by Dantzig and Ramser [3], is one of the most important problems in combinatorial optimization and has many applications in logistics and transportation. One of its best-known variants is the Capacitated Vehicle Routing Problem (CVRP). In this problem, a fleet of vehicles with limited capacity must deliver goods to a set of customers, and the objective is to minimize the total travel cost.

The CVRP is an NP-hard problem, meaning that it becomes very difficult to solve exactly when the number of customers increases. Exact optimization methods [14], [16] can find optimal solutions, but their computational cost grows quickly, which limits their use to small problem instances. For this reason, heuristic and metaheuristic algorithms are widely used to obtain good-quality solutions in a reasonable time, especially for medium- and large-scale CVRP instances.

Over the years, many classical metaheuristics have been applied to solve the CVRP, such as Genetic Algorithms (GA) [7], [8], Ant Colony Optimization (ACO) [4], [5], Particle Swarm Optimization (PSO) [9], [15], and Hill Climbing (HC). In addition, modern state-of-the-art VRP solvers often rely on more advanced strategies, including Adaptive Large Neighborhood Search (ALNS), Variable Neighborhood Search (VNS), Tabu Search, and hybrid metaheuristics that combine different optimization mechanisms to improve performance on large-scale instances [10], [11], [17], [18], [19].

More recently, learning-assisted methods, such as reinforcement learning approaches, have also gained attention as promising directions for solving routing problems. These developments further highlight the importance of establishing reliable baselines through systematic benchmarking of classical metaheuristics [2], [6].

Despite the large number of studies in this area, direct comparisons of classical metaheuristics under the same experimental conditions remain limited. Many works focus on improving a single algorithm or use different benchmarks and implementations, which makes it difficult to clearly understand the relative strengths and weaknesses of these approaches [1], [21].

The main objective of this study is to provide a structured comparison of four widely used metaheuristic algorithms (HC, ACO, PSO, and GA) for solving the CVRP. All algorithms are implemented within the same experimental framework and tested on identical problem instances. Their performance is evaluated in terms of solution quality and computational time, using optimal solutions obtained with CPLEX as reference.

This study is structured as follows: Section II reviews related work on metaheuristics for CVRP. Section III presents the mathematical formulation of the problem. Section IV describes the four metaheuristic algorithms evaluated in this study. Section V discusses the experimental results, and Section VI concludes the study and suggests directions for future research.

II. Related Work

The CVRP has been widely studied using constructive heuristics, local search methods, population-based metaheuristics, and hybrid matheuristic frameworks. Although significant progress has been achieved, systematic benchmarking of classical metaheuristics under unified experimental conditions remains relatively limited. As a result, their comparative strengths and weaknesses are not always clearly established.

Avdoshin and Beresneva [1] provide one of the few comparative studies based on a common framework, focusing on constructive heuristics [25] such as nearest neighbor, savings, sweep, and insertion methods. Tiwari and Sharma [21] further investigate local search and Tabu Search strategies for last-mile delivery, comparing neighborhood structures under controlled settings. While informative, these works mainly address local search-based approaches and do not cover broader metaheuristic families. Tan and Yeh [20] note that many recent VRP studies rely on different or newly generated datasets, which makes it difficult to compare algorithms under a unified standard. They emphasize the need for publicly available benchmark datasets to improve comparability across studies.

In recent years, research has increasingly shifted toward advanced and state-of-the-art methods. Variable Neighborhood Search (VNS) and its extensions have been successfully applied to several VRP variants with capacity and time constraints [11], [19], [24]. Adaptive Large Neighborhood Search (ALNS) has also emerged as one of the most effective frameworks for large-scale routing problems, with important contributions on operator design and adaptive mechanisms [17],[18],[23]. In parallel, hybrid matheuristics combining metaheuristics with exact optimization components have achieved highly competitive results for the CVRP and related routing problems [13],[14],[16].

Despite these advances, existing studies often rely on different benchmarks, problem variants, or experimental protocols, which makes direct comparison across algorithms difficult. In particular, a controlled evaluation of classical metaheuristics such as Hill Climbing, Ant Colony Optimization, Particle Swarm Optimization, and Genetic Algorithms under identical conditions is still missing.

To address this gap, we implement HC, ACO, PSO, and GA within a single unified experimental framework and benchmark their performance against exact optimal solutions obtained via CPLEX. By providing a reproducible baseline, this work supports future research on hybrid, adaptive, VNS-based, and learning-assisted approaches for scalable vehicle routing optimization.

III. Mathematical Formulation

- Notation

N : Set of nodes, including the depot and customers.

K : Number of vehicles.

d_{ij} : Euclidean distance between node i and node j.

q_j : Demand of customer j.

C : Vehicle capacity.

x_{ijk} : Binary decision variable that equals 1 if vehicle k travels from node i to node j, and 0 otherwise.

u_{ik} : Continuous variable representing the load of vehicle k when leaving node i.

- Objective Function

The objective is to minimize the total distance traveled by all vehicles:

$$Min \ \sum_{k=1}^{K} \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ijk} \quad (1)$$

- Constraints

Capacity Constraints: Ensure that the total demand served by each vehicle does not exceed its capacity:

$$\sum_{j \in N} q_j \sum_{i \in N} x_{ijk} \leq C \ / \ k \in \{1, .., \mathrm{K}\} \quad (2)$$

Visit Constraints: Each customer must be visited exactly once by one vehicle:

$$\sum_{k=1}^{K} \sum_{i \in N} x_{ijk} = 1 \ / \ \forall j \in N \quad (3)$$

Flow Conservation Constraints: Ensure that if a vehicle arrives at a node, it must leave that node:

$$\sum_{j \in N} x_{ijk} = \sum_{j \in N} x_{jik} \ / \ \forall i \in N, \forall k \in \{1, .., \mathrm{K}\} \quad (4)$$

Depot Constraints: Each vehicle must start and end at the depot:

$$\sum_{j \in N} x_{0jk} = 1 \ / \ \forall k \in \{1, .., \mathrm{K}\} \quad (5)$$

$$\sum_{j \in N} x_{j0k} = 1 \ / \ \forall k \in \{1, .., \mathrm{K}\} \quad (6)$$

Subtour Elimination (MTZ Constraints): To prevent subtours (cycles that do not include the depot), we adopt the classical Miller–Tucker–Zemlin (MTZ) formulation [12]:

$$u_{jk} \geq u_{ik} + \ q_j - C \, . \left(1 - x_{ijk}\right) \ / \ \forall i \in N, \forall j \in N \backslash \{0\}, \mathrm{i} \neq \mathrm{j}, \forall k \in \{1, .., \mathrm{K}\} \quad (7)$$

The load variables are bounded as follows:

$$q_j \leq u_{ik} \leq \ C \ / \ \forall i \in N \ , \forall k \in K \quad (8)$$

These constraints eliminate subtours by ensuring that the vehicle load increases consistently along each route, thereby forcing all visited customers to be connected to the depot.

IV. Metaheuristic Algorithms

Metaheuristic algorithms are generally classified into two categories: single solution-based algorithms and population-based algorithms [1]. In this study, we include algorithms from both categories: HC as a single solution-based metaheuristic, and ACO, PSO, and GA as population-based metaheuristics. These methods are selected as representative classical approaches, providing a reproducible baseline for comparing solution quality and computational efficiency on the Capacitated Vehicle Routing Problem (CVRP) [22]. In the following subsections, we briefly describe each algorithm and its adaptation to the CVRP.

A. Hill Climbing

Hill Climbing is a local search optimization algorithm that starts from an initial solution and iteratively improves it by moving to a neighboring solution with a better objective value. At each step, the algorithm selects the best available improvement in the local neighborhood and continues the search until no further improvement is possible. The process stops when a local optimum is reached.

The application of HC to the CVRP is summarized in Algorithm 1:

Algorithm 1: HC Algorithm applied to the CVRP

x_0 : Initial solution
x : Current solution
x^* : Best solution
$x \leftarrow x_0$
$x^* \leftarrow x_0$
While *True* do
  $x \leftarrow \underset{x_i \text{ neighbor of } x}{\text{argmin}} f(x_i)$
  if ($f(x) < f(x^*)$) then
    $x^* \leftarrow x$
  else
    Break
  End
End

Solutions are represented as permutations of customer indices, with vehicle routes separated by depot markers. The neighborhood operator generates new solutions through 2-opt exchanges within routes and relocation moves between routes, ensuring feasibility by checking capacity constraints before accepting modifications.

B. Ant Colony Optimization

Ant Colony Optimization (ACO) is a metaheuristic inspired by the behavior of ants in nature. Artificial ants construct their solutions by selecting components based on two values: the pheromone trails $(\tau_{ij})_{i,j\in N}$ and heuristic information $(\eta_{ij})_{i,j\in N}$, in the CVRP case, it's the inverse of the distance between customers. The pheromone trails are updated over time to reinforce good solutions and reduce the influence of poor ones.

The application of ACO to the CVRP is summarized by Algorithm 2:

Algorithm 2: ACO Algorithm applied to the CVRP

Initialize the pheromone trail by a small value : $\tau_{ij} \leftarrow \tau_0$
For (a Maximal number of iterations) do
  For (each ant k) do
    Initialize the current vehicle v by 1 : $v \leftarrow 1$
    Initialize the vehicle's route by depot 0
    While (The set of unserved clients is not empty) do
      Select a client j according to :

$$P_{ij} = \frac{\eta_{ij}^{\alpha}.\tau_{ij}^{\beta}}{\sum_l \eta_{il}^{\alpha}.\tau_{il}^{\beta}}$$

      If (vehicle capacity is not exeed) then
        Insert j into the route
      else
        return to depot and $v \leftarrow v + 1$
      End
      Remove j from the unserved clients set
    End
  End
  Calculate the Best Solution
  Update the pheromone
End
End

Each ant constructs a solution by sequentially selecting customers using the probabilistic transition rule. Feasibility is maintained by tracking cumulative vehicle load and returning to the depot when adding the next customer would violate capacity constraints.

C. Genetic Algorithm

Genetic Algorithms (GA) are population-based metaheuristics inspired by Darwin's theory of evolution. A set of solutions (population) evolves over a given number of iterations by applying genetic operators such as selection, crossover, and mutation. The quality of a solution (individual), called fitness, depends on the objective function of this solution. At each iteration, several individuals are selected according to their fitness in order to create new ones by applying two operators: Crossover and Mutation. At the end of the iteration, a phase of replacement is applied to select the individuals that pass to the next iteration. Algorithm 3 presents the application of GA to the CVRP.

Algorithm 3: GA applied to the CVRP

Generate the initial population
For (a maximal number of iterations) do
  Evaluate individuals
  Select a subset of individuals for crossover
  Apply crossover operator to selected individuals taken two by two.
  Select a subset of individuals for mutation
  Apply mutation operator to selected individuals.
  Replace the current population by the new one
  Calculate the best solution
End

Individuals are encoded as permutations of customers. The crossover operator preserves relative customer order from parents, while mutation randomly swaps two customers. Infeasible solutions are repaired by redistributing customers among vehicles to satisfy capacity constraints.

D. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a population-based metaheuristic inspired by the collective behavior of social organisms such as bird flocks or fish schools. In PSO, each particle represents a candidate solution and moves through the search space by updating its position based on its own best

solution and the best solution found by the swarm. The application of PSO to the CVRP is presented in Algorithm 4.

Algorithm 4: PSO applied to the CVRP

```
Initialize particles with random positions and velocities
Calculate the best solution x*
For (a maximal number of iterations) do
    For (each particle p) do
        If (f(p) < f(x*)) then
            x* ← p
        End
    p ← Update personal best position
    End
    For (each particle i)
        Update velocity for particle i
        Update position for particle i
    End
End
```

Particle positions are represented as priority-based encodings, where real-valued priorities determine customer visit order. Positions are decoded into feasible routes using a greedy insertion procedure that respects vehicle capacity constraints.

The detailed parameter settings for each algorithm, including population size, iteration limits, and coefficients, are reported in Section V. All algorithms are executed multiple times to account for stochastic behavior, and results are averaged over these runs to ensure statistical reliability.

V. Experiment Results

To evaluate the performance of the four metaheuristic algorithms, we implemented them in Python and applied them to randomly generated CVRP instances. The number of clients in each instance varies from 5 to 15, while the number of vehicles ranges from 2 to 5. Exact solutions for these instances were obtained using the CPLEX solver in Python, providing a reference for solution quality.

The CVRP instances used in this study are deliberately small (5–15 customers) so that they can be directly compared with exact optimal solutions obtained using CPLEX. This creates a controlled testing environment where the solution quality of each metaheuristic can be evaluated in a reliable way. Although these instances are not representative of large real-world logistics problems, they provide a useful baseline for understanding and comparing the behavior of classical metaheuristic algorithms before considering larger instances in future work.

All computations were performed on a laptop with an Intel® Core™ i5-8300H processor (2.30 GHz) and 8 GB of RAM. Each metaheuristic was executed multiple times to account for its stochastic nature, and the results reported are averaged over these runs. Detailed parameter settings for each algorithm are presented in the following subsection.

A. Parameter Settings of Metaheuristic Algorithms

The parameter values for all four metaheuristic algorithms (HC, ACO, PSO, and GA) were chosen based on preliminary tests on a small set of instances. Table I shows the settings used for all methods.

TABLE I. Parameter Settings for Metaheuristic Algorithms

Algorithm	Parameter	Value
HC	Number of iterations	1000
ACO	Number of ants	100
	Pheromone importance (α)	1
	Heuristic importance (β)	3
	Pheromone deposit factor	70
GA	Population size	150
	Number of generations	500
	Mutation rate	0.1
PSO	Number of particles	50
	Number of iterations	200
	Inertia weight (w)	0.4
	Cognitive coefficient (c_1)	1.5
	Social coefficient (c_2)	1.7

B. Computational Results

Table II presents the computational results obtained for the CVRP using the four metaheuristic algorithms and compares them with the exact optimal solutions provided by the CPLEX solver.

For each problem instance, the table reports the optimal value found by CPLEX as well as the results obtained by each metaheuristic method. The "Optimal Value" column corresponds to the exact solution returned by CPLEX, while the "CPU" column indicates the computational time (in seconds) required to obtain this solution. Table II shows computational results.

Since metaheuristic algorithms are stochastic, each experiment was repeated 10 independent times for every instance. For each algorithm, the solution value reported corresponds to the best result obtained over these 10 executions.

For the metaheuristic algorithms, three performance indicators are presented:

- BD (Best Distance): the best solution value obtained over the 10 runs.
- GAP (%): the relative difference between BD and the optimal solution provided by CPLEX. It is computed as follows:

$$GAP = \frac{BD - Optimal\ value}{Optimal\ value} \times 100 \quad (8)$$

- CPU: the execution time required by each algorithm.

The problem instances are grouped according to their size and the number of vehicles considered in each case.

TABLE II. COMPUTATIONAL RESULTS

Instances	CPLEX		ACO			HC			GA			PSO		
	Optimal Value	CPU	BD	GAP	CPU	BD	GAP	CPU	BD	GAP	CPU	BD	GAP	CPU
A-n5-k2	737,3459	0,03	737,3459	0%	0,6423	737,3459	0%	0,0015	737,3459	0%	3,4149	737,3459	0%	0,2787
B-n10-k3	1488,97	0,92	1488,9701	0%	5,4658	1590,455	7%	0,0266	1747,5083	17%	7,6644	1698,1143	14%	0,9505
C-n10-k3	1043,24	0,05	1043,2374	0%	1,9157	1184,322	14%	0,0165	1284,0174	23%	8,3398	1076,0501	3%	0,4345
D-n10-k3	1345,6564	0,09	1345,6565	0%	2,1337	1410,9831	5%	0,013	1618,81817	20%	19,3497	1407,0059	5%	0,3069
E-n10-k3	1344,8953	0,28	1344,8954	0%	1,4901	1344,8954	0%	0,0207	1931,4923	44%	18,35	1561,5705	16%	0,3965
F-n10-k3	1205,1432	0,05	1205,1432	0%	1,9442	1205,1432	0%	0,0202	1578,3889	31%	17,0545	1312,8234	9%	0,3379
G-n10-k3	1055,0557	0,12	1119,7954	6%	1,9904	1219,6917	16%	0,0085	1216,5738	15%	19,7098	1226,3541	16%	0,3473
H-n10-k3	1164,7701	0,05	1164,7702	0%	2,3681	1164,7702	0%	0,0368	1511,8412	30%	7,8897	1506,6458	29%	0,3494
X-n15-k5	1292,7975	0,11	1292,7975	0%	4,5217	1292,7975	0%	0,1402	2459,6503	90%	13,9031	1814,1539	40%	0,2681
A-n15-k5	1812,6965	0,14	1903,1864	5%	4,6785	2255,2393	24%	0,0693	2396,3247	32%	11,7859	2292,7553	26%	2,5267
B-n15-k5	1432,8015	0,34	1432,8015	0%	4,4889	1549,2988	8%	0,0847	2592,7037	81%	11,7976	1863,9569	30%	0,2843
C-n15-k5	1403,0113	0,5	1403,0114	0%	3,4049	1952,2029	39%	0,0747	2582,2474	84%	12,075	1852,3453	32%	0,2852
D-n15-k5	1437,4749	0,06	1437,4749	0%	4,2683	1731,5309	20%	0,1034	2590,5014	80%	12,1401	1742,3011	21%	0,2902
E-n15-k5	1469,6608	0,22	1469,6609	0%	3,2084	1473,7939	0%	0,1027	2663,3091	81%	15,9196	1862,7521	27%	0,2625
F-n15-k5	1677,3105	0,05	1677,3106	0%	4,4511	1695,7807	1%	0,0993	2543,7429	52%	12,1651	2165,1462	29%	0,4388
G-n15-k5	1575,154	0,53	1593,1939	1%	4,5538	1978,5961	26%	0,0461	2134,9409	36%	12,4126	2212,5714	40%	0,4285
H-n15-k5	1401,1991	0,08	1411,4728	1%	4,4402	1403,0068	0%	0,106	2216,4903	58%	10,8925	2218,7381	58%	0,4055
I-n15-k5	1673,7506	0,23	1673,7506	0%	4,5742	1789,8502	7%	0,0939	2503,7557	50%	13,0543	2705,0827	62%	0,486

According to these results, we note that for small instances with 5 customers, all metaheuristics reached the optimal solution. With small differences in execution time, HC achieves the solution almost instantly, while ACO and PSO require more computational time. This indicates that for very small problem sizes, simple local search methods are highly efficient.

For instances with 10 customers, we observe differences between the algorithms in terms of solution value and computational time. ACO consistently provides high-quality solutions, often reaching the optimal solution or maintaining a very small gap. However, this accuracy comes at the cost of higher computational times compared to HC and PSO. Hill Climbing remains extremely fast, but its solution quality degrades as the problem becomes more complex, with gaps reaching up to 44% in some instances. Genetic Algorithm (GA) shows moderate performance, producing acceptable solutions but with relatively long execution times. PSO generally achieves a good compromise between solution quality and computational efficiency, maintaining low gaps in most cases while keeping execution times reasonable.

When the problem size increases to 15 customers, the differences between the algorithms become clearer. ACO continues to provide the most accurate solutions, often reaching the optimal value or showing very small gaps. However, its execution time increases significantly. HC remains the fastest algorithm, but its solution quality decreases with very large gaps that can exceed 80%, which limits its use for larger instances. GA shows weak performance in both solution quality and execution time, with large gaps and slow convergence. PSO exhibits more stable behavior, achieving better solutions than HC and GA while keeping reasonable execution times, although it does not always reach the optimal solution.

To better visualize the differences between the metaheuristics in terms of CPU time, Fig. 1 presents the execution time required by each algorithm for all instances.

As discussed previously, we observe that the Genetic Algorithm is the most time-consuming metaheuristic, requiring significantly more CPU time than the other methods. While Hill Climbing (HC) is the fastest algorithm, obtaining solutions almost instantly. ACO needs moderate computational times, generally higher than HC and PSO but lower than GA. PSO demonstrates a balanced performance, maintaining reasonable execution times across all instances.

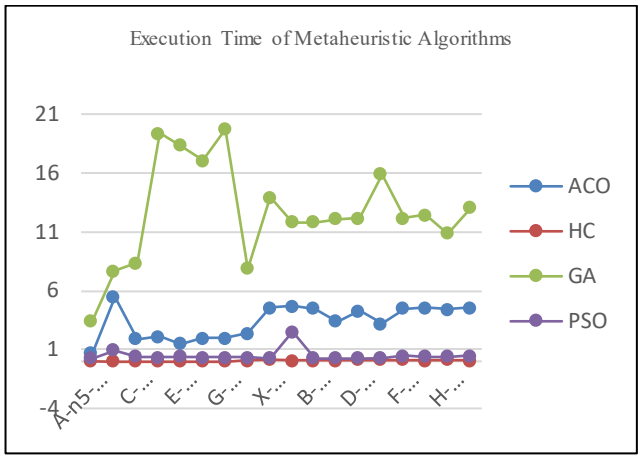


Fig. 1. Execution time of metaheuristic algorithms.

Fig. 2 presents the performance gaps of the four metaheuristic algorithms relative to the optimal solutions obtained by CPLEX.

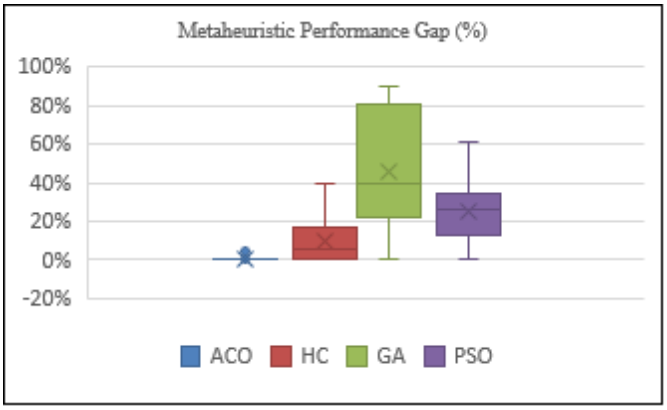


Fig. 2. Metaheuristic performance gap (%).

As shown in Fig. 2, ACO consistently achieves the smallest gaps, with median values near zero and little variation across all instances, confirming its high solution quality. HC performs well on small instances with very low gaps, but its gaps increase and become more variable as the problem size grows, showing that it is less effective for larger instances. GA has large gaps and wide variation, indicating inconsistent performance and slower convergence. PSO achieves moderate gaps with relatively stable performance, offering a good balance between solution quality and reliability.

VI. Conclusion and Future Directions

This study evaluated the performance of four metaheuristics (ACO, PSO, HC, and GA) for solving the CVRP using randomly generated instances. The algorithms were implemented under a unified framework and benchmarked against optimal solutions obtained via CPLEX.

The results reveal distinct performance trade-offs among the methods, consistent with the "no free lunch" theorem: no single metaheuristic dominates across all problem characteristics. ACO consistently achieved the highest solution quality, frequently matching or closely approximating optimal values, though at the cost of substantially longer computation times. PSO demonstrated the most balanced performance, maintaining reasonable solution quality while keeping execution times low, making it well-suited for time-sensitive applications. HC excelled in computational efficiency, generating solutions almost instantaneously, but exhibited significant performance degradation as problem complexity increased, with optimality gaps exceeding 80% on the largest instances tested. GA showed the weakest overall performance in our experiments, requiring the longest execution times while producing solutions with large and inconsistent optimality gaps. This latter result warrants further investigation, as it may reflect parameter sensitivity or implementation-specific issues rather than fundamental algorithmic limitations.

The current study has important limitations. Our analysis focused on small-scale instances (5-15 customers) to enable rigorous validation against exact solutions. Additionally, the relatively poor GA performance diverges from typical findings in the literature and requires further examination.

Future research will proceed along two complementary directions. First, we will extend this comparative analysis to larger problem instances (50-200 customers) and incorporate additional modern metaheuristics, including ALNS, VNS, and ILS, to better understand algorithmic scalability and behavior under more challenging conditions. Second, given the complementary strengths observed across methods, we will investigate hybrid and adaptive approaches. Specifically, we plan to develop a multi-agent framework that dynamically selects or combines these metaheuristics based on problem characteristics. This approach aims to leverage the solution quality of ACO, the efficiency of PSO, and the speed of HC within an integrated system that adapts to instance-specific features, potentially offering more robust performance across diverse CVRP scenarios.

References

[1] Avdoshin, S. M., & Beresneva, E. N. (2019). Constructive heuristics for Capacitated Vehicle Routing Problem: a comparative study. Proceedings of ISP RAS, 31(3), 145–156.

[2] Bencheikh, G. (2024). Metaheuristics and machine learning convergence: A comprehensive survey and future prospects. In Metaheuristic and Machine Learning Optimization Strategies for Complex Systems (pp. 276–322). Springer.

[3] Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. Management Science, 6(1), 80–91.

[4] Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics—Part B, 26(1), 29–41.

[5] Dorigo, M., & Stützle, T. (2019). Ant colony optimization: Overview and recent advances. In M. Gendreau & J.-Y. Potvin (Eds.), Handbook of Metaheuristics (3rd ed., pp. 311–351). Springer.

[6] El Jaouhari, M., Bencheikh, G., & Bencheikh, Gh. (2025). Metaheuristic and reinforcement learning techniques for solving the vehicle routing problem: A literature review. Journal of Traffic and Transportation Engineering (English Edition). In press.

[7] Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press.

[8] Kala, R. (2024). An introduction to evolutionary computation. In R. Kala (Ed.), Autonomous Mobile Robots (pp. 715–759). Academic Press.

[9] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. Proceedings of ICNN'95 - International Conference on Neural Networks, 1942–1948.

[10] Lubnina, E., & Kochetov, Y. (2024). VNS algorithm for the periodic pick-up and delivery helicopter routing problem in oil and gas offshore. Yugoslav Journal of Operations Research, 34(3), 407–421.

[11] Matijević, L., Ilin, V., Davidović, T., Jakšić-Krüger, T., & Pardalos, P. M. (2024). General VNS for asymmetric vehicle routing problem with time and capacity constraints. Computers & Operations Research, 167, 106630.

[12] Miller C.; Tucker A.; Zemlin R. (1960) Integer programming formulations and travelling salesman problems. J. ACM 7: 326-329.

[13] Xue, N., Bai, R., Qu, R., & Aickelin, U. (2021). A hybrid pricing and cutting approach for the multi-shift full truckload vehicle routing problem. European Journal of Operational Research, 292(2), 500–514

[14] Pecin, D., Pessoa, A., Poggi, M., & Uchoa, E. (2017). Improved branch-cut-and-price for capacitated vehicle routing. Mathematical Programming Computation, 9(1), 61–100.

[15] Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization: An overview. Swarm Intelligence, 1(1), 33–57.

[16] Sadykov, R., Uchoa, E., & Pessoa, A. (2021). A bucket graph-based labeling algorithm with application to vehicle routing. Transportation Science, 55(1), 4–28.

[17] Shi, J., Mao, H., Zhou, Z., & Zheng, L. (2023). Adaptive large neighborhood search algorithm for the unmanned aerial vehicle routing problem with recharging. Applied Soft Computing, 147, 110831.

[18] Shi, Y., Liu, W., & Zhou, Y. (2023). An adaptive large neighborhood search based approach for the vehicle routing problem with zone-based pricing. Engineering Applications of Artificial Intelligence, 124, 106506.

[19] Tadaros, M., Sifaleras, A., & Migdalas, A. (2024). A variable neighborhood search approach for solving a real-world hierarchical multi-echelon vehicle routing problem involving HCT vehicles. Computers & Operations Research, 165, 106594.

[20] Tan, S. Y., & Yeh, W. C. (2021). The vehicle routing problem: State-of-the-art classification and review. Applied Sciences, 11(21), 10295.

[21] Tiwari, K. V., & Sharma, S. K. (2023). An optimization model for vehicle routing problem in last-mile delivery. Expert Systems with Applications, 222, 119789.

[22] Toth, P., & Vigo, D. (Eds.). (2014). Vehicle Routing: Problems, Methods, and Applications (2nd ed.). SIAM.

[23] Voigt, S. (2025). A review and ranking of operators in adaptive large neighborhood search for vehicle routing problems. European Journal of Operational Research, 322(2), 357–375.

[24] Woller, D., Kozák, V., Kulich, M., & Přeučil, L. (2025). Variable Neighborhood Search for the Electric Vehicle Routing Problem. arXiv preprint.

[25] Lima, S., Santos, R., De Araujo, S. A., & Schimit, P. (2016). Combining genetic algorithm with constructive and refinement heuristics for solving the capacitated vehicle routing problem. In IFIP International Conference on Advances in Production Management Systems (APMS) (pp. 113–121).

A Confidence-Aware Multi-Layer Framework for Drone Forensic Inference Using Heterogeneous Digital Evidence Sources and Flight Logs Analysis

Nidhiba Parmar, Naveen Kumar Chaudhary
School of Cyber Security and Digital Forensics, National Forensic Sciences University, Gandhinagar, India

***Abstract*—The increasing use of unmanned aerial vehicles (UAVs) in criminal and adversarial contexts has created new challenges for digital forensic investigations. Current UAV forensic research primarily emphasizes artefact extraction and platform-dependent analysis, while insufficient attention has been given to uncertainty modelling and confidence quantification in forensic inference. This study addresses this methodological gap by proposing a confidence-aware multi-layer UAV forensic framework designed to support legally defensible forensic conclusions. The framework integrates chip-off memory acquisition, logical flight log analysis, companion mobile device artefact examination, and wireless trace correlation within a unified analytical architecture. Physics-based flight trajectory reconstruction and cross-device temporal alignment algorithms enhance reproducibility and platform independence. To reflect varying levels of evidentiary reliability, a structured evidence-weighting approach is introduced alongside a novel Forensic Confidence Index (FCI) that quantifies evidentiary support without implying absolute certainty. Validation using a Yuneec Typhoon Q500 4K dataset demonstrates feasible trajectory reconstruction, temporal correlation, and confidence-constrained attribution under realistic investigative conditions. By explicitly incorporating uncertainty modeling and confidence articulation into UAV forensic workflows, the proposed framework improves scientific rigor, transparency, and legal defensibility while providing a scalable foundation for future cyber-physical forensic investigations.**

Keywords*—*UAV forensics; drone investigations; forensic inference; confidence quantification; flight log analysis; digital forensic framework

I. Introduction

The operational footprint of unmanned aerial vehicles (UAVs) has expanded beyond recreational and commercial domains into increasingly adversarial and criminal contexts, including illicit surveillance, cross-border smuggling, critical-infrastructure reconnaissance, and targeted cyber-physical intrusions [1]. The proliferation of UAV use has concomitantly increased the importance of drones from artefacts to a primary vector of evidentiary importance in digital forensics investigations [2]. However, contemporary UAV forensic methodology is largely data-centric (focusing more on extracting information from data than on forensic inference), which limits its probative and explanatory value in high-stakes investigative and judicial settings [3], [4], [5].

Recent UAV forensic studies between 2023 and 2025 have shown significant progress in terms of systematic cataloguing of the artefact locations, demystifying proprietary flight logs, and retrieving multimedia and telemetry information from widely used drone platforms, especially the DJI and, to a lesser extent, Yuneec and Parrot ecosystems [6], [7], [8]. However, these contributions are less than comprehensive in this regard; they outline the data that can be obtained from such types of evidence, but not the degree of confidence with which such data can be used to support forensic attribution, a point in time, or the association with an operator. Consequently, many essential forensic issues related to evidentiary reliability, cross-source consistency, and the propagation of uncertainty are often left implicit or unanswered. A critical, unanswered problem in modern UAV forensics is the absence of formal mechanisms for quantifying evidentiary confidence when analyzing diverse, heterogeneous sources, including chip-off memory, removable storage, companion mobile devices, and wireless communication traces.

Specifically, this study makes the following contributions:

- Proposes a unified multi-layer UAV forensic architecture integrating physical, logical, companion-device, and wireless evidence sources.
- Introduces formal algorithms for flight path reconstruction and cross-device temporal alignment.
- Develops uncertainty-aware evidence weighting model grounded in forensic reliability principles.
- Defines a novel Forensic Confidence Index (FCI) to quantify attribution strength.
- Validates the framework through a Yuneec Typhoon Q500 4K case study while ensuring device-agnostic generalizability.

Although UAV forensic research has advanced in artefact extraction and platform-specific analysis, there remains a lack of structured frameworks for confidence-based forensic inference across heterogeneous evidence sources. Existing approaches rarely incorporate uncertainty modelling or reliability weighting, which limits the scientific rigor and legal defensibility of forensic conclusions. This study addresses this gap by proposing a confidence-aware multi-layer UAV forensic framework integrating diverse evidence sources with explicit confidence quantification.

The remainder of this study is organized as follows: Section II reviews related work and identifies key

methodological gaps in UAV forensic research. Section III presents the theoretical foundation underlying confidence-aware forensic inference. Section IV describes the proposed multi-layer UAV forensic framework and methodological approach. Section V details the associated analytical models and algorithms. Section VI presents the case study validation using the Yuneec Typhoon Q500 4K dataset. Section VII discusses comparative and theoretical implications of the findings. Section VIII addresses generalization, admissibility, and practical deployment considerations. Section IX outlines study limitations and future research directions, and finally, Section X concludes the study.

II. Related Work and Gap Synthesis

The rapid proliferation of unmanned aerial vehicles (UAVs) across civilian, commercial, and adversarial domains has significantly increased the importance of UAV digital forensic investigations. Recent research has focused on artefact extraction, flight log interpretation, and forensic framework development; however, methodological maturity in forensic inference, particularly regarding uncertainty quantification and confidence modeling, remains limited.

Recent UAV forensic studies emphasize systematic artefact acquisition and device-specific forensic analysis. For example, recent forensic investigations of modern DJI drones have demonstrated the feasibility of extracting telemetry logs, encryption structures, and internal storage artefacts through structured parsing frameworks [9]. Similarly, controller-level forensic analysis has shown that remote controller data can reveal pilot identity, pairing timestamps, and operational locations, thereby expanding evidentiary scope beyond onboard storage [10].

Several contemporary studies have proposed broader forensic frameworks. A conceptual digital forensic investigation model for UAVs was introduced to guide the identification, preservation, analysis, and documentation of drone evidence [11]. Likewise, recent survey research has categorized UAV forensic artefacts, tools, and frameworks while identifying unresolved research themes such as evidence standardization and reproducibility [12]. These studies highlight the growing recognition of UAV forensics as a specialized domain but also indicate persistent methodological fragmentation.

Emerging research between 2024 and 2025 has increasingly focused on integrating diverse forensic data sources. Case studies involving DJI Mini and Phantom drones demonstrate that correlating internal storage, controller data, and flight telemetry can improve event reconstruction accuracy [13], [14]. Additionally, recent investigations into live and static digital evidence traceability have emphasized challenges related to data integrity, repeatability, and evidentiary admissibility [15]. These challenges are particularly critical when forensic findings are presented in legal contexts.

Innovative methodological approaches are also emerging. Digital twin technology has recently been explored for drone accident reconstruction, enabling simulation-based validation of forensic hypotheses and improved investigative accuracy [16]. Metamodeling approaches have likewise been proposed to standardize UAV forensic investigation processes across heterogeneous platforms [17]. Furthermore, recent cybersecurity-oriented forensic frameworks advocate real-time anomaly detection and automated evidence preservation mechanisms for UAV ecosystems [18].

Recent literature also highlights operational and legal challenges. Law-enforcement-focused research indicates that drone forensic investigations face difficulties related to proprietary software, encryption, anti-forensic techniques, and privacy concerns [19]. Similarly, studies on tool-based forensic analysis emphasize the need for standardized methodologies to improve evidentiary reliability and legal defensibility [20].

Another significant development is the exploration of multi-source data correlation. Studies involving industrial and security applications demonstrate that combining flight logs, SD card data, and mobile application artefacts can support attribution and event reconstruction [21]. Transformer-based approaches have also been proposed to link mobile devices with UAV activity, indicating increasing interest in AI-assisted forensic correlation [22].

Despite these advancements, several limitations persist. First, most existing studies focus on artefact extraction rather than forensic inference. Second, uncertainty modeling and confidence quantification are rarely explicitly addressed. Third, cross-layer evidence fusion is typically informal and lacks structured reliability weighting. Finally, empirical validation often relies on single case studies or platform-specific datasets, limiting generalizability.

Therefore, a critical methodological gap exists in UAV forensic research: the absence of a confidence-aware, multi-layer forensic inference framework capable of integrating heterogeneous evidence sources while explicitly modeling uncertainty. Addressing this gap is essential for improving the scientific rigor, transparency, and legal defensibility of UAV forensic investigations.

The present study addresses these limitations by proposing a structured multi-layer UAV forensic framework incorporating physics-based trajectory reconstruction, cross-device temporal alignment, explicit evidence reliability weighting, and a novel Forensic Confidence Index (FCI). This approach moves beyond artefact extraction toward confidence-aware forensic inference suitable for investigative and judicial applications.

A. Methodological and Theoretical Weaknesses

Despite substantial progress in UAV forensic artefact extraction and analytical tool development, several methodological limitations remain. First, uncertainty associated with UAV forensic data is rarely formally modelled, even though sensor noise, timestamp drift, synchronization errors, and partial data loss are inherent characteristics of cyber-physical evidence. Second, structured evidence fusion frameworks capable of systematically integrating heterogeneous sources with differing reliability profiles remain limited. Third, there is no widely accepted mechanism for expressing the strength of forensic attribution as a measurable confidence level, despite increasing emphasis on confidence reporting in forensic admissibility discussions.

These limitations constrain the evolution of UAV forensics from a primarily technical data extraction activity toward a mature forensic science discipline grounded in transparent inference and defensible interpretation. To further illustrate these methodological gaps, Table I presents a comparative analysis of recent UAV forensic studies, highlighting the limited treatment of uncertainty modelling, confidence quantification, and multi-layer evidence integration.

TABLE I. COMPARATIVE ANALYSIS OF UAV FORENSIC LITERATURE (2020–2025)

Author & Year	Study Focus / Platform	Evidence Sources	Methods / Approach	Uncertainty Treatment	Confidence Quantification	Key Limitations
(Zhao et al., 2024) [9]	DJI Mavic 2 Pro forensic analysis	Drone telemetry logs, mobile artefacts	Artefact extraction, structured log parsing	Not explicitly addressed	No	Platform-specific focus
(Lee et al., 2023) [10]	DJI remote controller forensic analysis	Controller logs, pairing data	Controller artefact examination	Limited discussion	No	Limited multi-source integration
(Alotaibi et al., 2023) [11]	Conceptual UAV forensic investigation model	General UAV artefacts	Investigation framework modelling	Conceptual treatment	No	Limited empirical validation
(Studiawan et al., 2023) [12]	UAV forensic survey study	Multi-source UAV evidence	Literature classification and analysis	Minimal discussion	No	Survey-based study
(Halim & Luthfi, 2025) ,[13]	UAV flight data forensic case study	Flight telemetry logs	Static and dynamic forensic analysis	Limited treatment	No	Application-specific validation
(Tombari Sibe & Bekom, 2025) ,[14]	DJI Phantom forensic investigation	UAV internal logs	Technical case study analysis	Not explicitly addressed	No	Single platform focus
(Salamh et al., 2021),[15]	Evidence traceability analysis	Static and live UAV artefacts	Comparative forensic analysis	Partial discussion	No	Earlier UAV ecosystem
(Almusayli et al., 2024) , [16]	Digital twin forensic reconstruction	Telemetry + simulation data	Digital twin modelling approach	Partial discussion	No	Simulation dependency
(Alhasan et al., 2025),[17]	Metamodel-based forensic review	Multi-source UAV evidence	Systematic literature review	Conceptual discussion	No	Limited operational validation
(Mohammed et al., 2025),[18]	UAV cybersecurity forensic readiness	Operational UAV data streams	Security + forensic readiness framework	Partial discussion	No	Security-oriented emphasis
(Thantilage et al., 2025),19]	Law enforcement drone forensics	Logs, encrypted UAV data	Operational forensic investigation	Not explicitly addressed	No	Legal/privacy constraints
(Viswanathan & Baig, 2020),[20]	Drone forensic tools study	Tool-generated artefacts	Tool evaluation study	Minimal discussion	No	Older tool ecosystem
(Ankit et al., 2025) ,[21]	Multi-source forensic extraction case study	Chip-off, SD card, controller artefacts	Data extraction and analysis	Limited treatment	No	Platform-specific validation
(Nieszporek et al., 2026)[22]	AI-based UAV forensic identification	Mobile–UAV telemetry	Transformer-based modelling	Partial discussion	No	Emerging research stage
This Work	Multi-layer UAV forensic framework	Chip-off memory + Flight logs + Mobile artefacts + Wireless traces	Physics-based reconstruction + Temporal alignment + Evidence fusion	Explicit reliability weighting	Yes (Forensic Confidence Index)	Single case validation (future expansion planned)

III. Theoretical Foundation of UAV Forensic Inference

When treated as a purely procedural matter, digital forensic science may degenerate into a repertoire of extraction methods that lack explanatory or even inferential coherence. This danger is especially urgent in UAV research, as heterogeneous information sources, unstable storage devices, and distributed control systems make it harder to apply conventional concepts of evidence persistence and attribution. A conceptually grounded framework is therefore necessary to advance UAV forensics to the next level of technical analysis in forensic science before the court.

This is based on the Exchange Principle of Locard, but redefined for the digital and cyber-physical spaces. In UAV flights, each flight triggers cross-turning tracks across several layers: memory registers embedded in the flight, removable storage, accompanying mobile devices, and wireless communication channels. These exchanges are neither homogeneous nor equally persistent; they fluctuate, varying in volatility, granularity, and resistance to manipulation. To achieve principled evidence integration, it is essential to identify and institutionalize this asymmetric exchange of trace [23].

It should also be noted that forensic reliability and forensic certainty should be critically distinguished. The reliability is the subject of trustworthiness of some artefact under specified circumstances of acquisition, whereas the certainty suggests an unconditional conjecture that can hardly be justified by digital evidence. Modern forensic science is more often dismissing pronouncements of certainty in favor of probabilistic or confidence-bounded conclusions [24]. An example of this is UAV data, which is susceptible to sensor noise, timestamp drift, packet loss, and partial overwriting.

Therefore, confidence modelling is not a choice in UAV forensics; it is a methodological requirement. Courts and expert review panels are increasingly demanding that investigators not only state what evidence points to, but also the degree of strength the evidence has for a particular inference and the conditions under which it is made. Without direct measures of confidence, UAV forensic determinations are prone to admissibility challenges.

IV. Proposed Multi-Layer UAV Forensic Framework

A. Multi-Layer Evidence Architecture

To put the above theoretical principles into practice, a multi-layer forensic architecture is proposed. The framework clearly captures UAV evidence as a stratified system, with each layer providing different levels of information, reliability properties, and volatility restrictions.

1) Layer 1: Physical Memory (Chip-off)

This layer comprises raw non-volatile memory extracted directly from the UAV's onboard storage via chip-off procedures. It provides the best evidentiary persistence and resistance to post-incident contamination and must be acquired by invasive means and with sophisticated laboratory controls. There is high reliability and low volatility.

2) Layer 2: Logical Logs (Internal/SD Storage)

The second layer comprises flight logs, configuration files, and cached telemetry on internal flash disks or removable SD cards. These artefacts provide valuable, well-organized records of operations, but can be obsolete, overwritten, or formatted by their users. The reliability is moderate to high, depending on the time of acquisition.

3) Layer 3: Companion Device Artefacts

UAV control mobile apps produce high-context data, such as coordinated timestamps, user feedback, and stalled flight data. Although these artefacts are very informative, they are subject to the operating system's abstraction and update cycles, which create interpretive uncertainty.

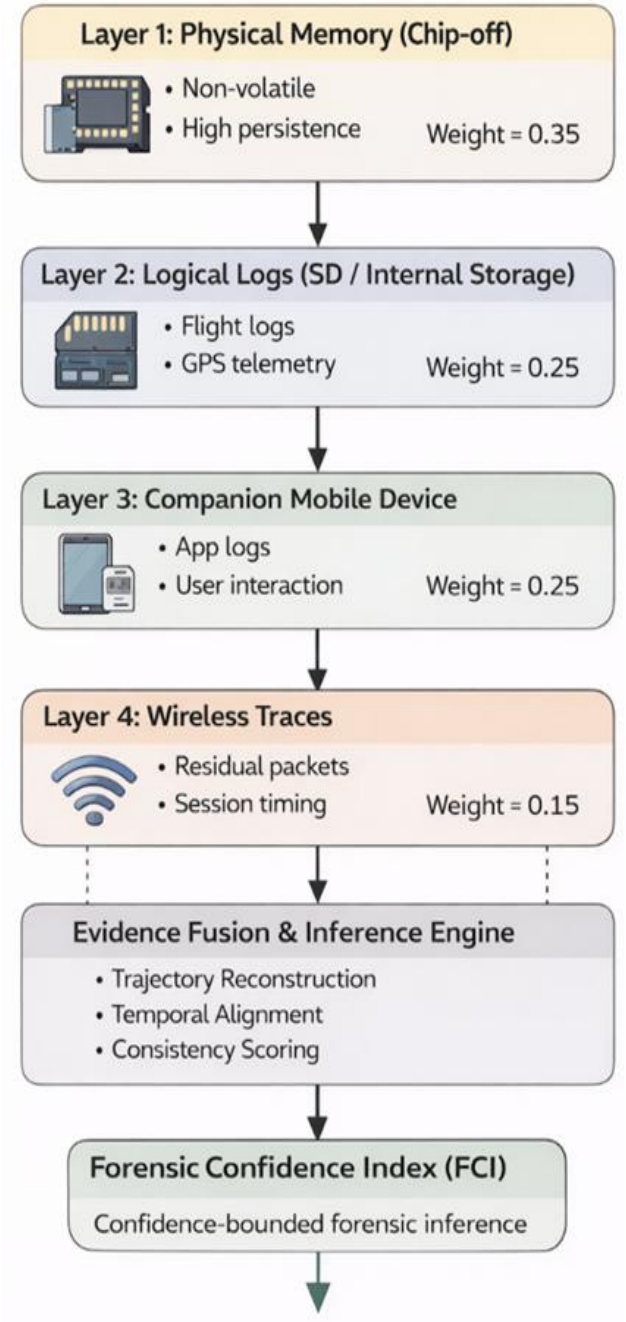


Fig. 1. Multi-layer UAV forensic evidence architecture.

4) Wireless traces: The uppermost layer comprises Wi-Fi, RF, and telemetry transmissions intercepted by access points or monitoring systems. Although partial, these transmissions provide autonomous support for UAV activity and time sequencing. This multi-layered architecture helps provide principled evidence fusion while maintaining source-specific reliability differences.

The general layout of the proposed forensic methodology is shown in Fig. 1, which models UAV evidence as a stratified, reliability-weighted system rather than a flat set of artefacts. The figure shows four forensic layers, each with distinct persistence and trust attributes: physical memory, logical logs, companion device artefacts, and wireless traces. These layers have different persistence and trust properties that are downstream of the inference process.

B. Algorithm 1: Physics-Based Flight Path Reconstruction

Reconstruction of flight paths has been a fundamental inference problem in UAV forensics. This work uses a physics-based, vendor-neutral visualization model rather than vendor-specific tools, ensuring reproducibility and cross-platform applicability.

Given sequential GPS coordinates (ϕi,λi), the geodesic distance between consecutive points is computed using the Haversine formula:

$$d = 2r\arcsin\left(\sqrt{\sin^2\left(\frac{\phi_{i+1}-\phi_i}{2}\right)+\cos(\phi_i)\cos(\phi_{i+1})\sin^2\left(\frac{\lambda_{i+1}-\lambda_i}{2}\right)}\right)$$

where,

- R= Earth radius
- ϕ= latitude
- λ= longitude

Velocity and acceleration are subsequently derived as:

$$v_i = \frac{d_i}{\Delta t_i}$$

$$a_i = \frac{v_{i+1}-v_i}{\Delta t_i}$$

where,

- d_i= distance between consecutive trajectory points
- Δt_i= time interval between successive timestamps
- v_i= velocity at the i-th interval
- a_i= acceleration derived from velocity changes

It is a formulation that allows physically implausible maneuvers to be detected, missing segments to be interpolated, and consistency checking across layers of evidence to be performed. Since the model is not based on proprietary log semantics but on the foundations of basic kinematics, it applies to UAV makers and firmware releases. Algorithm 1 presents the trajectory reconstruction.

Algorithm 1: **Trajectory Reconstruction**

GPS coordinates, timestamps

```
Compute
While (consecutive GPS points exist) do
    For (each consecutive GPS point pair) do
    Update
        Compute velocity using distance and time difference
        Compute acceleration from consecutive velocities
            If (physical plausibility condition satisfied) then
                Search
    End         End
    End
```

C. Algorithm 2: Cross-Device Temporal Alignment

Temporal congruence among heterogeneous devices remains a persistent challenge in UAV forensic investigations, particularly when correlating drone telemetry with artefacts from companion mobile devices. To address this issue, a formal synchronization model is adopted to provide structured and defensible temporal alignment.

For any event i logged on the UAV and event j recorded on the companion device, the temporal discrepancy is defined as:

$$\Delta t_{ij} = | t_i^{\text{drone}} - t_j^{\text{mobile}} |$$

To compensate for clock drift, logging latency, acquisition resolution differences, and communication delays, a synchronization tolerance parameter $\varepsilon$is introduced. Clock drift may arise from hardware timing inaccuracies, firmware offsets, or environmental conditions. Although explicit drift estimation is beyond the scope of the present study, the tolerance parameter accommodates moderate drift effects while maintaining practical forensic applicability. Future work may incorporate formal drift estimation techniques to enhance alignment precision.

False temporal alignment may occur when unrelated events fall within the tolerance window. To reduce this risk, temporal correlation is interpreted alongside contextual evidence such as flight telemetry consistency, operator interactions, and communication traces rather than relying solely on timestamp proximity.

Synchronization uncertainty is inherent in multi-device forensic investigations due to logging latency, sampling resolution, and transmission delays. The proposed tolerance-based alignment model provides a transparent mechanism for acknowledging such uncertainty while preserving

interpretability in forensic reporting. Future work will explore probabilistic synchronization models to improve robustness.

Events are considered temporally consistent when:

$$\Delta t_{ij} \leq \varepsilon$$

where,

- t_i^{drone}= UAV timestamp
- t_j^{mobile}= mobile device timestamp
- ε= synchronization tolerance

This parameterized tolerance model replaces ad hoc time matching with a more defensible alignment strategy, enabling investigators to justify synchronization decisions, quantify temporal uncertainty, and explicitly report assumptions in forensic analysis and expert testimony. Consequently, the framework helps strengthen evidentiary linkage between operator actions, device artefacts, and observed UAV behavior while addressing a methodological gap identified in prior UAV forensic research.

V. Forensic Uncertainty and Confidence Modeling

Conclusions in digital forensics that fail to acknowledge uncertainty implicitly exaggerate the strength of the evidence and, in an adversarial context, compromise their own integrity. The issue is intensified in UAV studies, where evidence is found in heterogeneous subsystems that operate under different time, physical, and operational factors. This means that uncertainty modelling and expressing confidence are not supportive additions but fundamental needs of court-facing UAV forensic science.

A. Evidence Reliability Weighting

The evidence sources in the proposed framework do not receive epistemic equivalence. Instead, each layer gets a reliability weight indicating three forensic dimensions: data persistence, user manipulation, and integrity during acquisition. These weights are not chosen unnecessarily, as they are based on the principles of digital forensic reliability established, known to date [25], and reflect empirical practice that has been reported in the literature of UAV and mobile forensics [26], [27].

The weight of physical chip-off memory is the highest because it is non-volatile and not subject to change after the event, though it is acquired invasively. So logical flight logs, which are stored on internal or removable media, are structured data on operational data but can be deleted or overwritten. Artefacts of companion mobile devices are highly detailed contextual links that are mediated by operating system abstractions and application-layer conversions. Wireless traces, although useful in corroboration, are naturally discontinuous and volatile over time [28].

The explicit weighting of these layers formalizes what is often informal in expert reasoning, thereby enhancing transparency, reproducibility, and defensibility. Extracted evidence sources are not equally forensically reliable due to variability in persistence, manipulation vulnerability, and integrity of acquisition across the ecosystem of UAVs. The weights of reliability that are attributed to each evidentiary layer and applied in this study are captured in a table in the form of Table II.

TABLE II. Evidence Reliability and Weight Assignment

Evidence Layer	Forensic Characteristics	Assigned Weight
Physical memory (chip-off)	Non-volatile, low manipulability, high persistence	0.35
Logical logs (SD/internal)	Structured records, moderate volatility	0.25
Companion mobile device	Context-rich, OS-mediated, moderate uncertainty	0.25
Wireless traces	Volatile, fragmentary, corroborative	0.15

The evidentiary weights are based on forensic reliability factors such as data persistence, susceptibility to manipulation, and acquisition integrity. Physical memory artefacts receive higher weights due to their non-volatile nature, while wireless traces are assigned lower weights because of their volatility.

B. Forensic Confidence Index (FCI) – Novel Contribution

To operationalize uncertainty-sensitive inference, the study proposes a Forensic Confidence Index (FCI), a scalar measure intended to convey the cumulative viability of evidentiary support for a particular forensic statement. The FCI is defined as:

$$\text{FCI} = \sum_{i=1}^{n} w_i C_i$$

where,

- w_i= reliability weight of the i-th evidence layer
- C_i= normalized consistency score (0–1)
- n= number of evidence layers

The proposed Forensic Confidence Index represents a methodological framework rather than a fully statistically validated model. Future studies will focus on empirical calibration using larger UAV forensic datasets.

Where w_i represents the reliability weight of the i -th evidence layer, and $C_i$denotes the normalized consistency score (ranging from 0 to 1) reflecting the internal coherence and cross-layer corroboration of that evidence layer.

The thresholds of the interpretation are established to embrace forensic reporting and expert testimony:

FCI ≥ 0.80: Strong forensic support

0.60 ≤ FCI < 0.80: Moderate support with acknowledged uncertainty

FCI < 0.60: Weak or insufficient support

The FCI threshold values are heuristic and intended to support interpretive guidance rather than absolute probabilistic certainty. Further empirical validation will refine these thresholds.

More importantly, the FCI does not purport to know. Instead, it offers a clear, quantitative expression of confidence that can be scrutinized, challenged, and contextualized - a requirement also emphasized in digital forensic admissibility criteria. An overview of UAV forensic research published in 2020-2025 shows that no similar tool exists to articulate attribution confidence using a combination of evidence types, making this contribution both innovative and needed.

The mathematical calculations behind the Forensic Confidence Index are illustrated in Fig. 2, which shows how heterogeneous evidence inputs are transformed by weighting reliability and scoring consistency, and then aggregated into a single confidence-constrained measure.

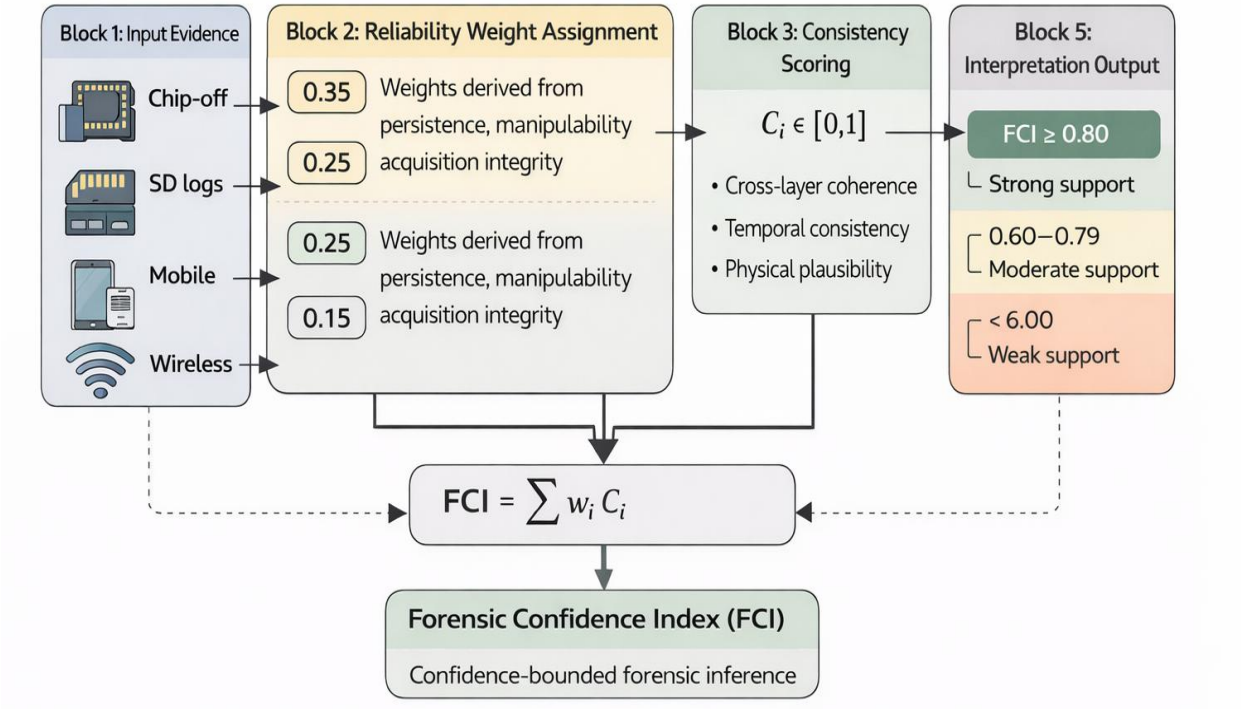


Fig. 2. Forensic Confidence Index computation flow.

VI. Case Study: Yuneec Typhoon Q500 4K (Validation Only)

Forensic science case studies are confirmatory, not creative: they show that a proposed set of concepts can work within real-world constraints. Although the validation focuses on a single UAV platform, this case study serves as a proof-of-concept for the proposed framework under realistic forensic conditions. In emerging forensic domains, initial validation often relies on representative case studies before broader multi-platform evaluation becomes feasible. Accordingly, the Yuneec Typhoon Q500 4K is used as a validation platform for the multi-layer forensic framework, algorithms, and confidence modelling methodology presented earlier.

The inspected UAV was recovered after it was suspected of being used in an unauthorized surveillance operation. The acquisition process was forensically sound, including invasive chip-off acquisition of onboard flash memory, logical acquisition of SD card flight logs, forensic imaging of the companion device, which was also based on Android, and preservation of residual wireless communication artefacts on the local access point. Write-blocked workflows were used for all acquisitions, and the integrity of the evidence was verified through cryptographic hashing of the message.

The artefacts across the four layers of evidence differed in persistence, granularity, and internal coherence. Physical memory provided low-level configuration residues and partial flight buffers that were difficult for the user to erase. Logical logs provided formatted GPS telemetry and timed flight events. Artifacts from companion devices provided application-level sync points and operator touchpoint metadata, whereas wireless traces provided corroborative time anchors at takeoff and landing. More importantly, none of the layers was considered dispositive; the value of the evidence appeared only in the form of systematic composition.

The reconstructed flight path algorithm produced a continuous three-dimensional path within the physical limits of the Q500 platform, and discrepancies between the drone and mobile timestamps were addressed by time-matching within the specified synchronization error. Using the Forensic Confidence Index (FCI), a composite score indicating strong evidentiary support but not absolute confidence was obtained.

The case study focuses on implementing the proposed framework on real forensic artefacts recovered from a Yuneec Typhoon Q500 4K UAV. Table III provides a summary of all recovered artefacts, their provenience, and the confidence measures.

TABLE III. EXTRACTED FORENSIC ARTEFACTS SUMMARY

Source	Artefacts Identified	Count	Confidence Contribution
Chip-off Memory	Configuration remnants, partial buffers	0.35	High
SD card logs	GPS points, timestamps, flight states	0.25	Moderate - High
Companion mobile device	App logs, user actions, sync markers	0.25	Moderate
Wireless traces	Connection timestamps, session IDs	0.15	Low - Moderate

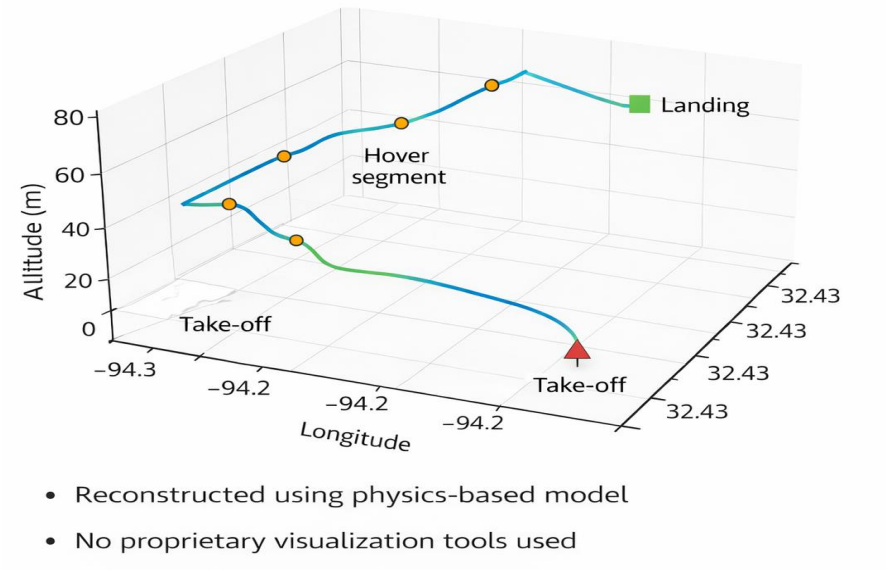


Fig. 3. 3D Flight path reconstruction of yuneec typhoon Q500 4K.

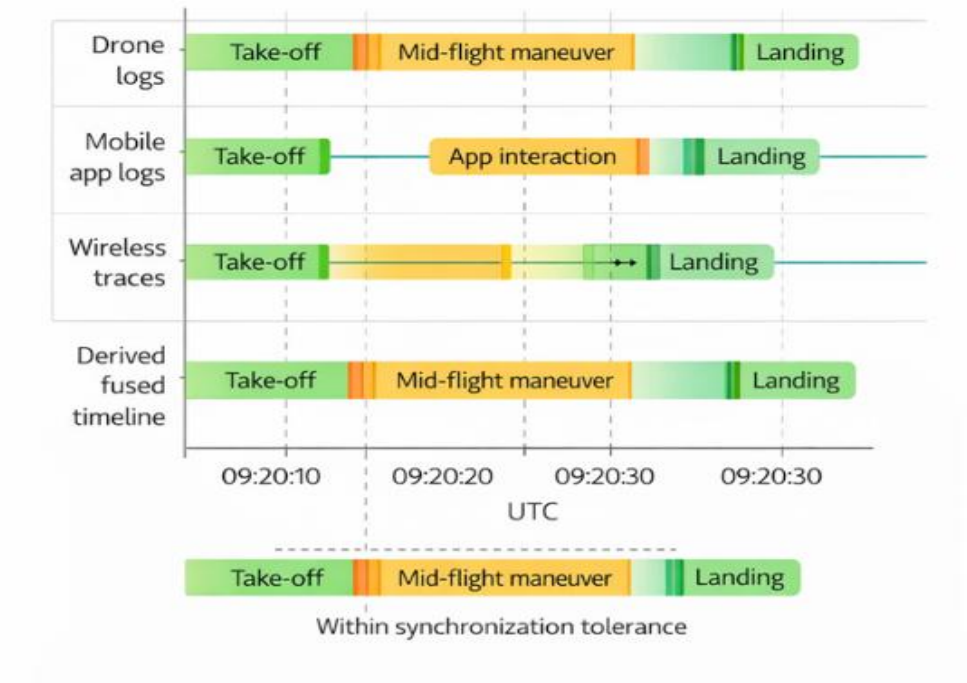


Fig. 4. Cross-device evidence correlation timeline.

The recovered flight path, based on telemetry from the SD card, is plotted in Fig. 3 and shows a physically possible three-dimensional flight path that aligns with the reported take-off, maneuvering, and landing events.

Temporal correspondence between drone logs, mobile application artefacts, and wireless traces is shown in Fig. 4, which presents a coherent timeline extracted within specified synchronization margins.

VII. Discussion: Comparative and Theoretical Implications

The significance of the present study's findings extends beyond artefact retrieval to their implications for the conceptualization, integration, and interpretation of UAV forensic evidence. Compared with previously reported UAV forensic case studies involving DJI and Yuneec platforms, which largely emphasize artefact extraction and platform-specific analysis, the proposed framework explicitly integrates multi-layer evidence fusion, uncertainty modelling, and confidence quantification. This methodological shift enables more transparent forensic inference while maintaining compatibility with established UAV forensic workflows. Earlier studies have demonstrated successful access to forensic artefacts; however, the evidentiary strength often remained implicit and dependent on investigator interpretation rather than structured inference models.

The proposed framework advances UAV forensic investigation along three complementary dimensions. First, physics-based trajectory validation replaces purely descriptive reconstruction, reducing reliance on proprietary visualization tools and improving reproducibility. Second, structured cross-device temporal alignment provides a systematic mechanism for addressing timestamp inconsistencies frequently encountered in forensic investigations and litigation contexts. Third, the explicit incorporation of uncertainty management through reliability weighting and the Forensic Confidence Index (FCI) introduces a quantified representation of evidentiary confidence that has been largely absent from prior UAV forensic literature.

From an interpretive standpoint, the FCI results illustrate how heterogeneous evidence sources contribute differently to overall forensic confidence. More persistent artefacts, such as onboard memory and structured flight logs, typically provide stronger evidentiary stability, whereas volatile wireless traces act primarily as corroborative indicators. This structured integration enhances evidentiary coherence compared with single-source forensic approaches while maintaining transparency regarding uncertainty.

Theoretically, these findings reinforce the applicability of the digital Locard's exchange principle within cyber-physical forensic environments and clarify the distinction between evidentiary certainty and inferential confidence. Rather than asserting deterministic conclusions, the framework promotes calibrated confidence reporting, an approach increasingly recognized as essential for scientific credibility and legal defensibility in forensic science.

More broadly, the framework suggests a transition in UAV forensic practice from artefact-oriented reporting towards inference-oriented reasoning. Such a transition supports improved transparency, reproducibility, and judicial admissibility while aligning UAV forensic methodologies with evolving forensic science standards. Sensitivity considerations indicate that moderate variations in evidentiary weights do not substantially alter overall confidence classification, suggesting preliminary robustness of the framework. Nevertheless, comprehensive quantitative sensitivity analysis and multi-platform empirical validation remain important directions for future work.

VIII. Generalization, Admissibility, and Practical Deployment

To have long-term value, a UAV forensic framework should not be confined to a single platform, while remaining in line with operational and legal limitations. Even though its validation is performed with the Yuneec Typhoon Q500 4K, the proposed framework is deliberately manufacturer-agnostic. The fact that its use of physics-based trajectory reconstruction, abstracted temporal alignment, and evidence-layer weighting is directly applicable to the ecosystems of DJI, Parrot, and Autel UAVs, despite their proprietary log formats or telemetry encodings, is possible [29]. More to the point, the framework is independent of tools. The algorithms are applied to normalized artefact representations rather than vendor-specific parsers, allowing the investigator to combine commercial, open-source, or lab-written tools without modifying the inferential logic. This design option directly reinforces reproducibility and overcomes tool-bias challenges, which are increasingly contentious in forensic admissibility trials.

Legally, the framework corresponds to the principles of reliability used in Daubert-style tests: methods are testable, sources of error are recognized through uncertainty modelling, and inferential steps are transparent and reproducible. Chain-of-custody integrity is maintained by supporting standard forensic acquisition processes and by hash-verified handling of evidence. Conclusions derived from the framework can therefore be expressed as expert opinions with confidence limits, rather than unprovable claims.

IX. Limitations and Future Directions

The strengths notwithstanding, the framework suggested has some limitations. Streaming encrypted telemetry and firmware seals might curtail access to artefacts on newer UAV systems, diminishing the degree of multi-layer evidence integration. Also, the current implementation of wireless trace analysis relies on residual or post-event data; real-time software-defined radio (SDR) capture would improve time resolution but raise logistical and legal issues. The current FCI formulation relies on theoretically grounded weight assignments and requires broader empirical validation across diverse UAV forensic scenarios. Future work will extend validation across multiple UAV platforms, including DJI, Parrot, and Autel ecosystems, to evaluate generalizability and robustness under diverse operational conditions.

To address the problem of time-dependent updating of confidence scores as new evidence is found, future work may generalize the Forensic Confidence Index using probabilistic or Bayesian machine learning models that dynamically update confidence scores as new evidence becomes available. These extensions, though, should be rigorously scrutinized to maintain their interpretability and to remain non-opaque in their explanations before the court.

X. Conclusion

This study advances UAV forensic science by shifting the focus from artefact-centric examination toward confidence-aware forensic inference. A theoretically grounded multi-layer framework was proposed that integrates physics-based trajectory reconstruction, structured cross-device temporal

alignment, and explicit uncertainty modelling. The introduction of the Forensic Confidence Index (FCI) provides a systematic mechanism for expressing evidentiary strength without overstating certainty, thereby supporting transparent and defensible forensic reporting.

The findings demonstrate that multi-layer evidence integration improves evidentiary coherence and interpretability compared with purely descriptive artefact analysis. This approach has practical implications for enhancing transparency, reproducibility, and legal defensibility in UAV forensic investigations while aligning forensic practice with emerging scientific and judicial expectations.

However, validation in the present study is based on a single UAV platform (Yuneec Typhoon Q500 4K), and broader empirical evaluation across diverse UAV ecosystems remains necessary to strengthen generalizability. Despite this limitation, the framework establishes a methodological foundation for structured confidence-aware inference in cyber-physical forensic environments.

Future work will focus on multi-platform empirical validation, refinement of reliability weight assignments, and quantitative sensitivity analysis to further strengthen confidence assessment in UAV forensic investigations.

REFERENCES

[1] D. Y. Kao, M. C. Chen, W. Y. Wu, J. S. Lin, C. H. Chen, and F. Tsai, "Drone forensic investigation: DJI spark drone as a case study," Procedia Comput. Sci., vol. 159, pp. 1890–1899, 2019, doi: 10.1016/J.PROCS.2019.09.361.

[2] "Unmanned Aerial Vehicle Forensic Investigation Process: Dji Phantom 3 Drone As A Case Study | Request PDF." Accessed: Jan. 24, 2026. [Online]. Available: https://www.researchgate.net/publication/324745129_Unmanned_Aerial_Vehicle_Forensic_Investigation_Process_Dji_Phantom_3_Drone_As_A_Case_Study

[3] G. De La Torre, P. Rad, and K. K. R. Choo, "Driverless vehicle security: Challenges and future research opportunities," Future Generation Computer Systems, vol. 108, pp. 1092–1111, Jul. 2020, doi: 10.1016/J.FUTURE.2017.12.041.

[4] D. Kovar, "Who We Are."

[5] "Gartner Says Almost 3 Million Personal and Commercial Drones Will Be Shipped in 2017." Accessed: Jan. 24, 2026. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2017-02-09-gartner-says-almost-3-million-personal-and-commercial-drones-will-be-shipped-in-2017

[6] K. Chávez and O. Swed, "The proliferation of drones to violent nonstate actors," Defence Studies, vol. 21, no. 1, pp. 1–24, Jan. 2021, doi: 10.1080/14702436.2020.1848426;WGROUP:STRING:PUBLICATION.

[7] N. Riya Rajendran, G. Hemi, M. Dipak Kumar, P. Behumiraj, and P. Ankita, "Drone Forensics," Advanced Techniques and Applications of Cybersecurity and Forensics, pp. 99–123, Jul. 2024, doi: 10.1201/9781003386926-6.

[8] G. Horsman, "Unmanned aerial vehicles: A preliminary analysis of forensic challenges," Digit. Investig., vol. 16, pp. 1–11, Mar. 2016, doi: 10.1016/J.DIIN.2015.11.002.

[9] Z. Zhao, Y. Wang, and G. Liao, "Digital Forensic Research for Analyzing Drone and Mobile Device: Focusing on DJI Mavic 2 Pro," Drones 2024, Vol. 8, vol. 8, no. 7, Jun. 2024, doi: 10.3390/drones8070281.

[10] S. Lee, H. Seo, and D. Kim, "Digital Forensic Research for Analyzing Drone Pilot: Focusing on DJI Remote Controller," Sensors (Basel)., vol. 23, no. 21, p. 8934, Nov. 2023, doi: 10.3390/s23218934.

[11] F. Alotaibi, A. Al-Dhaqm, and Y. D. Al-Otaibi, "A Conceptual Digital Forensic Investigation Model Applicable to the Drone Forensics Field," Engineering, Technology & Applied Science Research, vol. 13, no. 5, pp. 11608–11615, Oct. 2023, doi: 10.48084/etasr.6195.

[12] H. Studiawan, G. Grispos, and K. K. R. Choo, "Unmanned Aerial Vehicle (UAV) Forensics: The Good, The Bad, and the Unaddressed," Comput. Secur., vol. 132, no. 13, p. 103340, Sep. 2023, doi: 10.1016/j.cose.2023.103340.

[13] M. Y. Halim and A. Luthfi, "Digital Forensic Analysis of UAV Flight Data Using Static and Dynamic Methods in Coal Mining Area," Journal of Information Systems and Informatics, vol. 7, no. 2, pp. 1042–1060, Jun. 2025, doi: 10.51519/journalisi.v7i2.1061.

[14] R. Tombari Sibe and D. Bekom, "Digital Forensic Investigation of an Unmanned Aerial Vehicle (UAV): A Technical Case Study of a DJI Phantom III Professional Drone," Journal of Cybersecurity and Information Management, vol. 15, no. 1, pp. 197–210, 2025, doi: 10.54216/JCIM.150115.

[15] F. E. Salamh, U. Karabiyik, M. K. Rogers, and E. T. Matson, "A comparative uav forensic analysis: Static and live digital evidence traceability challenges," Drones, vol. 5, no. 2, Jun. 2021, doi: 10.3390/drones5020042.

[16] A. Almusayli, T. Zia, and E. U. H. Qazi, "Drone Forensics: An Innovative Approach to the Forensic Investigation of Drone Accidents Based on Digital Twin Technology," Technologies 2024, Vol. 12, vol. 12, no. 1, Jan. 2024, doi: 10.3390/technologies12010011.

[17] S. A. M. Alhasan, S. H. Othman, and A. Al-Dhaqm, "Metamodeling-Based Drone Forensics Investigation: A Systematic Literature Review," International Journal of Safety and Security Engineering, vol. 15, no. 1, pp. 1–12, Jan. 2025, doi: 10.18280/ijsse.150101.

[18] U. M. Mohammed et al., "Cyber threat in drone systems: bridging real-time security, legal admissibility, and digital forensic solution readiness," Frontiers in Communications and Networks, vol. 6, p. 1661928, Sep. 2025, doi: 10.3389/frcmn.2025.1661928.

[19] R. D. Thantilage, G. Buttner, and R. Genoe, "Drone forensics in law enforcement: Assessing utilisation, challenges, and emerging necessities," Forensic Science International: Digital Investigation, vol. 55, no. 13, p. 302003, Dec. 2025, doi: 10.1016/j.fsidi.2025.302003.

[20] S. Viswanathan and Z. Baig, "Digital Forensics for Drones: A Study of Tools and Techniques," Communications in Computer and Information Science, vol. 1338, pp. 29–41, 2020, doi: 10.1007/978-981-33-4706-9_3.

[21] Ankit, T. Kumar, and H. K. Singhal, "Forensics Case studies – Extraction of deleted and live data from DJI Agras MG 1s Drone Through Chip-off, Internal SD Card and Controller Chip-off Extraction.," International Journal of Innovative Research in Technology, vol. 11, no. 8, pp. 1628–1636, 2025, Accessed: Feb. 13, 2026. [Online]. Available: https://ijirt.org/article?manuscript=172020

[22] K. Nieszporek, J. Bernacki, K. A. P. Costa, Q. Ke, and W. Wei, "UAV Forensics: Transformer-Based Identification of Mobile Devices for Drone Piloting," Lecture Notes in Computer Science, vol. 15949 LNCS, pp. 355–363, 2026, doi: 10.1007/978-3-032-03708-4_29.

[23] E. Mantas and C. Patsakis, "Who watches the new watchmen? The challenges for drone digital forensics investigations," Array, vol. 14, p. 100135, Jul. 2022, doi: 10.1016/J.ARRAY.2022.100135.

[24] R. Tombari Sibe and D. Bekom, "Digital Forensic Investigation of an Unmanned Aerial Vehicle (UAV): A Technical Case Study of a DJI Phantom III Professional Drone," Journal of Cybersecurity and Information Management, vol. 15, no. 1, pp. 197–210, 2025, doi: 10.54216/JCIM.150115.

[25] F. Alotaibi, A. Al-Dhaqm, and Y. D. Al-Otaibi, "A Conceptual Digital Forensic Investigation Model Applicable to the Drone Forensics Field," Engineering, Technology & Applied Science Research, vol. 13, no. 5, pp. 11608–11615, Oct. 2023, doi: 10.48084/ETASR.6195.

[26] H. Studiawan, G. Grispos, and K. K. R. Choo, "Unmanned Aerial Vehicle (UAV) Forensics: The Good, The Bad, and the Unaddressed," Comput. Secur., vol. 132, Sep. 2023, doi: 10.1016/J.COSE.2023.103340.

[27] A. Taylor, "A Digital Forensics Case Study of the DJI Mini 3 Pro and DJI RC," Sep. 2023, Accessed: Jan. 24, 2026. [Online]. Available: https://arxiv.org/pdf/2309.10487

[28] S. Lee, H. Seo, and D. Kim, "Digital Forensic Research for Analyzing Drone Pilot: Focusing on DJI Remote Controller," Sensors 2023, Vol. 23, vol. 23, no. 21, Nov. 2023, doi: 10.3390/S23218934.

[29] D. Lee and W. Kang, "Drone forensics redefined: Integrating live, digital, and non-digital evidence acquisition systems," Forensic Sci. Int., vol. 11, p. 100635, Dec. 2025, doi: 10.1016/J.FSISYN.2025.100635.

Real-Time Data-Driven Decision Support in Retail: A Hybrid GraphSAGE+XGBoost Model for Predicting Reorder Behavior and Unraveling Consumer Communities

Balayet Hossain[1], Md Deluar Hossen[2], Md Nuruzzaman Pranto[3], Belal Hossain[4],
Sabrina Shamim Moushi[5], Nusrat Ameri[6], Khandakar Rabbi Ahmed[7]*
Department of Business Administration & Management, International American University, Los Angeles, United States[1]
Department of MBA in Business Analytics, International American University, Los Angeles, United States[2]
Department of Management Information Systems, International American University, Los Angeles, United States[3, 4, 5]
Physics Discipline, Khulna University, Bangladesh[6]
Miyan Research Institute, International University of Business Agriculture and Technology, Bangladesh[7]

***Abstract*—The rising demand for real-time, data-driven decision support in retail platforms has underscored the need for intelligent systems capable of modeling both behavioral sequences and product relationships. This study introduces a hybrid architecture for real-time decision support in retailing by coupling graph-based learning with conventional machine learning methods. Based on Instacart 2017 data, it constructs a heterogeneous user-product graph and utilizes GraphSAGE to obtain relational embeddings. This combination of embeddings and domain-specific features is then fed into an XGBoost classifier to predict reorder behavior. Empirical findings show that the proposed GraphSAGE+XGBoost model outperforms conventional baselines, including the sole XGBoost, Multilayer Perceptron (MLP), and Long Short-Term Memory (LSTM) models. In particular, the hybrid model outperformed all baselines across all metrics, achieving a precision of 0.82, a recall of 0.78, an F1-score of 0.76, and a mean Average Precision (mAP) of 0.75. Furthermore, within the co-purchase network, product-level community identification identified significant clusters (such as breakfast staples, health-conscious products, and impulsive snacking) that provided insights into customer demographics and marketing potential. The experimental analysis comparing the proposed GraphSAGE+XGBoost with baseline models, including LSTM, XGBoost, and MLP, demonstrates that the proposed hybrid model outperforms in terms of modeling accuracy, Precision, and generalizability. The system is optimized for real-time inference and can operate in a dynamic commercial landscape, unraveling complex co-purchase behavior and hidden consumer communities.**

***Keywords*—Real-time business intelligence; graph analytics; machine learning; GraphSAGE; XGBoost; retail analytics; recommendation systems; decision support; instacart dataset; hybrid model**

I. Introduction

In the current digital economy, post-hoc analysis and static reporting are no longer the only options available to business intelligence systems [1]. As online platforms and transaction-heavy ecosystems continue to grow rapidly, businesses today require real-time decision-support systems that can extract meaningful insights from continuous streams of client data [2]. In retail and e-commerce settings, in particular, precise, fast, and scalable intelligence solutions are crucial for anticipating customer demands, personalizing recommendations, and optimizing inventory and marketing campaigns [3].

Retail transaction logs, such as those on the Instacart platform, reveal a complex interplay between behavioral, category, and sequential information [4]. Along with what clients buy, these statistics also record when, how often, and in what circumstances. Models that can go beyond conventional feature engineering are needed to utilize such data for predictive purposes.

These models must be able to incorporate both temporal patterns and the structural links between users and items. Existing models commonly used in this domain—such as decision tree ensembles, deep feedforward networks, or sequential architectures like LSTM—tend to address only one aspect of the data. While XGBoost and MLP models excel at handling structured features, they fall short at capturing relational dependencies among entities, such as users who purchase similar items or products that are frequently bought together [5]. Conversely, while LSTM models can model ordered sequences, they often overlook the underlying graph structure of interactions and may struggle when sequences are sparse or highly variable [6].

Furthermore, many real-time systems cannot generalize to unseen products or users, and they offer limited interpretability—both of which are critical in commercial applications. The objective of this study is to bridge the gap between graph-based representation learning and high-performance predictive modelling for real-time business intelligence. Graph-based methods, such as GraphSAGE, can effectively model user-product interactions as nodes and edges in a bipartite graph. At the same time, classifiers like XGBoost remain essential due to their interpretability, scalability, and ability to incorporate domain-specific features. By combining these two strengths—relational learning via graph embeddings

*Corresponding author.

and structured inference via gradient-boosted trees—our proposed model aims to deliver a more accurate and generalizable decision-making system suitable for large-scale retail platforms. This study presents a hybrid framework for real-time reorder prediction using the Instacart 2017 dataset. The main contributions are summarized below:

- We created a heterogeneous graph from transactional data, defining users and products along with temporal and categorical features.
- We presented a hybrid model that combines GraphSAGE embeddings with XGBoost classification, incorporating the strengths of structural learning and boosted decision trees.
- We compared our proposed framework against established baselines, including LSTM, XGBoost, and MLP, demonstrating its superior predictive accuracy and generalization.
- We developed a design optimized for real-time inference, supporting decision support applications in dynamic commercial environments.

II. Literature Review

Recently, the application of Artificial Intelligence (AI) and Machine Learning (ML) in retail analytics and business intelligence has become quite expensive. All of these issues on how these technologies help in decision-making, streamlining operations, and how they can help in the unraveling of latent consumer behaviors have been the subject of numerous studies. This part critically examines advancements over the past few years and the practical applications of AI-based methods in business analytics, co-purchasing modeling, recommendation systems, and customer-behavior prediction across academic and industrial settings. Badmus et al. [7] analyzed the applications of AI and ML in business analytics and their contribution to predictive decision-making, business operations efficiency, and real-time information. SAP case studies showed that AI can boost conversion rates and optimize resource allocation across industries and businesses.

Chintala and Thiyagarajan [8] emphasized that AI-enhanced Business Intelligence (BI) revolutionizes classical business analysis by providing real-time insights, predictive analysis, and automation. They focused on the importance of using technologies such as machine learning and NLP to improve decision-making, streamline processes, and reveal previously inaccessible patterns, backed by real-world experience. Reddy et al. [9] noted that Artificial Intelligence enhances Business Intelligence by improving efficiency, automation, decision-making, and problem-solving. A study conducted by 204 professionals highlights the growing importance of AI for real-time data processing and strategic business development in the context of big data. Niu et al. [10] proposed the ODM-BDA framework, which leverages backtracking and steep optimization methods to enhance business intelligence and decision-making. According to their simulation results, organizations are more effective, better manage risks, and make more accurate data-driven decisions. Hanumanth [11] introduced a bright idea of an intelligent e-shopping system, SmartCart, which allows users to search, in a unified place, across different e-commerce stores and compare them in real-time, providing personalized product suggestions with the support of learning user preferences, to overcome most of the weaknesses of existing online shopping systems. Farheen and Dharani [12] constructed a retail analytics pipeline to reveal association rules and customer online buying trends employing the Apriori algorithm and cloud-based tools. They used their system, augmented with GPT-3.5 Turbo, which reduced error rates and provided useful insights into customer feedback for marketing, bundling, and inventory management.

Meftah, Ounacer, and Azzouazi [13] harmonized network science and AI, incorporating a centrality measure from a co-purchase network into ML models to improve customer purchase prediction. A Random Forest model they developed had an AUC of 0.82, indicating the usefulness of network-based characteristics in retail analytics. Ariannezhad et al. introduced ReCANet, a specialized neural network for next-basket recommendation that explicitly models repeat consumption behavior in grocery.

Ariannezhad et al. [14] introduced ReCANet, a specialized neural network for next-basket recommendation that explicitly models repeat consumption behavior in grocery. Their methodology significantly surpasses state-of-the-art models and demonstrates that repeat items, which comprise only ∼1% of the catalog, account for more than 54% of the recommendation performance. Taken together, the literature explored the ushering effects of AI and ML in retail and e-commerce. These studies offer a reasonable basis for developing innovative, data-driven systems, including leveraging business intelligence and personalization, improving prediction accuracy with network-based features, and modeling repeat consumption.

Nevertheless, though a range of approaches indicate that they are effective in particular operations, like association rule mining, next-basket processing, or user preference (learning), there is still a great chance to clarify these points of view. The study will consequently follow up on these findings by combining graph-based community detection and centrality measures with hybrid machine learning methods to present an efficient approach to constructing recommendations that meet the requirements of retail analytics.

III. Methods and Materials

This section outlines the overall methodology of our study, encompassing the dataset description, preprocessing steps, and the predictive framework adopted. The complete workflow is illustrated in Fig. 1, providing a visual summary of the key components and processes.

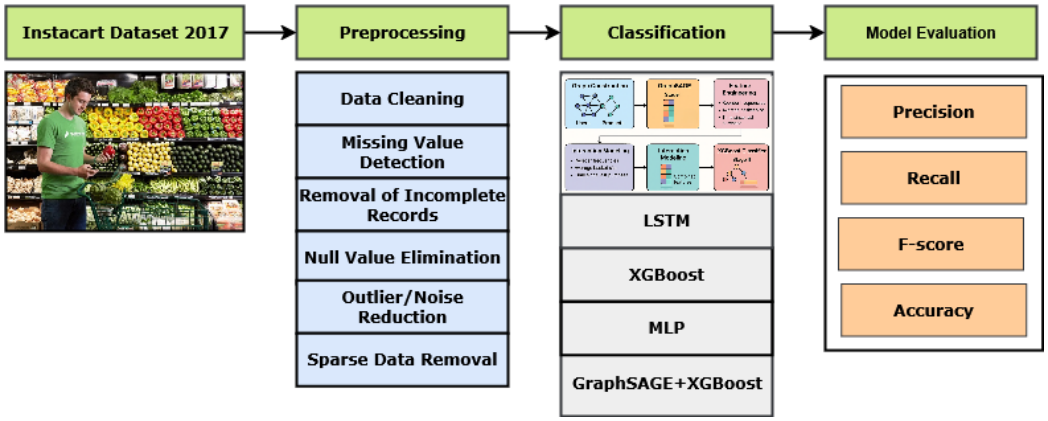


Fig. 1. Overall workflow of the research.

A. Dataset Description

In this work, we utilized the publicly available Instacart Market Basket Analysis 2017 dataset, which provides detailed transactional records from an online grocery delivery platform comprising over 3 million orders placed by more than 200,000 unique users [15]. Each user has between 4 and 100 sequential orders, capturing their longitudinal purchasing behaviour. Each order record includes both the list of purchased products and contextual attributes, such as the day of the week, the hour of the day, and the number of days since the previous order, enabling the modelling of temporal purchasing patterns. Fig. 2 visualizes the number of customers, orders, aisles, and products.

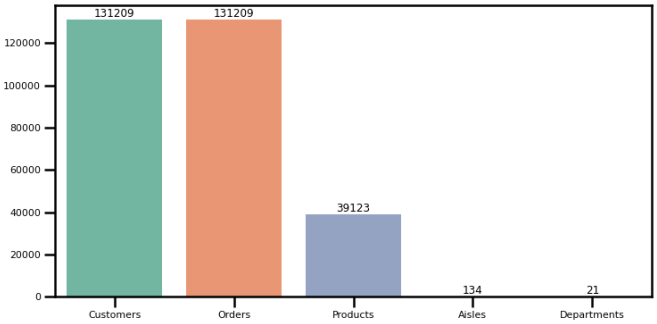


Fig. 2. Structure of the dataset.

The dataset is organized into six structured tables: aisles, departments, products, orders, order_products_prior, and order_products_train. The aisles provide intermediate-level product categories, each with a corresponding aisle identifier, while the departments define higher-level department groupings. The products contain product metadata, including product names, aisle IDs, and department IDs. The order store maintains detailed user order histories, including order identifiers, user identifiers, order sequence numbers, order timing information (day of the week and hour), and the number of days since the previous order was placed. The order products prior and order products train describe the contents of each order, capturing which products were added, the order in which they were added to the cart, and whether the same user reordered the product.

B. Data Preprocessing

Following the first data integration phase, a structured data preprocessing strategy was involved to provide analytical consistency and model readiness. To find missing attribute values and incomplete records, the dataset was thoroughly examined. To maintain the integrity of subsequent studies, these null items were systematically excluded because they accounted for a small portion of the dataset and appeared to be randomly distributed. The dataset's representativeness and statistical balance were preserved, while noise was reduced by removing sparse and uninformative data. As a result, the final dataset was well-structured and free from inconsistencies, making it suitable for robust training of machine learning models and the deployment of graph-based decision support systems in the context of real-time business intelligence. Fig. 3 shows the frequency of days of the week vs. hours of the day.

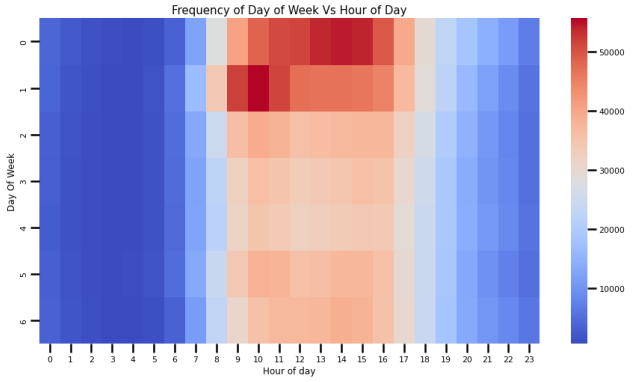


Fig. 3. Frequency of day of week vs hour of day.

C. Baseline Model

Three baseline models were selected for comparison, each representing a distinct modelling approach: sequential, ensemble-based, and feedforward.

1) LSTM: The Long Short-Term Memory (LSTM) model is developed to capture temporal dependencies across ordered user interactions, which possesses information across time steps and generates predictions using the final hidden state [16]:

$$\hat{y} = \sigma(w^T h_T + b) \quad (1)$$

Here, h_T indicates the hidden state at the last time step, and σ is the sigmoid function.

2) XGBoost: XGBoost is a gradient-boosted tree model that creates an ensemble of regression trees in sequence [17], and each tree attempts to minimize the residual error of the previous trees:

$$\hat{y} = \sum_{t=1}^{T} f_t(x) \quad (2)$$

Each function f_t denotes a regression tree, and x is the input feature vector.

3) MLP: The Multilayer Perceptron (MLP) consists of fully connected layers with nonlinear activations, which transform input features through successive linear combinations and nonlinearities [18]:

$$\hat{y} = \sigma(w_T \phi(W_x + b) + b') \quad (3)$$

where ϕ is a ReLU activation involved in the hidden layer output, and σ is a sigmoid function at the output layer.

D. Proposed Model: (GraphSAGE+XGBoost) Model

A hybrid framework designed for effective and timely decision-making in retail analytics, which combines graph-based representation learning with a gradient-boosted classification model to leverage relational patterns in user-product interactions alongside engineered behavioural features. The framework proceeds in two stages, as depicted in Fig. 4:

Stage I: Representation learning on user-product graphs employing GraphSAGE.

Stage II: Prediction through a gradient-boosted decision tree classifier (XGBoost).

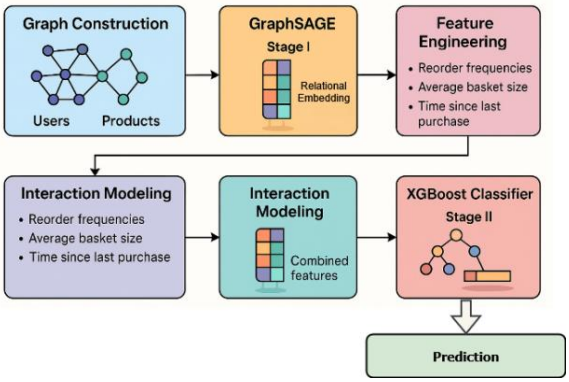


Fig. 4. Architecture of the proposed model.

1) Graph construction: The Instacart transaction data is modeled as a bipartite graph:

$$mathcalG = (\mathcal{V}, \mathcal{E}) \quad (4)$$

where nodes define users $\mathcal{U}$ and products $\mathcal{P}$, so that $\mathcal{V} = \mathcal{U} \cup \mathcal{P}$. An edge $\mathcal{U} \cup \mathcal{P}. \in \mathcal{E}$ exists if user $\mathcal{U}$ purchased product $\mathcal{P}$.

Each node *v* is defined by a feature vector $x_v \in \mathbb{R}^d$ Product nodes encode department and aisle identifiers, reorder frequency, and category popularity. User nodes include metrics such as average basket size, purchase timing, and intervals between orders. This formulation captures both structural relationships and temporal behavior in customer transactions.

2) Representation learning via GraphSAGE: Node embeddings are learned through an inductive aggregation mechanism, where each node updates its representation by integrating its features with those of its neighbors. Formally, for each node *v* at layer *k*:

$$h_v^{(k)} = \sigma(W^{(k)}.AGG^{(k)}\left(\left\{h_v^{(k-1)}\right\} \cup \left\{h_v^{(k-1)} : u \epsilon \mathcal{N}(v)\right\}\right) \quad (5)$$

where, $h_v^{(0)} = x_v$, $AGG^{(k)}$ is the aggregation function (mean aggregator), $W^{(k)}$ is a trainable weight matrix, and σ indicates a nonlinear activation function (ReLU).

The final embedding $z_v = h_v^{(2)}$ is acquired after two such layers, with a hidden dimension of 128.

3) Interaction modeling and feature fusion: To evaluate the potential of a user reordering product p, their embeddings are combined as:

$$z_{(u,p)} = \left[z_u \,||z_p||z_u \Theta z_p\right] \quad (6)$$

where, ‖ signifies concatenation and ⊙ represents elementwise multiplication. This joint vector captures both individual characteristics and interaction effects.

This representation is further enhanced by domain-informed features derived from transaction history, including reorder frequencies, average basket size, time since last purchase, product reorder ratios, and department popularity.

4) Prediction with gradient-boosted trees: The combined feature vector is input to a gradient-boosted decision tree model (XGBoost), which produces an additive sequence of regression trees:

$$\hat{y}_{(u,p)} = \sum_{t=1}^{T} f_t(z_{(u,p)}) \quad (7)$$

with f_t defining individual regression trees. The training objective includes logistic loss connected with regularization to control model complexity:

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(z_{(u,p)})\right) + \Omega(f_t) \quad (8)$$

where,

$$\Omega(f_t) = \Upsilon T + \frac{1}{2}\lambda \sum_j \omega_j^2 \quad (9)$$

Here, ω_j is the score of leaf j in tree f_t. The resulting prediction $\hat{y}_{(u,p)}$ is interpreted as the probability of a reorder event.

IV. Result and Discussion

In this section, a detailed assessment of the chosen GraphSAGE+XGBoost combination model for real-time retail analytics, using the Instacart 2017 data, is presented. This will aim to evaluate how the model can be used to predict user-product reorder behavior by capturing both the time-purchase pattern and the structural relationships in user-product interactions. This section starts with the predictive evaluation of the hybrid model in comparison with three proper baselines: LSTM, MLP, and XGBoost, guided using typical recommendation metrics. Afterwards, we investigate the training and validation behavior of our proposed model to evaluate its learning dynamics and generalization ability. Lastly, the co-purchase surface of products is investigated in this section, using community detection to reveal latent consumer groups and behavioral patterns.

A. Performance Evaluation of Predictive Models

Table I presents a detailed comparison of four predictive models used in next-basket recommendation or co-purchase prediction tasks. The evaluation criteria comprise Precision@10, Recall@10, F1-score, and Mean Average Precision (mAP), all of which are standard evaluation measures in recommendation system evaluation that capture both ranking satisfiability and relevance of prediction. LSTM achieved Precision@10 of 0.76 and a recall@10 of 0.72, indicating moderate performance in predicting the sequential dependencies of purchase behavior. Nevertheless, its F1-score (0.70) and mAP (0.69) indicate that it is not the best at ranking the most relevant items at the top in every case. MLP, a feedforward neural network, scores higher than LSTM with a Precision@10 of 0.77 and a Recall@10 of 0.73, which results in an F1-score of 0.71. XGBoost, which is a gradient-boosted decision tree model, performs better than LSTM and MLP with high scores on all the scoring criteria, and particularly on Precision (0.79) and Recall (0.74), noting the mode of capturing nonlinear association with tabular data, which is one of the additional advantages of the model.

TABLE I. Performance Comparison of Different Models

Model	Precision@10	Recall@10	F1-score	mAP
LSTM	0.76	0.72	0.70	0.69
XGBoost	0.79	0.74	0.72	0.71
MLP	0.77	0.73	0.71	0.70
GraphSAGE+XGBoost (Proposed)	0.82	0.78	0.76	0.75

The GraphSAGE+XGBoost model is the proposed hybrid model that shows the overall performance. Using the graph structure of the purchase network and GraphSAGE embeddings, this model input is fed into the XGBoost classifier. The performance gains are dramatic: Precision@10 of 0.82, Recall@10 of 0.78, F1-score of 0.76, and mAP of 0.75. These measures not only improve the accuracy of recommendations but also enhance the ability to rank products as pertinent. The addition of a strong gradient-boosting classifier to the graph context has significantly improved the comprehensibility of basic user-product associations, yielding much more personalized and meaningful recommendations. To better represent this performance comparison, Fig. 5 provides a clearer view of the benefits of the suggested method.

B. Training and Validation Performance of the Proposed GraphSAGE+XGBoost Model

To determine the learning pattern and generalization ability of the proposed hybrid model (GraphSAGE+XGBoost), training and validation metrics were monitored, with a maximum of 15 training epochs. These are Precision and loss, as depicted in Fig. 6 and Fig. 7, respectively.

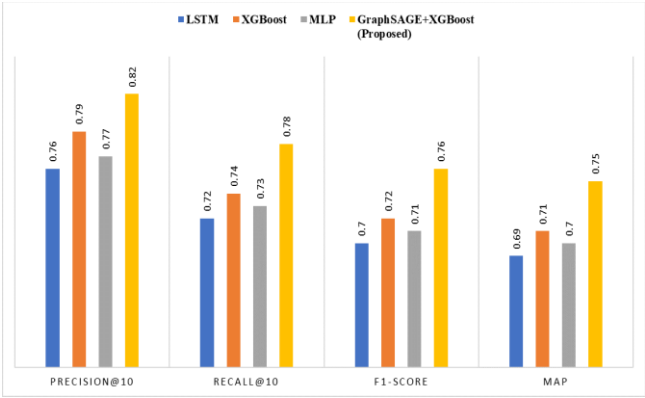


Fig. 5. Visual comparison of different predictive models.

Fig. 6 shows the curves of the training and validation accuracy of the proposed model. The model persistently increases in training and validation accuracy by the last epoch to about 0.89 and 0.84, respectively. That shows it can learn useful features from graph-structured user-product interaction data and achieve excellent classification performance with XGBoost. The monotonically increasing trendline of the validation accuracy suggests good generalization and a low risk of overfitting.

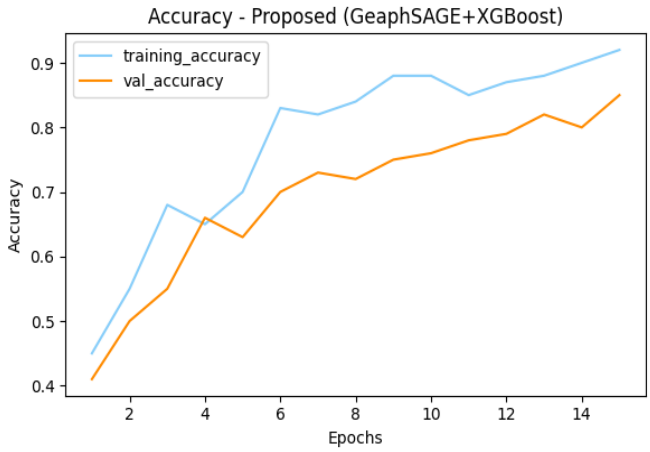


Fig. 6. Training and validation accuracy trends over epochs.

The loss curves during training and verification are set out in Fig. 7. The two loss values drop significantly as the epochs increase, with the training loss tending towards 0.3 and the validation loss towards 0.4. The fact that the curve is declining and the difference between the training and validation losses is minimal further assures the model's stability and appropriateness for recommending tasks that entail intricate interactions.

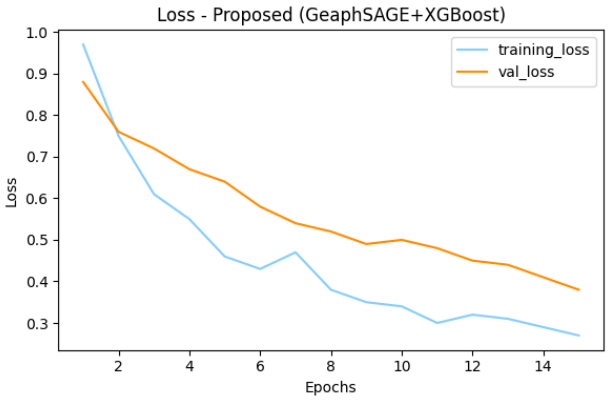


Fig. 7. Training and validation loss trends over epochs.

The effectiveness of the proposed model in revealing meaningful patterns in co-purchase behavior is evidenced by these visualizations, which, in turn, yield better performance than traditional methods.

C. Product Co-Purchase Network with Community Detection and Edge Weights

Fig. 8 illustrates a sample of frequently bought-together grocery items and creates a co-purchasing network based on the sample. The nodes represent products, and the edges indicate co-purchase relationships; the width of an edge indicates the relative frequency of co-purchases. The graph is a good representation of most shopping patterns on online platforms like Instacart. For example, the relationships with high co-purchase rates, like Milk Bread, Milk Eggs, and Banana Yogurt, are not the only ones at a high level; however, such relationships can indicate the possibility of developing repeat consumer behavior.

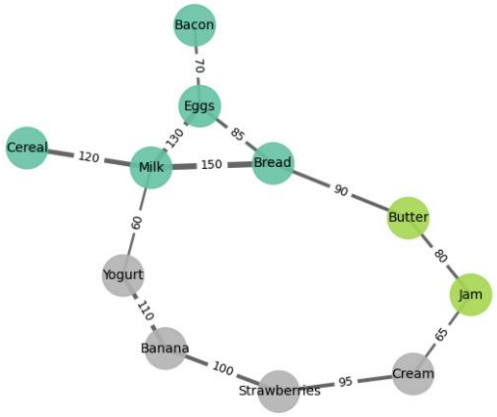


Fig. 8. Product co-purchase network with community detection and edge weights.

The main characteristic of this network is that it automatically partitions products into distinct communities via modularity optimization. The graph has various colors to represent these communities. Remarkably, there is one such community, including Milk, Bread, Eggs, and Butter, which conveys a traditional image of breakfast or a basic grocery basket. The Strawberry, Banana, Yogurt, and cream community correspond to another community that falls into the health-conscious or smoothie-related content basket. The visualization already shows core and peripheral products and indicates that some, such as Milk, are central hubs, connecting several communities.

This would mean that this type of product may become a critical indicator point in the recommendation engines and promotional activities. In addition, it is possible to present edge labels with precise co-purchase frequencies, which provide an even greater opportunity to infer the strength of connections between products. This gives a better view of the structure and behaviors in consumer buying information, which can be helpful in market basket analysis or inventory planning models in online retail platforms, as well as consumer personalization strategies.

D. Expanded Community Analysis in a Diverse Product Network

Fig. 9 extends the discussion of the community detection algorithm's usefulness to a less homogeneous product network. The graph includes not only regular grocery items but also narrower, health-related products like Tofu, Quinoa, and Lentils. The resultant communities are then visualized using a unique color map, and the networks are Spring-embedded to capture both intra- and inter-community associations.

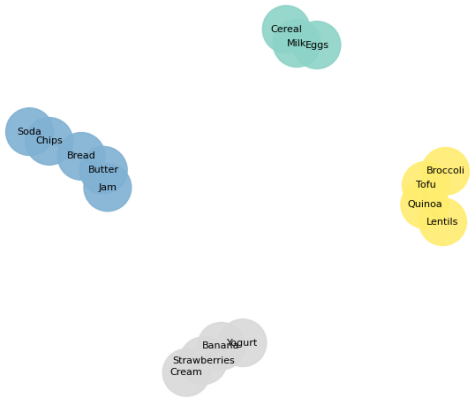


Fig. 9. Community Detection in Product Co-Purchase Network.

This elaborate analysis reveals several interesting clusters. As an example, there is a community with Milk, Eggs, and Cereal, or the essentials of a planetary breakfast. The other cluster comprises Tofu, Quinoa, Broccoli, and Lentils, representing a vegetable- or vegan-oriented consumer group. In the meantime, Chips and Soda have an intimate compact community, which is probably a sign of impulse purchase or a bundle of snacks. The findings illustrate the effectiveness of the community detection method for identifying latent consumer segments based on co-purchasing behavior. The clusters might present distinct shopper profiles, such as health-centered, cost-conscious, or indulgent customers. Recognition of these communities can have important implications for customized marketing, special product packages, and supply chain management. Moreover, the size of nodes and the thickness of edges still provide an intuitive clue as to the importance of the items or the strength of the relation, and the visualization is thus both analytically advantageous and simple to decode.

V. Conclusion

Financial. The research proposed a scalable hybrid model that combines graph representation learning with GraphSAGE

and an XGBoost gradient-boosted model for real-time decision support in retail analytics. The test helped the framework measure both structural and behavioral patterns among consumers by modelling transactional data from the Instacart platform as a bipartite user-product graph. It was noted that the proposed model showed considerable enhancements to the conventional models of LSTM, MLP, and unaligned XGBoost, in predictive accuracy, Precision, Recall, and average mean Precision. Moreover, bigram market structure analysis identified groups of latent consumer segments and frequently occurring co-purchasing activity, yielding a valuable source of actionable insights into how stock should be controlled and how products could be marketed on a one-to-one basis.

Nevertheless, there are still some limitations. To begin with, the current structure uses a static graph that cannot accurately reflect the real-time environment of changing user requirements or products under pre-launch. Second, the model fails to include either textual or image data, which could enhance product representation.

Third, despite the GraphSAGE+XGBoost architecture's effectiveness as an effective middle ground between interpretability and accuracy, it may not be possible to extend it to ultra-large settings with millions of nodes and dynamic edges on some computers due to resource constraints. Also, the analysis was restricted to one data set (Instacart). It cannot be applied to various retail environments or even other application areas. Future directions will focus on overcoming these limitations and extending the framework to support dynamic graph learning and the inclusion of multimodal data, including user reviews and product descriptions. Future work on self-supervised graph pretraining and contrastive learning methods can ultimately help the model achieve high Precision in cold-start conditions. Besides, online learning processes and immediate response chains can be incorporated to maintain readiness to change the market grid. Lastly, during model implementation in live e-commerce systems, it will be practically validated that the model is effective and efficient for decision support.

Disclousure and Conflict of Interest

The author declares that there are no conflicts of interest related to this research. Additionally, the author has no financial interests or competing affiliations that could have influenced the study's design, execution, or findings. This manuscript is the author's original work and has not been previously published or submitted for review to any other journal or conference.

References

[1] I. B. Ridwan, "Optimizing enterprise decision-making through causal machine learning models and real-time business intelligence integration."

[2] N. Prova, "Multilingual emotion classification in e-commerce customer reviews using gpt and deep learning-based meta-ensemble model," Available at *SSRN* 5161505, 2025.

[3] D. Patil, "Artificial intelligence in retail and e-commerce: Enhancing customer experience through personalization, predictive analytics, and real-time engagement," *Predictive Analytics, and Real-Time Engagement* (November 26, 2024), 2024.

[4] K. T. Giang, "A data-driven approach to smart shopping: Optimizing grocery trips using geolocation and store inventory data," 2025.

[5] S. Shahrukhan et al., "Enhanced sales analysis: predictive insights from machine learning models using the xgboost regressor approach," in *2nd International Conference on Computer Vision and Internet of Things* (ICCVIoT 2024), vol. 2024. IET, 2024, pp. 199–205.

[6] X. Liang, L. Lin, X. Shen, J. Feng, S. Yan, and E. P. Xing, "Interpretable structure-evolving lstm," in Proceedings of the *IEEE conference on computer vision and pattern recognition*, 2017, pp. 1010–1019.

[7] O. Badmus, S. A. Rajput, J. B. Arogundade, and M. Williams, "AI-driven business analytics and decision making," *World Journal of Advanced Research and Reviews*, vol. 24, no. 1, pp. 616–633, 2024.

[8] S. Chintala and V. Thiyagarajan, "AI-driven business intelligence: Unlocking the future of decision-making," *ESP International Journal of Advancements in ComputationalTechnology,* vol. 1, pp. 73–84, 2023.

[9] V. H. D. Reddy, U. SM, M. Yadav, D. Hazarika, and A. Sharma, "Role of artificial intelligence in business intelligence and decision making: an empirical study," *European Chemical Bulletin*, vol. 12, no. 3, pp. 165–172, 2023.

[10] Y. Niu, L. Ying, J. Yang, M. Bao, and C. Sivaparthipan, "Organizational business intelligence and decision making using big data analytics," *Information Processing & Management*, vol. 58, no. 6, p. 102725, 2021.

[11] S. Hanumanthu, "Towards a novel and intelligent e-commerce framework for smart-shopping applications," Master's thesis, *The University of Western Ontario (Canada)*, 2022.

[12] Z. Farheen and A. Dharani, "Prediction of customer purchasing patterns for retail optimization using market basket techniques," in 2024 *8th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS). IEEE*, 2024, pp. 1–5.

[13] M. Meftah, S. Ounacer, and M. Azzouazi, "Optimizing purchase predictions in retail: A network science and artificial intelligence approach." *International Journal of Intelligent Engineering & Systems*, vol. 18, no. 6, 2025.

[14] M. Ariannezhad, S. Jullien, M. Li, M. Fang, S. Schelter, and M. De Rijke, "Recanet: A repeat consumption-aware neural network for next basket recommendation in grocery shopping," in Proceedings of the *45th international ACM SIGIR conference on research and development in information retrieval*, 2022, pp. 1240–1250.

[15] K. Patel, "Instacart market basket analysis," Ph.D. dissertation, *California State University, Northridge*, 2022.

[16] J. Sun and J. Kim, "Joint prediction of next location and travel time from urban vehicle trajectories using long short-term memory neural networks," *Transportation Research Part C: Emerging Technologies*, vol. 128, p. 103114, 2021.

[17] N. N. I. Prova, "Healthcare fraud detection using machine learning," in 2024 Second *International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI). IEEE*, 2024, pp. 1119–1123.

[18] I. Petkovski, "Optimizing multilayer perceptron neural network hyperparameters," *Journal of process management and new technologies*, vol. 13, no. 1-2, pp. 81–101, 2025.

Evaluating the Efficiency of LLM-Generated Software in Resisting Malicious Attacks

Dominic Niceforo, Haydar Cukurtepe
Department of Computing and Information Sciences,
Valparaiso University, Valparaiso, Indiana 46383-7123

***Abstract*—This study introduces a structured framework for evaluating the security of Java applications generated by large language models (LLMs) and presents the results from its implementation across three models: DeepSeek, GPT-4, and Llama 4. The framework integrates Open Web Application Security Project (OWASP)-supported tools, such as SpotBugs with FindSecBugs, OWASP Dependency Check, and OWASP Zed Attack Proxy (ZAP), alongside the NIST Risk Management Framework. These tools and standards were selected for being publicly available, allowing this process to be replicated and extended without proprietary licensing, and for their alignment with widely adopted industry benchmarks. The testing methodology for generated Java applications includes static code analysis, third-party dependency checking, and dynamic attack simulation. Each of the specified tools for this study corresponds to identifying a specific category of critical vulnerabilities. Identified vulnerabilities are then evaluated against NIST risk analysis standards to characterize their threat sources, likelihoods, and impacts, as well as their implications for the overall security risk profile of each application. The effect of prompt design is also explored by comparing a neutral prompt against a security-emphasized prompt incorporating OWASP best practices. Results varied considerably across models: GPT-4 showed noticeable improvements across critical and high-severity vulnerabilities, with 33.3% and 53.8% reductions, respectively. However, Llama 4 and DeepSeek saw an increase in vulnerabilities from the neutral to the secure prompt. Llama 4 had a general increase of 10-15% across critical, high, and medium-severity vulnerabilities, while DeepSeek saw no change in high-severity vulnerabilities and a 40% increase in low-severity vulnerabilities. The framework presented provides a structured process for evaluating LLM-generated code against established software development and security standards, while identifying present limitations and possible directions for future work.**

***Keywords*—Artificial intelligence; cyber security, large language models (LLMs); software security evaluation**

I. Introduction

A core goal in computer security is to reduce software vulnerabilities, both by identifying existing flaws and preventing new ones from emerging. Yet, security problems are still common in software systems. For instance, a report shows that 48% of tested software contains critical, high-risk vulnerabilities [1]. Another report indicates the common vulnerabilities and exposures (CVEs) volume for 2025 is rapidly increasing, with a significant jump from previous years, 16-18% by mid-year. This change is driven by faster reporting and an increased focus on software flaws [2]. Even as advancements in AI offer new ways to aid software development, these security concerns remain, potentially creating new attack vectors in the quality of code generated.

In [3], it is noted that more than 800 million users actively engage with ChatGPT [4] each week, collectively generating over 5.8 billion queries per month. LLMs have been trained to work in a wide variety of domains and have also been integrated into software development environments. The growing use of LLMs for code generation presents both opportunities and significant challenges. As models like GPT-4 [4], DeepSeek [5], and Llama 4 [6] become better trained at generating code, there is a growing interest in handing over parts of the development process to LLMs [7]. These reasons include streamlining coding tasks, enhancing productivity, and reducing development time. While LLMs can speed up the process of writing functional code, this convenience comes with substantial risks to software quality, security, and developer accountability. A growing reliance on these models to handle code generation does not address the challenges, particularly regarding security, which remains largely untested in real-world scenarios. Traditional software development methodologies prioritize secure coding practices, such as threat modeling, code reviews, and penetration testing, to identify and address potential vulnerabilities. In contrast, LLMs face limitations during training, where models lack proper optimization for secure code generation, and during deployment, when LLMs are used for more sensitive tasks without appropriate oversight or validation. As a result, this raises significant concerns about the overall security of LLM-generated solutions. Some studies indicate that the use of LLMs negatively affects code generation. In [8], the authors conducted a controlled experiment with two groups, one of which worked with the help of AI code generators, and the other without their assistance. The authors observed that participants with access to an AI assistant wrote more insecure solutions than those without it. Similarly, Sarkar et al. [9] observed that LLM-supported programming assistants have a negative influence on code development. They also found that inexperienced users often trust code generated by AI tools without much consideration. The survey [10] also highlights the importance of clean and healthy code during LLM training.

Understanding the code-generation capabilities of LLM tools will benefit all relevant groups. This research examines the extent to which the generated code adheres to current security standards and performs under structured vulnerability assessments. Our study integrates established frameworks, such as the OWASP Top 10 [11] and the NIST Risk Management Framework [12], to evaluate Java applications generated by different LLMs. Using tools such as SpotBugs, FindSecBugs, OWASP Dependency Check, and OWASP ZAP, we conduct both static and dynamic testing, as well as third-party dependency analysis. We then analyze, classify, and score

the findings using the NIST framework to produce quantifiable risk assessments.

This research aims to answer multiple key questions about the capabilities of commonly used LLM tools:

- How effective are LLMs in generating code that can resist specific cyberattacks?
- What security benchmarks can accurately evaluate the resilience of this code?
- What attack vectors are these applications most vulnerable to?

This study hypothesizes that secure prompting will influence the security profile of generated applications, with the direction and degree of change expected to vary across the different models, given potential differences in architecture and training.

By exploring these questions, this research establishes a replicable framework for evaluating the security of code generated by LLMs for a structured application. Additionally, it will provide valuable insights into how prompt design can influence more secure code output.

The following sections organize the remainder of this study: Section II provides a brief overview of the background of AI-assisted software development, along with related works, with a special emphasis on performance evaluations and security concerns. Section III expands on how the evaluation process will be implemented within the proposed framework. Section IV presents experimental results, performance comparisons, and discusses the key findings. Lastly, the conclusion in Section V discusses the current limitations of the research, possible future work, and concludes the study.

II. Background and Related Work

This study aims to investigate the efficiency of LLMs in software development, specifically their ability to generate code that withstands commonly known attack types. As AI-powered tools have become increasingly integrated into the general population, their ability to create high-quality code has been the subject of numerous studies.

The survey [10] highlights the powerful capability of LLMs, especially for code generation, when high-quality, clean, and healthy code is used for training. The authors systematically review the use of LLMs in software security. They break down software security into stages and analyze how LLMs can be utilized in each stage. The authors emphasize the powerful potential of LLMs, especially when pretrained with clean data, fine-tuned for specific tasks, and carefully crafted input prompts. A comprehensive survey of LLMs for Software Engineering (SE) is given in [13]. This survey concludes that traditional SE methods, combined with LLMs, play a critical role in enabling reliable and efficient LLM-based SE. The surveys [13], [14] provide AI-assisted development and open technical challenges that highlight their capabilities, limitations, potentials, and security concerns. In [14], the authors reviews how embeddings can be utilized for detecting security vulnerabilities by comparing them to known code patterns. They also suggest that the transformers can be trained to detect security vulnerabilities by learning from labeled data. In [15], the authors provide a survey on the usability of AI tools to assist in software development, with a focus on Copilot and GitHub. The study is based on a survey of a diverse population of developers. The authors observe that developers are motivated to use AI programming assistants; however, 54% are reluctant to do so due to security concerns. In [16], the authors provide a comprehensive survey on LLMs for security. The authors investigate the use of LLMs for identifying possible vulnerabilities in natural language descriptions and source code, as well as generating security-related code, such as patches and exploits. In [17], the authors provide a survey on the security aspects of code generated by LLMs. The authors emphasize the importance of employing robust tools and approaches to enhance code quality.

In [18], the authors provide an analysis of the quality of solutions generated by AI tools, including code writing, testing, reviewing, and documentation. The study is conducted using Claude 3 Opus and GitHub Copilot. The performance of the tools is compared with three security factors: “Provided security configuration from the start”, “Filter out malicious input”, “Up to date on libraries information”, but it does not provide a comprehensive analysis of vulnerabilities. In [19], the authors focus on generating Solidity code with GitHub Copilot. AI-generated code is scanned for vulnerability using a static analysis framework for smart contracts. Despite the advantages of using AI, there are significant challenges, particularly concerning the security, reliability, and quality of the generated code. In [20], [21], the authors focus on developer use of AI tools. In [22], the authors researches the security concerns of developers.

There are many studies evaluating the correctness and usability of the LLM-generated code [23], [24], [25]. The study [26] analyzes developments in AI-supported code generation, focusing on safety, reliability, and code quality. In [27], the authors evaluates the code generated from GitHub Copilot using MITRE vulnerability standards. The authors suggest pairing Copilot with appropriate security-aware tooling during both training and generation to minimize the risk of introducing security vulnerabilities. The study [28] proposes a controlled code generation method (svGen) using LLMs. The authors train the model with secure-accepted code and evaluate the results using a model proposed by [27]. The authors employ this approach to improve the security of LLM-generated code from 59% (for considered cases) to 92%. The evaluation model used in this study needs to be aligned with industry standards and repeatable.

In [29], the authors tested CodeBERT, GPT-3.5, and CodeX for their code generation capabilities and security concerns related to 10 different vulnerabilities, which were selected from common CVEs, Common Weakness Enumerations (CWEs), NIST vulnerabilities, and OWASP’s Top 10 web application security risks. The authors conclude that ChatGPT excels at explaining potential security issues and suggest that all AI models should be integrated with cybersecurity practices. Our study differs from this study in its comprehensive analysis of vulnerabilities, repeatability of implementation framework, modern LLM implementations, and NIST evaluations. Berabi et al. [30] investigate the effectiveness of LLMs for solving code-repair tasks. The authors present a new learning model

that trains LLMs to fix coding errors and vulnerabilities. The authors concluded that improving code quality and security is difficult, as it requires large amounts of high-quality, labeled training data. Another type of risk associated with using AI tools in code generation is highlighted by [31]. The study emphasizes that an attacker can poison the data, infecting AI code generators and leading them to generate code that contains vulnerabilities and security defects. The studies [9], [22] indicate that, although developers have serious security concerns about AI-generated code, AI use will continue to increase for productivity purposes.

Prior studies in this area can be generally categorized into three groups: the first focuses on performing static analysis on LLM-generated code, such as [27] and [19]. The second group examines developer behavior and potential concerns arising from the use of AI tools in practice, such as [9], [22], and [15]. The third group explores the broader capabilities and generation approaches, such as [29] and [28].

The present study differs from all three categories by combining static analysis, third-party dependency checking, and dynamic attack simulation within a single evaluation pipeline and assessing it against the NIST Risk Management Framework. This reflects the multi-faceted nature of real-world security threats, where no single analysis method captures the full risk profile of an application. By integrating the specializations of prior work into a unified framework, this study provides a more complete picture of security vulnerabilities in a given code-generation scenario.

III. Methodology

This section details the implementation of the evaluation process and proposes a framework that integrates software generation, security benchmarking, and attack testing across selected LLMs.

The attack vectors tested in this study are based on the OWASP Top Ten [11], an openly accessible document that outlines the most common and critical security risks faced by web applications. These include:

- Broken Access Control
- Cryptographic Failures
- Injection
- Insecure Design
- Security Misconfiguration
- Vulnerable and Outdated Components
- Identification and Authentication Failures
- Software and Data Integrity Failures
- Security Logging and Monitoring Failures
- Server-Side Request Forgery

The OWASP Top Ten list is globally recognized for providing a thorough, systematic approach to identifying, prioritizing, and classifying vulnerabilities relevant to the type of application generated in this study. We specifically chose this benchmark as our primary reference because of its widespread acceptance in academia and industry [32], [33], [34]. While the OWASP Top Ten is updated regularly to keep pace with new threats and research, using the current version at the time of our study, OWASP Top Ten 2025, ensures our methods are transparent, consistent, and reproducible. By clearly documenting these categories and their integration into our testing, our framework directly aligns with current security standards, making it more credible and applicable. The framework also incorporates the NIST Risk Management Framework to deliver clear, quantifiable assessments of potential security risks.

The proposed framework comprises seven primary components, which form a sequential process for evaluating LLM-generated code from a security perspective, as illustrated in Fig. 1. Details about its implementation and varying components are explained in the following subsections.

A. LLMs and Prompts

LLMs are widely adopted and used intensively for many purposes, including software development assistance [12].

1) LLM selection: In this study, we have selected three distinct large language models: GPT-4, DeepSeek, and Llama 4. Each LLM offers a different perspective on what comprises the evolving landscape of AI. GPT-4 stands as the industry standard, well-known, widely trusted, and high-performing. DeepSeek is a newer model, and its inclusion in this study prompts a different methodological model design. Lastly, Llama 4, which is partially open-source, invites innovation and transparency. Together, these choices form a well-rounded representation of the variety expected of available models and, therefore, an accurate basis for comparison in this study.

2) Prompt preparation: This step involves drafting comprehensive project specifications for an LLM to generate a Java-based e-commerce web application. The primary goal is to ensure the instructions are easy to understand and consistently reproducible across different models. Additionally, by adding intentional complexity, the prompt acts as a controlled environment for testing how the model handles various categories of security vulnerabilities.

The neutral prompt, which serves as the baseline for creating the project, is as follows:

"Create a simple Java web application for an e-commerce platform. Include the following features: user registration and login, a product catalog with basic search functionality, an admin page to add products, and file upload support for product images. Keep the structure as simple as possible, and make sure the core features are implemented. Use H2 as the database, Spring MVC for routing, and Thymeleaf templates for the UI. Structure the application as a Maven project and include a working pom.xml with necessary dependencies to build and run the app."

In comparison, the second prompt is meant to introduce a greater focus on security. For the context of this study, only one secure prompt is used. However, a greater number of prompts and variance in its security emphasis can be beneficial, as mentioned in the Limitations section of this study. While the secure prompt is identical in its project specifications, this prompt incorporates safe coding practices that would typically be followed in a traditional development cycle, as well as types

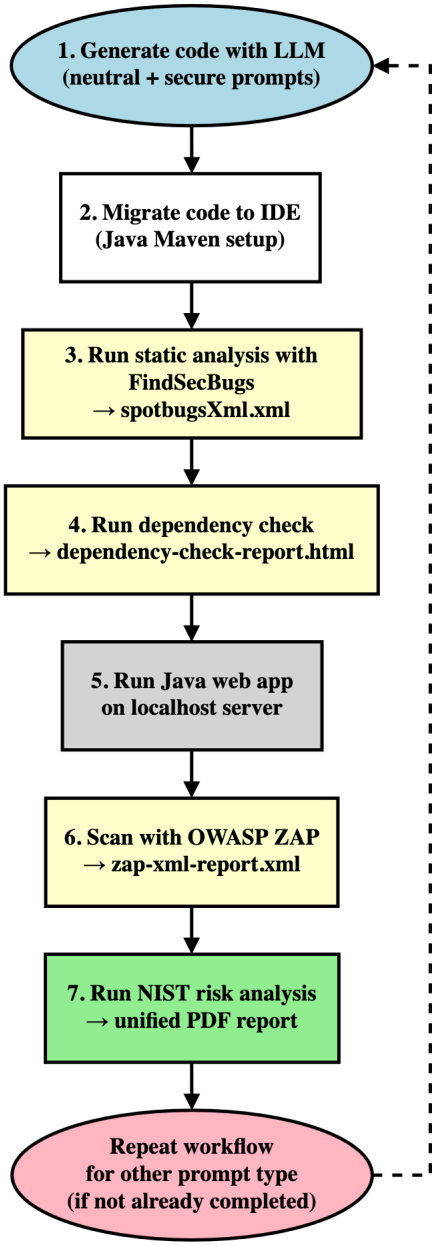


Fig. 1. Overview of the primary components within the proposed framework for evaluating LLM-generated applications.

of potential vulnerabilities to avoid when implementing core features.

"Implement the following OWASP best security practices: use the latest version of dependencies, input validation and output encoding, use prepared statements for database access, hash passwords using BCrypt, implement access control (e.g., admin vs user), validate uploaded files and store them safely, avoid hard coded secrets (use application.properties), handle errors securely (do not expose stack traces), and include comments explaining your security-related decisions."

B. Code Migration

In order to evaluate the security of LLM-generated applications, a practical testing environment is created by initializing a new Java project in an Integrated Development Environment (IDE). This approach closely follows a real-world developer workflow, enabling an accurate testing environment that simulates realistic development and security assessments. In the current version of the study, code is migrated manually from the LLM's output into a Maven project. However, automated integration, as an extension of this study, is considered in the Limitations section.

Within the project, the security tools explained in the following subsections are integrated via the Maven pom.xml configuration file.

C. Static Vulnerability Analysis

Static code analysis is essential in preventing costly security breaches by identifying vulnerabilities early, saving time and resources. It also helps improve code quality by enforcing standards.

SpotBugs and FindSecBugs are commonly used tools for static analysis of the source code, without actually running the code. SpotBugs checks for over 400 bug patterns, categorized by severity [35]. This tool is particularly effective in identifying issues such as hard-coded credentials, directory traversal, SQL injection, and weak cryptographic implementations. These vulnerabilities align with multiple OWASP categories, including Broken Access Control, Injection, and Cryptographic Failures. In this study, we utilized SpotBugs version 4.9.2.0 and FindSecBugs version 1.12.0 to identify vulnerabilities.

The FindSecBugs tool is compatible with the Java development environment. After configuring the tools, the IDE performs automated scans and generates detailed reports. Exported in both HTML and XML formats, these reports are stored for further review.

D. Dependency Analysis

Dependency analysis is another essential component of the study methodology, as it mitigates vulnerabilities caused by outdated dependencies and libraries in the generated application.

OWASP Dependency Check is a command-line tool that scans for known vulnerabilities in project dependencies by comparing them against databases like the National Vulnerability Database (NVD) [36]. It is used to identify vulnerabilities in third-party libraries. This tool scans through the project's dependencies and maps them to known CVEs from public vulnerability databases. As one of the most prevalent risks for web applications, using Dependency Check in this testing methodology addresses the OWASP category of Vulnerable and Outdated Components. In this study, we used version 6.5.3 to identify vulnerabilities.

OWASP Dependency-Check tool is compatible with the Java development environment. After configuring the tools, the IDE performs automated scans and generates detailed reports. Exported in both HTML and XML formats, these reports are stored for further review.

E. Running Code on a Server

Once the static analysis and dependency checks are complete, the application is deployed locally to a web server from the IDE. This step enables dynamic analysis, which involves observing and testing the application while it is running.

F. Dynamic Vulnerability Analysis

OWASP ZAP is used at this point to scan the live application, conducting various vulnerability tests that mimic possible attack vectors by an external attacker.

OWASP ZAP (Zed Attack Proxy) simulates live attacks against an application while running on a web server for dynamic testing [37]. ZAP intercepts and manipulates HTTP requests, helping to satisfy the OWASP categories of Broken Access Control, Cross-Site Scripting (XSS), and Authentication Failures. These dynamic tests are insightful for better understanding how this application will behave under real-world conditions, where it may be subject to more varied testing than provided by static analysis or unit testing. In this study, we used version 2.16.1 to identify vulnerabilities.

G. NIST Framework Analysis

The NIST Cybersecurity Framework serves as a comprehensive and valuable resource for organizations to manage and mitigate cybersecurity risks. It offers a structured, flexible, and repeatable methodology for identifying, assessing, and addressing security vulnerabilities across diverse operational contexts [12]. By adopting this framework, organizations can not only improve their alignment with regulatory and industry compliance requirements but also substantially reduce their exposure to cyber threats. Moreover, the framework supports the development of a resilient and proactive security posture, ultimately contributing to the organization's long-term stability and risk governance.

The NIST Risk Management Framework is used to convert raw output from the OWASP vulnerability assessments into quantifiable risk levels. This step provides greater structure to the security findings, supporting risk-based decision-making for individual developers and organizations throughout the software development process.

Data collected from the OWASP tools, such as SpotBugs, FindSecBugs, Dependency Check, and ZAP, is extracted and standardized. These include information such as CWEs, Common Vulnerabilities and Exposures (CVEs), and severity scores as reported by the tools.

Additionally, Python scripts are used to automate the process of querying the National Vulnerability Database (NVD) API. These scripts retrieve CVSS (Common Vulnerability Scoring System) data for each CVE, helping to determine the likelihood and impact of each vulnerability using NIST's criteria:

- Likelihood is determined by factors such as exploitability, prevalence, and exposure, referenced from CVSS subscores and vulnerability history.
- Impact is evaluated based on the nature of the affected component and the sensitivity of the data it handles.

Final risk levels are calculated by multiplying the likelihood and impact weights. They are categorized as Low, Moderate, or High as outlined by NIST standards.

IV. Experimental Results

This section presents the experimental results from implementing the proposed framework with two prompts: a *neutral* prompt, where security requirements are not emphasized, and a *secure* prompt, where the LLM is explicitly asked to create secure code that avoids specific vulnerability groups. Results are contextualized by the framework with an additional focus on implementation details that required manual intervention.

TABLE I. Security Findings by Severity for Each LLM and Prompt.

LLM	Prompt	Critical	High	Medium	Low	Info
DeepSeek	Neutral	3	13	10	11	3
DeepSeek	Secure	2	13	14	14	4
GPT-4	Neutral	3	13	10	8	3
GPT-4	Secure	2	6	15	6	3
Llama 4	Neutral	10	9	6	11	3
Llama 4	Secure	11	10	7	11	3

A. LLM and Prompt Behavior

During the code generation process, each LLM's behavior is evaluated based on the functional quality of its output, responsiveness to the given prompt, and the degree of manual refinement required to produce a working Java application from the model (see Table I).

1) GPT-4: The neutral prompt generated a working Spring Boot web application that included basic controllers, models, and template views. Fig. 2 depicts the directory structure produced along with the source files. However, GPT-4's implementation introduced several security vulnerabilities, including 34 in total: 3 Critical, 13 High, 10 Medium, and 8 Low. These include nine significant issues reported by SpotBugs/FindSecBugs, such as a potential path-traversal issue (CWE-22) caused by improper file handling, and a failure to check the return value of the mkdirs() method (CWE-253).

OWASP ZAP reported five issues, including the absence of a Content Security Policy (CSP) header being set (CWE-693) and the use of cookies without the SameSite attribute (CWE-1275), both of which increase the web application's susceptibility to client-side attacks. However, most of the reported issues originated from OWASP Dependency Check, which identified all three vulnerabilities marked as critical. These include SnakeYAML, Spring Boot Starter Security, and Tomcat Embed Core. One or multiple published CVEs could reference each of these outdated dependencies. For example, the SnakeYAML version was linked to CVE-2022-1471, involving a deserialization vulnerability. On the other hand, Tomcat Embed Core was linked with 7 CVEs, ranging from improper input validation to path equivalence and the improper neutralization of escape sequences. Each of these vulnerabilities matches with one or more OWASP Top Ten vulnerabilities [11].

The secure prompt reduced critical-rank vulnerabilities by 1 to 2 and High-rated vulnerabilities by 7 to 6. These results

```
ecommerce-webapp/
├── pom.xml
├── src/
│   ├── main/
│   │   ├── java/
│   │   │   └── com/example/ecommerce/
│   │   │       ├── EcommerceApplication.java
│   │   │       ├── controller/
│   │   │       │   ├── AdminController.java
│   │   │       │   ├── AuthController.java
│   │   │       │   └── ProductController.java
│   │   │       ├── model/
│   │   │       │   ├── Product.java
│   │   │       │   └── User.java
│   │   │       ├── repository/
│   │   │       │   ├── ProductRepository.java
│   │   │       │   └── UserRepository.java
│   │   │       └── config/
│   │   │           └── SecurityConfig.java
│   │   └── resources/
│   │       ├── static/
│   │       ├── templates/
│   │       │   ├── login.html
│   │       │   ├── register.html
│   │       │   ├── catalog.html
│   │       │   └── admin.html
│   │       └── application.properties
│   └
```

Fig. 2. GPT-4 responding to the neutral prompt with a directory structure of files necessary for generating the Java-based web application.

include the SnakeYAML dependency issue, which was no longer reported by Dependency Check. However, new vulnerabilities were also introduced, including disabling cross-site request forgery (CSRF) protection in the security configuration (CWE-352) and omitting an anti-clickjacking header (CWE-1021). Additionally, some vulnerabilities classified as high were downgraded to medium, contributing to the increase in that category by 5 to 15 when compared to the neutral prompt. The cumulative total for this version was 29 vulnerabilities. These results suggest that while the secure prompt made modest improvements in its use of dependencies, it did not consistently address all vulnerabilities and also introduced new configuration risks.

2) DeepSeek: Overall, a comparison of the outputs from the two prompts shows that DeepSeek produced a more complex implementation of the Java-based web application than GPT-4. The applications generated by DeepSeek encompassed not only the configuration, controller, model, and repository classes, like with GPT-4, but also added additional files that enhanced functionality and structure. For example, the neutral prompt introduced the Data Transfer Object (DTO) and service classes for handling data manipulation and application logic.

After running the code through the OWASP security tools, the neutral prompt revealed 37 total vulnerabilities, including 3 Critical, 13 High, 10 Medium, and 11 Low. While these results were similar to those from the GPT-4 model, some notable exceptions warrant highlighting. One medium-level vulnerability was associated with version 3.1.2 of Spring

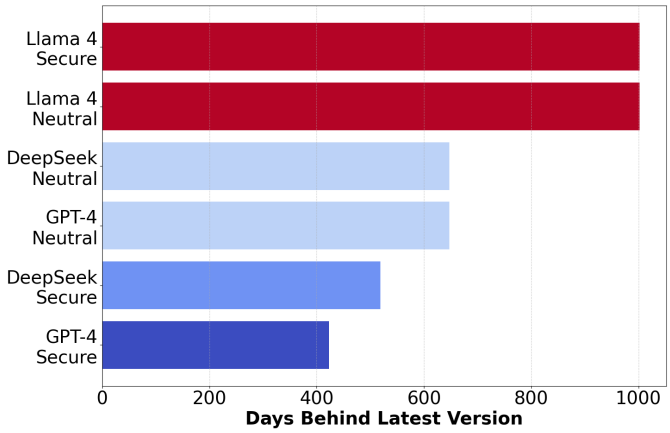


Fig. 3. Bar chart showing the number of days that the Tomcat dependency used in each application lagged behind the latest available version at the time of testing. Lower bars represent a more recent version of the dependency.

Boot DevTools, as detected by OWASP Dependency Check (CVE-2023-34055). Additionally, including the StorageService class by DeepSeek introduced another medium-level Potential Path Traversal vulnerability, in addition to what was already detected by SpotBugs with GPT-4 (CWE-22).

Looking into the secure prompt for DeepSeek reveals a greater degree of complexity than even its neutral variant. The changes included implementation classes for ProductService and UserService, which were separated from their interfaces to better align with the separation of concerns. A method call was also identified within the UserController that passed a null value to a non-null parameter during the user account registration process. This vulnerability, which involves input validation and error handling, is classified as CWE-476 in the public database. In addition, the secure prompt analysis flagged additional outdated dependencies, including version 1.70 of the Bouncy Castle Provider, which is used for cryptographic implementations. This issue was linked to five CVEs with low confidence. The secure prompt ultimately led to more vulnerabilities across all severity levels, highlighting the unintentional consequences that can arise from overengineering a secure prompt. Although there was one less critical vulnerability than in the neutral version, testing revealed four additional medium-level vulnerabilities and three new low-level vulnerabilities. The cumulative total for this version was 43 vulnerabilities.

3) Llama 4: Llama 4 exhibited less effective performance under both prompt conditions, showing minimal benefit from the introduction of secure coding practices. In both the neutral and secure variants, the model produced approximately 10 critical-rated vulnerabilities, the highest among the three LLMs tested. No measurable improvements were observed between prompts, with the secure version yielding an equal or greater quantity of vulnerabilities. These additional risks were mainly identified by OWASP Dependency Check. As shown in Fig. 3, the version of Tomcat Embed Core used for each model and prompt is graphed based on how far it lags behind the latest release available during testing in April 2025. Both outputs for Llama 4 included Tomcat Embed Core version 9.0.65, released in July 2022, in their pom.xml files. Lagging by over 2.5 years, this model's output is a relative outlier because it uses outdated

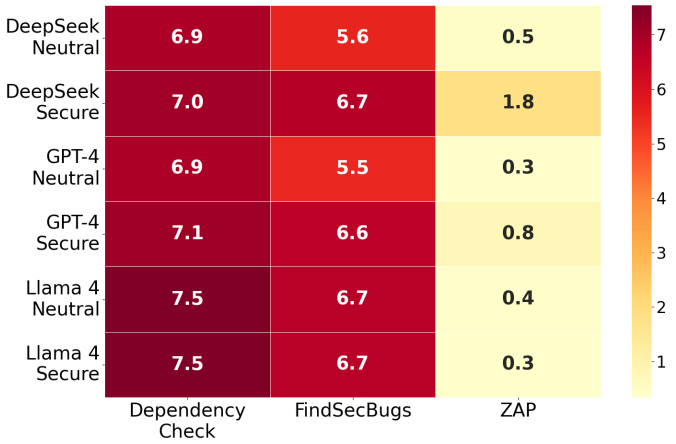


Fig. 4. Heatmap matrix showing average CVSS scores by OWASP tool across different LLMs and prompt types. Higher scores, visualized by darker colors, indicate more severe vulnerabilities.

components. This reliance reduces the application's resilience to withstand modern exploits and cyberattacks.

B. Overall Assessment

The results across all three models suggest that while using more secure prompts can influence improvements, such as using newer versions of dependencies and reducing the number of vulnerabilities, their impact is largely inconsistent and model-dependent. For example, GPT-4 responded most effectively to secure prompting, showing notable reductions in high-severity vulnerabilities. DeepSeek took a more detailed approach to its structure and implementation, but did not implement any consistent security improvements. And lastly, Llama 4 displayed no measurable improvements, continuing to use severely outdated components such as Tomcat Emded Core from mid-2022. The most critical issues across all models were identified by OWASP Dependency Check, highlighting common vulnerabilities in the versions of third-party libraries in use. Additionally, the secure prompts exhibited some natural variance in the generation process, including misconfigurations introduced from the added complexity (e.g., disabled CSRF protection). Fig. 4 presents a heatmap showing average CVSS scores by OWASP tool across different LLMs and prompt types.

C. NIST Risk Analysis

Before any risk could be scored using the NIST Risk Management Framework, it was necessary to normalize and enrich the raw vulnerability data generated from the OWASP security tools. Results from SpotBugs, FindSecBugs, OWASP Dependency Check, and OWASP ZAP were exported in HTML or XML, depending on the specific tool and configuration used. These saved reports contained valuable information for risk evaluation purposes, but required additional steps to easily extract identifiers such as CWEs and CVEs (see Fig. 5).

Custom Python scripts were used to parse for this type of information in each of the output files. For example, the XML-formatted reports from both SpotBugs and ZAP listed vulnerabilities with any CWEs associated with code smells

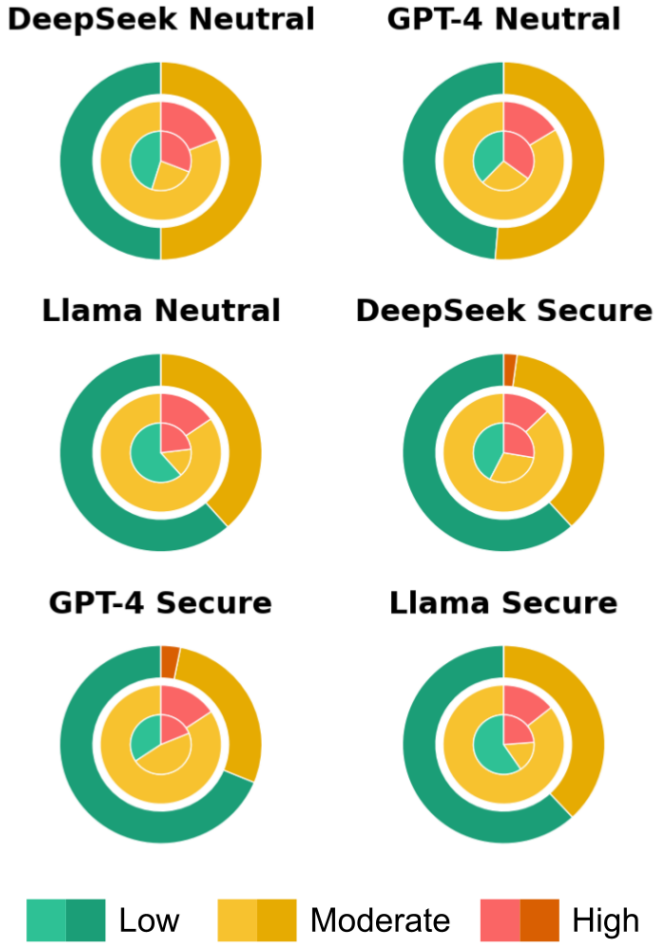


Fig. 5. Concentric pie charts visualizing the distribution of likelihood (inner ring), impact (middle ring), and overall risk (outer ring) for each LLM under neutral and secure prompt conditions. Severity levels are color-coded as Low (green), Moderate (amber), and High (red).

or dynamic flaws. Dependency Check, on the other hand, linked its findings directly to CVEs for a specific outdated and vulnerable component. The format and structure of the outputs generated by these tools served as the foundation for mapping findings to standardized definitions of known security issues.

V. Conclusion

In this study, we have examined LLM-generated code against modern security standards to better understand its resiliency to malicious attacks. We used the established OWASP Top Ten framework to categorize attack types, and the NIST Risk Management Framework to evaluate the risks associated with identified vulnerabilities. Using tools such as SpotBugs with FindSecBugs, OWASP Dependency Check, and OWASP ZAP, we conducted static analysis, third-party dependency scanning, and dynamic testing. Across the various testing configurations, all outputs, collected in either HTML or XML formats, were passed through the NIST Risk Management Framework. This necessary step standardized the findings and aligned each identified weakness, as outlined by Common Weakness Enumerations (CWEs), with Common Vulnerabil-

ities and Exposures (CVEs) from the National Vulnerability Database. By linking the identified weaknesses to known vulnerabilities already published, we have established a quantifiable basis for assessing security in applications influenced by large language models.

A. Research Limitations

Several limitations of this study should be acknowledged. First, all experimental results were obtained from testing a single application type, a Java-based e-commerce web application. This constrains how any conclusions about prompt-driven security improvements can be generalized. It also introduces a potential bias not only from the application type but also from the programming language used, as models may generate code differently to satisfy project requirements, which could inadvertently alter the output's security profile. Projects that vary in codebase size also pose a concern, as vulnerability counts in this study are reported in absolute terms without normalization. Overall, future work of this study should investigate how differences in programming languages, normalized project sizes, and application types affect a model's adherence to security-emphasized prompting.

Second, each prompt-model combination was evaluated using a single generated output. LLMs are inherently stochastic, meaning that repeated generations from the same prompt can yield different solutions, varying in how the code is structured, implemented, and supplemented with security requirements. Reproducibility was further hindered by reliance on human intervention during testing, including manual code migration into the development environment and error resolution. Without automation, replication across runs in the testing process cannot be guaranteed, as human intervention can influence the outcomes.

Third, this study used only two prompt conditions: a neutral prompt to establish the project requirements and a security-emphasized prompt incorporating OWASP best practices for secure code generation. While this binary design simplifies the study and allows for a controlled comparison, it does not capture the full range of outcomes that could be observed across a gradient of security emphasis. Exploring these variables within prompt design would provide valuable insights beyond the current state of this research.

B. Future Work

Experimental results were obtained using the latest version of the LLMs at the time of the study. This field of AI is highly dynamic and continuously improving. A future study can be designed to test the performance of newer versions to observe their improvements, specifically in resisting malicious attacks.

Also, the study can be designed to implement iterative code enhancements to exploit the refined reasoning ability of LLMs. Applications generated by LLMs would undergo an automated evaluation process using the OWASP tools and the NIST Risk Management Framework, with the results informing any necessary changes in subsequent versions of the application. This loop would involve multiple cycles of code generation, vulnerability scanning, and risk assessment, with each cycle feeding back into the prompt or model configuration to produce a more secure version. Over multiple iterations, LLMs could learn to avoid common vulnerabilities and generate code with a more secure profile. Such an implementation would not only automate vulnerability detection but also enable continuous security improvement, paving the way for LLM-driven solutions to create software that is production-ready and compliant with established policies.

In this rapidly evolving field of study, this work, despite its shortcomings, opens doors and guides research in different directions.

REFERENCES

[1] "Software Vulnerability Snapshot 2024" Report by Black Duck, Retrieved from http://www.blackduck.com

[2] Deepstrike (2025, Oct 08), "Vulnerabilities Statistics 2025: Record CVEs, Zero-Days & Exploits", Retrieved from https://deepstrike.io/blog/vulnerability-statistics-2025

[3] S. Singh, (2025, May 19) ChatGPT Statistics 2025 – DAU & MAU Data (Worldwide) https://www.demandsage.com/chatgpt-statistics/ Accessed 22 May 2025

[4] ChatGPT, GPT-4, OpenAI, chat.openai.com, Accessed 15 June 2025

[5] DeepSeek, https://www.deepseek.com/, Accessed 15 June 2025

[6] Llama-4, https://www.llama.com/models/llama-4/, Accessed 15 June 2025

[7] Stack Overflow. (2024). *2024 Developer Survey: AI and developer tools*. https://survey.stackoverflow.co/2024/ai Accessed 06 June 2025

[8] Perry, Neil, et al. "Do users write more insecure code with AI assistants?" Proceedings of the 2023 ACM SIGSAC conference on computer and communications security. 2023.

[9] Sarkar, Advait, et al. "What is it like to program with artificial intelligence?." arXiv preprint arXiv:2208.06213 (2022).

[10] Zhu, Xiaogang, et al. "When software security meets large language models: A survey." IEEE/CAA Journal of Automatica Sinica 12.2 (2025): 317-334.

[11] OWASP Foundation. (2021). OWASP Top 10: 2021. https://owasp.org/www-project-top-ten/2021

[12] National Institute of Standards and Technology. (2018). Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy. U.S. Department of Commerce, Washington, D.C., NIST Special Publication (NIST SP) 800-37r2. https://doi.org/10.6028/NIST.SP.800-37r2

[13] Fan, Angela, et al. "Large language models for software engineering: Survey and open problems." 2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering (ICSE-FoSE). IEEE, 2023.

[14] Kotsiantis, Sotiris, Vassilios Verykios, and Manolis Tzagarakis. "Ai-assisted programming tasks using code embeddings and transformers." Electronics 13.4 (2024): 767.

[15] Liang, Jenny T., Chenyang Yang, and Brad A. Myers. "A large-scale survey on the usability of ai programming assistants: Successes and challenges." Proceedings of the 46th IEEE/ACM international conference on software engineering. 2024.

[16] Xu, HanXiang, et al. "Large language models for cyber security: A systematic literature review." arXiv preprint arXiv:2405.04760 (2024).

[17] Ramírez, Leonardo Criollo, et al. "State of the Art of the Security of Code Generated by LLMs: A Systematic Literature Review." 2024 12th International Conference in Software Engineering Research and Innovation (CONISOFT). IEEE, 2024.

[18] Dincă, Ana-Maria, et al. "AI Tools introduced in Software Development. Analysis of Code quality, Security and Productivity Implications." 2024 IEEE 30th International Symposium for Design and Technology in Electronic Packaging (SIITME). IEEE, 2024.

[19] Baralla, Gavina, Giacomo Ibba, and Roberto Tonelli. "Assessing GitHub Copilot in Solidity Development: Capabilities, Testing, and Bug Fixing." IEEE Access (2024).

[20] Feng, Yunhe, et al. "Investigating code generation performance of ChatGPT with crowdsourcing social data." 2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC). IEEE, 2023.

[21] Grewal, Balreet, et al. “Analyzing Developer Use of ChatGPT Generated Code in Open Source GitHub Projects.” Proceedings of the 21st International Conference on Mining Software Repositories. 2024.

[22] Klemmer, Jan H., et al. “Using ai assistants in software development: A qualitative study on security practices and concerns.” Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security. 2024.

[23] OpenAI, “GPT-4 Technical report,” 2023, arXiv:2303.08774.

[24] Yetiştiren, Burak, et al. “Evaluating the code quality of ai-assisted code generation tools: An empirical study on github copilot, amazon codewhisperer, and chatgpt.” arXiv preprint arXiv:2304.10778 (2023).

[25] Borji, Ali. “A categorical archive of chatgpt failures.” arXiv preprint arXiv:2302.03494 (2023).

[26] Torka, Simon, and Sahin Albayrak. “Optimizing AI-Assisted Code Generation.” arXiv preprint arXiv:2412.10953 (2024).

[27] Pearce, Hammond, et al. “Asleep at the keyboard? assessing the security of github copilot’s code contributions.” Communications of the ACM 68.2 (2025): 96-105.

[28] He, Jingxuan, and Martin Vechev. “Controlling large language models to generate secure and vulnerable code.” arXiv preprint arXiv:2302.05319 (2023).

[29] Taeb, Maryam, Hongmei Chi, and Shonda Bernadin. “Assessing the effectiveness and security implications of ai code generators.” Journal of The Colloquium for Information Systems Security Education. Vol. 11. No. 1. 2024.

[30] Berabi, Berkay, et al. “DeepCode AI Fix: Fixing Security Vulnerabilities with Large Language Models. CoRR abs/2402.13291 (2024), 1–12.” arXiv preprint arXiv:2402.13291 (2024).

[31] Improta, Cristina. “Poisoning programs by un-repairing code: Security concerns of ai-generated code.” 2023 IEEE 34th International Symposium on Software Reliability Engineering Workshops (ISSREW). IEEE, 2023.

[32] Fredj, Ouissem Ben, et al. “An OWASP top ten driven survey on web application protection methods.” Risks and Security of Internet and Systems: 15th International Conference, CRiSIS 2020, Paris, France, November 4–6, 2020, Revised Selected Papers 15. Springer International Publishing, 2021.

[33] Higuera, Juan R. Bermejo, et al. “Benchmarking Approach to Compare Web Applications Static Analysis Tools Detecting OWASP Top Ten Security Vulnerabilities.” Computers, Materials & Continua 64.3 (2020).

[34] Mateo Tudela, Francesc, et al. “On combining static, dynamic and interactive analysis security testing tools to improve owasp top ten security vulnerability detection in web applications.” Applied Sciences 10.24 (2020): 9119.

[35] Spotbugs Documentation Release 4.0.0-RC2 by spotbugs community, https://spotbugs.readthedocs.io/downloads/en/issue-1075/pdf/ Accessed 04 June 2025

[36] OWASP Foundation. (2024). Dependency-Check. Retrieved from https://owasp.org/www-project-dependency-check/

[37] OWASP Foundation. (2022). OWASP Zed Attack Proxy (ZAP) (Version 2.11.1) [Computer software]. Available from https://www.zaproxy.org/.

Beyond the Interface: AI Integration Through Input Stream Mediation and Intelligent Output Simulation

Divij H. Patel
Independent Researcher, Ontario, Canada

***Abstract*—As Natural Language Processing (NLP) technologies continue to advance, their integration within everyday software tools remains fragmented and often constrained by environment-specific plugins or proprietary AI interfaces. This study introduces a lightweight, platform-independent framework that transforms any text-input surface, ranging from simple editors to browsers and full-featured integrated development environments (IDEs), into an intelligent, AI-assisted interface. Leveraging keystroke dynamics and key-chord recognition, the system operates unobtrusively in the background to interpret user intent and enable real-time interaction within active applications. It provides context-aware suggestions, completions, and insights directly within the active window, effectively turning ordinary typing environments into responsive, intelligent companions. Beyond unifying AI support across diverse applications, the framework enables users to seamlessly switch among multiple open-source AI and NLP models via simple key combination triggers, supported through flexible API integration, thereby providing dynamic access to different linguistic capabilities on demand. The architecture also incorporates a protective intelligence layer that detects suspicious behavior patterns and safeguards sensitive information, offering an additional shield against unauthorized data exposure. By employing a universal mediation layer for input and output, supported by controlled keystroke simulation, the approach eliminates the need for custom extensions or application-specific integrations, presenting a scalable model for embedding real-time artificial intelligence into everyday computing environments.**

Keywords*—Keystroke dynamics; keystroke simulation; AI-Assisted Interfaces; open-source AI; key-chord recognition; API integration; real-time interaction systems

I. Introduction

Over the past decade, the artificial intelligence (AI) ecosystem has experienced significant growth in the development of open-source AI and Natural Language Processing (NLP) models made available through well-documented APIs. These APIs facilitate integration into software ecosystems using widely adopted programming languages, thereby enabling AI-driven functionalities across a broad spectrum of applications [1]. Despite these advancements, interaction with such models has largely remained confined to proprietary interfaces, including standalone web portals, dedicated desktop applications, or platform-specific plug-ins. This architectural dependency often forces users to switch between environments to access AI capabilities, resulting in fragmented workflows and diminished productivity.

Recent AI-assisted interaction systems have primarily integrated NLP-based functionalities through application-specific graphical interfaces, browser extensions, or embedded plug-in architectures. While these solutions enable task-oriented assistance such as grammar correction or development support, they remain dependent on platform-level integration and typically require manual configuration for each environment. Consequently, such implementations offer limited interoperability across heterogeneous computing platforms and do not support unified AI-assisted interaction across multiple text-input surfaces.

Additionally, existing research has largely emphasized improvements in model performance or domain-specific deployment of NLP systems, with limited focus on system-level interaction frameworks that enable application-independent AI mediation. Most current solutions rely on proprietary interfaces or tightly coupled API integrations confined to specific software environments. In contrast, the proposed framework introduces a platform-independent architecture that enables real-time AI-assisted interaction across conventional text-based interfaces without reliance on model-specific plug-ins or environment-dependent integrations.

To address this limitation, this study proposes a hybrid architecture comprising three primary components: real-time keystroke monitoring, virtual keystroke simulation, and asynchronous NLP inference over open-access APIs. The proposed framework introduces a lightweight background service that operates independently of application-level interfaces and continuously monitors user input within text-based environments. Upon detecting predefined trigger conditions, such as a hotkey sequence, the captured input is transmitted to an open-source NLP model via an API endpoint, and the generated response is re-inserted into the active application using simulated keystrokes. This mechanism enables interface-independent AI interaction within commonly used applications, including standard text editors such as Notepad and WordPad, as well as embedded terminal environments [2].

Recent developments have attempted to mitigate this limitation through AI-powered browser extensions and application-level integrations tailored for specific use cases. Tools such as grammar correction engines and development assistance environments exemplify this trend toward embedded intelligence. However, these solutions remain tightly coupled to individual platforms or applications and typically require manual installation and configuration for each environment. As a result, users must rely on multiple independent tools for tasks such as grammar correction, paraphrasing, and code analysis, leading to inconsistencies in user experience and limited

interoperability between applications [3]. Existing literature primarily focuses on improving model performance or interface-level integration, with comparatively limited attention given to system-wide interaction frameworks capable of providing application-independent AI assistance.

From a systems perspective, modern operating environments interpret keyboard input through standardized scan-code interpretation and virtual key mapping mechanisms based on established character encoding standards [4]. When a user initiates a keypress event, the keyboard hardware generates a scan code that is interpreted by the operating system and mapped to a virtual keycode corresponding to the active keyboard layout [5]. In addition to capturing real-time keyboard input, contemporary operating systems provide low-level APIs capable of generating synthetic input events through functions such as keybd_event() and SendInput(), thereby enabling the simulation of keystrokes programmatically [6], [7]. These virtual input events are processed by the system kernel in the same manner as physical keystrokes, allowing applications to receive them as standard user input without requiring interface-specific modifications.

Concurrently, the emergence of transformer-based NLP models such as GPT-2, BERT, and spaCy has enabled access to powerful text-generation capabilities trained on large-scale linguistic corpora [8], [9]. These models are typically accessed through graphical interfaces or dedicated applications, necessitating environment switching for interaction. Although RESTful API endpoints allow developers to embed AI functionalities within software applications using languages such as Python or JavaScript, such integrations remain application-dependent and do not provide a unified mechanism for system-wide interaction [10], [11], [12]. This highlights a critical research gap in the development of interface-agnostic AI mediation frameworks capable of operating seamlessly across heterogeneous text-input environments.

The generalized framework eliminates the need for model-specific plug-ins by introducing a system-level layer of AI augmentation that decouples the input source from the NLP interface. This enables context-aware text generation across diverse computing environments while minimizing friction between users and AI-assisted tools. Furthermore, the proposed architecture is scalable to non-traditional text-editing platforms, including online content editors, live-stream overlays, and integrated development environments such as Visual Studio. In such scenarios, user input is intercepted by the background service, processed by an NLP engine, and the generated output is seamlessly appended within the same interface without reliance on third-party extensions [13]. This approach provides real-time assistance while reducing cognitive overhead associated with environment switching.

ANSI escape sequences and virtual terminal protocols form the foundational communication layer of the proposed system, given their widespread use in character encoding, network messaging, industrial control systems, and embedded device scripting. By incorporating AI-driven inference, secure API interactions, and automated keystroke simulation within ANSI-based workflows, the proposed framework advances existing interaction paradigms and introduces new opportunities for intelligent and secure human–computer interaction.

The primary contributions of this work are summarized as follows:

- The design of an interface-agnostic AI mediation framework for real-time interaction across heterogeneous text-input environments.
- A hybrid system architecture integrating keystroke monitoring, keystroke simulation, and asynchronous NLP inference via open-access APIs.
- A system-level AI augmentation layer that eliminates dependency on application-specific plug-ins or proprietary interfaces.
- A scalable implementation capable of enabling context-aware AI assistance within conventional text editors and development environments.

The subsequent Methodology section details the implementation of this framework and its integration within standard computing environments.

II. Methodology

This proposed functionality can be implemented in any modern programming language, such as Python, JavaScript, or C++. Once the main thread starts, which is responsible for continuously monitoring keystrokes and checking for matching key combinations to start and stop the child loops and threads [2]. These child threads can be programmed for different operations, which will be covered in the rest of the study. For example, they might send input to NLP and AI agents, trigger security events, and train existing models for behavior biometrics and typing patterns. These threads can be started/stopped through special key combinations matched via the main thread. Many of the programming flows of the child threads are described in the upcoming sections.

A. Legacy System Foundation and Transition to Intelligent NLP Framework

As previously discussed, the proposed framework aims to offer system-level natural language augmentation by combining open-source NLP models with traditional keystroke-capture techniques. In its initial prototype phase, the system used a multidimensional-array-based string-matching algorithm that provided a deterministic, hardcoded mapping between user input and predefined responses [2]. This is again not a GUI version; it provides a terminal-independent interface that simulates a virtual input/output environment using ANSI escape sequences and raw device file access.

If we see this program in the terminal, the captured keystrokes' ASCII values and converted string look like Fig. 1.

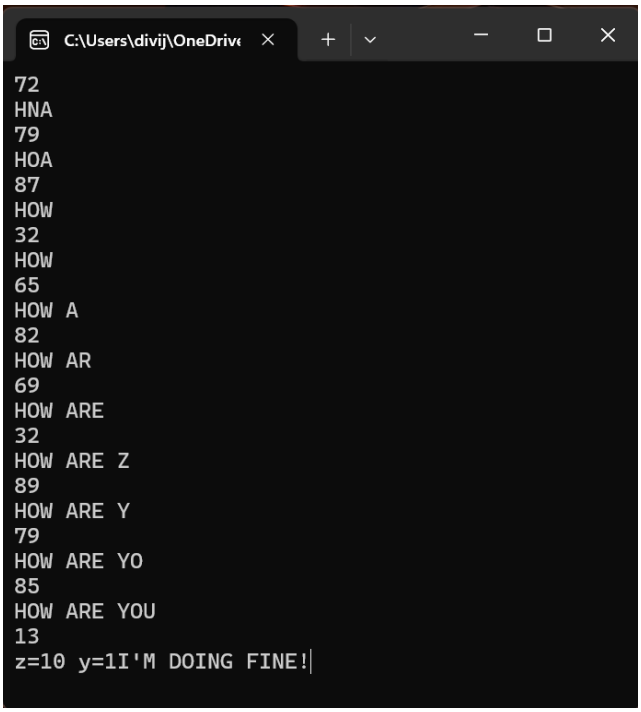


Fig. 1. Converting captured ANSI to text.

When a user enters a key event on the keyboard, such as "How are you doing?", the operating system's built-in module captures it and reads it using a built-in function of a programming language mentioned in section I. Once the keystroke is captured, like the ASCII number "72", the program converts it to the character 'H' (hex 48). It continues capturing and forming sentences until it reads the code "13", which is for the "enter key", indicating the sentence is complete. It then sends this data to either an NLP system or a conventional array-matching loop, which generates the virtual keystroke for the response, such as "I'M DOING FINE!" or whatever reply is received from the NLP server, by triggering the corresponding ASCII numbers in a specific sequence.

This low-level interaction model kept the system non-intrusive by running it in the background without disrupting everyday keyboard-based user actions. Behind-the-scenes control was implemented by custom hotkey combinations that trigger pause, resume, and terminate actions, as mentioned in the study [2].

By monitoring mode, the author refers to maintaining continuous availability of the proposed AI-assistance framework; the system must remain active in the background, independent of user interaction. This persistent operation can be achieved directly in Python and C++ through long-running program structures, as well as through operating system-level tools that support background execution. Modern operating systems provide native service managers, such as systemd on Linux and Windows Services on Microsoft platforms, which are specifically designed to oversee long-running programs, automatically restart them when needed, and ensure they stay active throughout system operation. These service frameworks enable the system to function reliably as a background component, even when no user interface is active [21], [22].

Within the program itself, Python programming language supports persistent background operations via an infinite event loop, asynchronous execution via the asyncio library, and thread-based concurrency via the threading module. These mechanisms enable continuous monitoring, periodic polling, and non-blocking communication with external NLP models. Likewise, C++ provides a rich background-execution model through its standard threading library (std::thread), enabling developers to create long-lived detached threads or asynchronous tasks (std::async) that persist independently of the main application flow. When combined with OS-level service management, these programmatic approaches ensure that the framework behaves as a resilient, always-active process capable of delivering real-time AI support across heterogeneous environments [23], [24].

The current iteration marks a significant architectural shift toward context-aware behavior by incorporating transformer-based NLP models such as BERT [8], GPT-2 [9], and spaCy [12]. These integrations allow for probabilistic, semantically coherent outputs rather than rigid string retrieval, dramatically improving the framework's versatility and scalability.

B. Proposed System Architecture and Operational Workflow

The config file, created during installation, sets up the framework's runtime behavior by defining how NLP endpoints, credential-based access, and linked hotkey connections are enabled. It indicates what actions to take when the key combination is triggered. Similar to file configuration formats, its style is consistent with .env or YAML, with each entry specifying an NLP model endpoint.

This configuration file is also encrypted (using the OpenSSL AES-256 symmetric encryption method) and can be decrypted only by the process in question, which does not compromise the integrity of the sensitive API tokens or service credentials [14]. This encryption strategy ensures confidentiality and integrity in multi-user or shared environments.

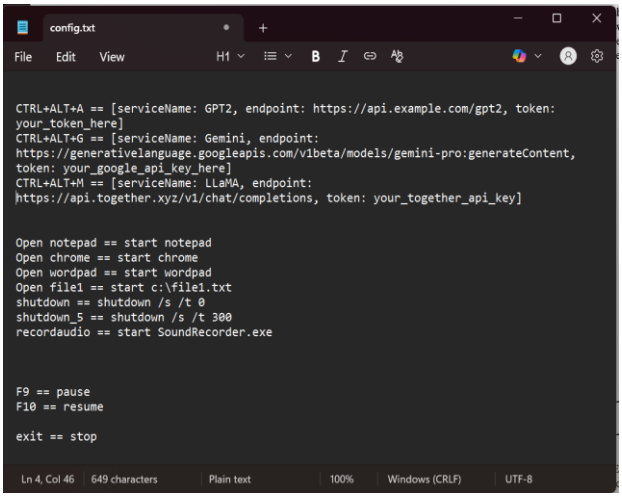


Fig. 2. Config file example.

Fig. 2 shows what the config file looks like when opened. It contains many predefined key combinations, and its system commands are already set for use. The developer can add more or modify the existing ones as needed.

As we discussed, the main thread will continuously check for the hot key combination and, if none are found. It will then generate keystroke-based output from a predefined, updateable array of sentences hardcoded into the program. If the user presses a combination to activate an NLP model, such as CTRL+ALT+A, the main thread spawns a child thread that sends registered keystrokes to the selected NLP model and returns the response by generating virtual keystrokes. Similarly, a user or developer can change the NLP model's preference by pressing a key combination to switch reactions in real time.

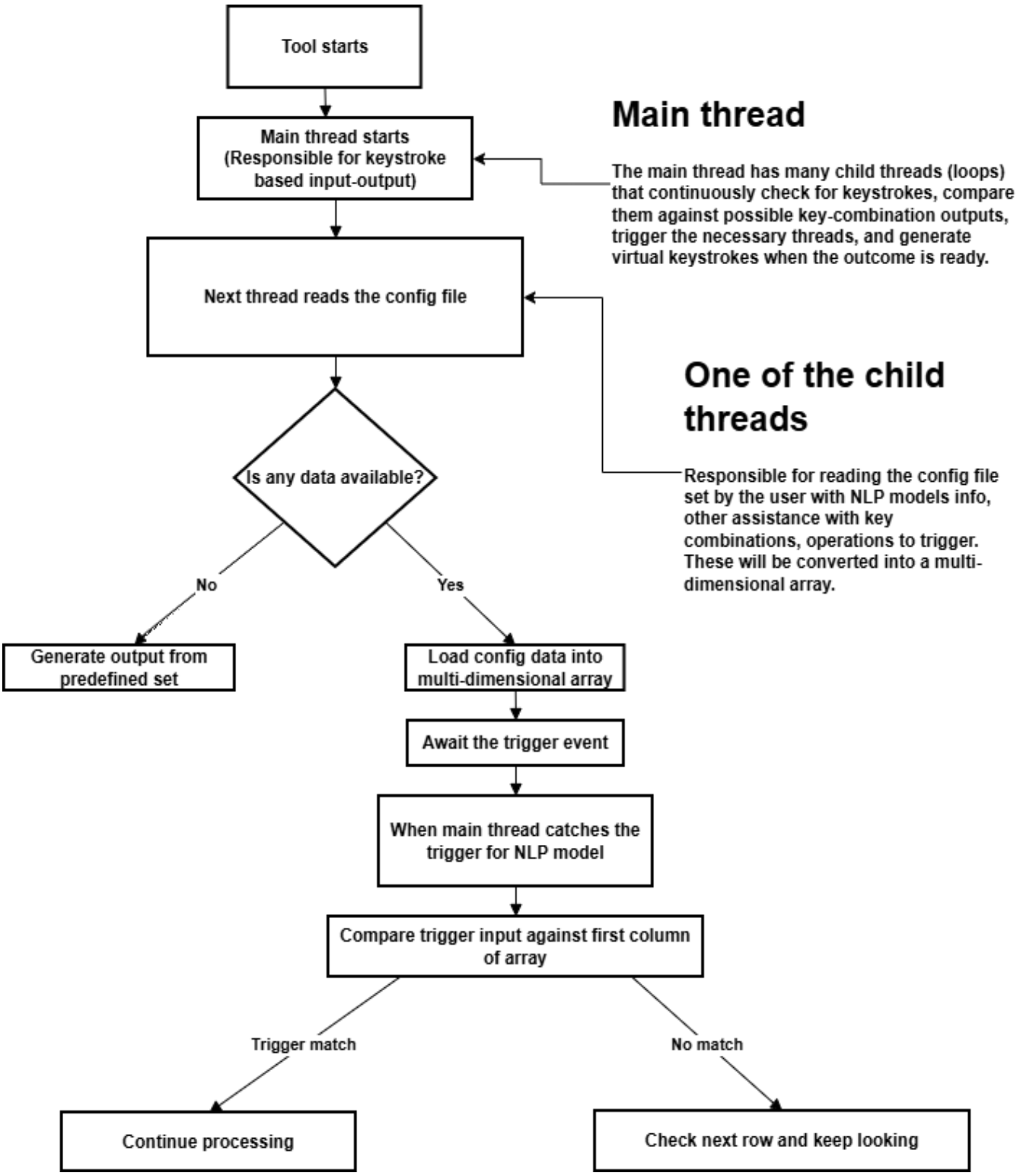


Fig. 3. Switching between NLP models.

Upon detecting a predefined key sequence, the framework executes the following logic pipeline:

1) Event handling and dispatch: A global key listener registers the triggering hotkey and asynchronously parses the associated configuration entry.

2) Secure API invocation: Using Axios (JavaScript) or libcurl (C++), depending on the platform context, a structured POST request is sent to the specified model endpoint, embedding the API token within the Authorization header [15], [16].

3) Response handling: The received NLP model output is parsed and sanitized, then passed to the UI injection engine, which uses virtual keystroke APIs (e.g., Windows SendInput, Linux uinput) to simulate direct text input into the active application window.

4) State management: The framework maintains a lightweight in-memory buffer to cache the last N responses, enabling rollback, regeneration, or comparison functionality within the interaction loop.

The above sequence enables real-time language augmentation across any text-editable interface, whether it be an IDE, browser, email client, or command shell. Unlike monolithic desktop NLP tools, this system features a modular, microservice-based design that supports plug-and-play integration with additional models or downstream services, such as spellcheckers or summarizers.

C. Technical Contributions and Deployment Considerations

1) Hotkeys binding logic: Hotkey binding is a crucial feature of modern operating systems. It enables users to activate specific actions through key combinations, boosting efficiency and workflow. In this work, the researcher has expanded on this traditional concept by employing keystroke-level event interception as an interaction layer for real-time AI assistance. This results in a more flexible and context-aware experience within operating systems. This system has been implemented using Xlib on Linux and SetWindowsHookEx on Windows. It supports dependable key-chord recognition and includes mechanisms to prevent race conditions, ensuring stable and predictable shortcut-event handling across platforms [18].

2) Cross-platform API wrappers: Native bindings to libcurl, along with Axios wrappers for Electron-based environments, enable seamless cloud communication.

3) Error handling: Includes retry logic with exponential backoff and token refresh middleware to maintain long-session stability with external NLP providers. If the retry fails, it automatically generates the appropriate error response and types it via virtual keystroke at the cursor position. The current tool does not have its own GUI (Graphical User Interface). It is crucial to generate proper error codes and display them at the cursor point to ensure clarity for users and developers.

4) Platform abstraction: Layer enables portability across Windows, Linux, and macOS by abstracting system-specific keystroke injection mechanisms. In this study, the author has defined a concept that can be implemented in any preferred programming language and on any operating system.

This design ensures high extensibility, allowing for future integration with on-device LLMs or fine-tuned domain-specific models using frameworks such as Hugging Face Transformers [19]. The following flowchart Fig. 3, illustrates the overall process and operational flow of the proposed framework.

Upon initialization, the Main thread starts and continues to look for the special key combinations and other keystrokes. For example, if the user presses the enter key after some keystrokes, it will be captured and processed as a text-based prompt.

Another child thread accesses a local configuration file through standard read/write operations, enabled by the host operating system's POSIX-compliant filesystem layer. The initialization routine first performs a structural integrity check on the configuration file. If the file is found to be empty or malformed, the system defaults to legacy mode, in which outputs are selected from a hardcoded static array using deterministic logic, as discussed in Section II. Conversely, when valid user-defined entries are detected, the framework parses the data upon receiving the NLP model's matching key combination. It maps it into a three-dimensional array, designated as a[x][y][z], where x indicates the number of configurations, y represents individual data fields (such as hotkey, endpoint, API token), and z specifies the character buffer allocated for each field.

For example, a configuration entry such as:

CTRL+ALT+G == [serviceName: GPT2, endpoint: https://api.example.com/gpt2, token: your_token_here]

Is stored in the array as follows:

- a[0][0]: Hotkey identifier (CTRL+ALT+G)
- a[0][1]: Model name (GPT2)
- a[0][2]: API endpoint URL
- a[0][3]: Authentication token

This data is subsequently referenced during runtime to dispatch model-specific API requests in response to user-defined hotkey events.

In addition to array-based data management, the system supports a modular configuration model that leverages JavaScript Object Notation (JSON) for serialization and deserialization [20]. JSON offers interoperability across multiple programming languages and ensures backward compatibility with web services. This dual-format support, with array-based low-level access and JSON for high-level abstraction, enhances the system's portability and extensibility across diverse environments.

Designed to operate in headless mode, the framework's main thread continuously intercepts and processes keystroke data from the hardware abstraction layer (HAL) without relying on graphical elements. Memorable hotkey sequences, such as CTRL+ALT combined with an alphanumeric trigger key, are intercepted via a low-level keyboard hook. If no special key is

pressed, input is evaluated against a static input-response map (Section II). When a trigger combination is detected, the framework performs a sequential scan through the array index a[x][1] to locate the appropriate service descriptor. Upon a successful match, it retrieves the associated serviceName, endpoint, and token, thereby enabling dynamic model switching.

This design enables users to switch effortlessly between multiple NLP backends (such as GPT-2, BERT, spaCy, and Google Gemini) by simply editing the hotkey configuration file and activating the relevant key combination. The modular and extensible nature of this configuration-based approach not only simplifies deployment but also facilitates real-time inference management across various AI engines.

Once an appropriate Natural Language Processing (NLP) model is selected and configured, the system enters a passive monitoring mode, continuously running in the background to intercept and log the user's keystrokes. The framework uses a low-level keystroke listener, often implemented via OS-level hooks or input stream interception (e.g., SetWindowsHookEx in Windows or XRecord in Unix-based systems) [17]. This enables the system to monitor input events without interfering with the user interface or application foreground processes.

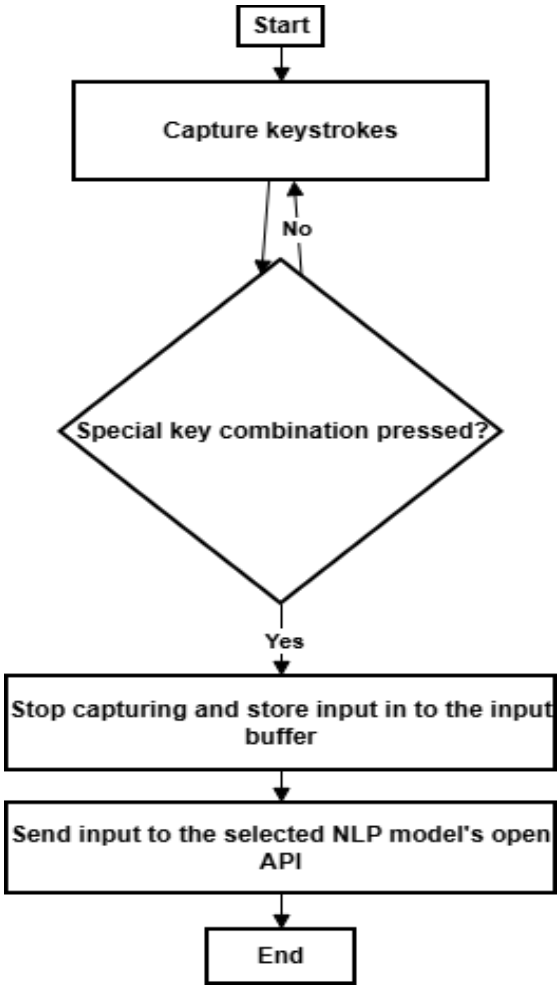


Fig. 4. Send keystrokes to NLP model.

As shown in Fig. 4, the captured string input is transmitted to the selected NLP model endpoint via an OpenAPI-compliant HTTPS request that includes an authentication token to validate the session. These requests are typically handled using lightweight client libraries, such as Axios for JavaScript or libcurl for C++ environments, which support asynchronous communication and robust error handling [15], [16].

The raw answer from the NLP model's inference is presented to the user, and the text is normalized and post-processed. During this step, control characters, special Unicode encodings, and formatting artifacts (such as escape sequences or HTML tags) are discarded to ensure the output conforms to human-readable formatting principles. These changes may be performed using natural language preprocessing libraries or custom sanitization routines to improve the clarity and readability of the output text [25].

After receiving the model's response, the framework dynamically injects the generated output into the user's active application window, using programmatic virtual keystroke generation at the cursor to render the results without modifying the application itself, as shown in Fig. 5.

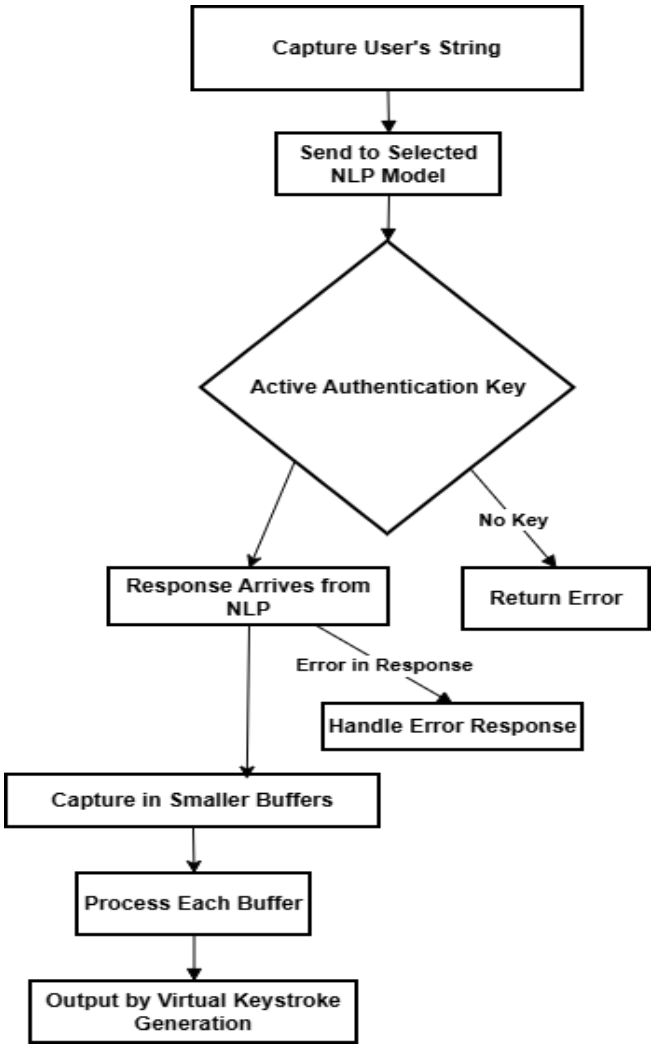


Fig. 5. NLP response to virtual keystroke.

To demonstrate in a real-world scenario, the researcher has simulated an interaction between a user and the GPT-4 model in a standard text editor such as Notepad. Once the program is initialized, users can start typing as usual. The framework remains locally held in memory, capturing all provided written prompts via capturing keystrokes, converting them into a string, and feeding them into the chosen NLP engine. The output is then fetched and formatted via the system and reinserted into the editor as it happens. The generated text is shown below in Notepad, providing both a context-aware interaction experience and an intuitive experience between the traditional input method and modern AI-based language models.

To illustrate the practical implementation of this feature, a scenario is presented in which a user interacts with OpenAI's GPT-4 model via a conventional text editor such as Windows Notepad. After the framework is initialized, it enters a persistent resident mode and operates as a background process. As the user types, the system captures keystroke sequences, processes them as input prompts, and dispatches them asynchronously to the selected NLP endpoint.

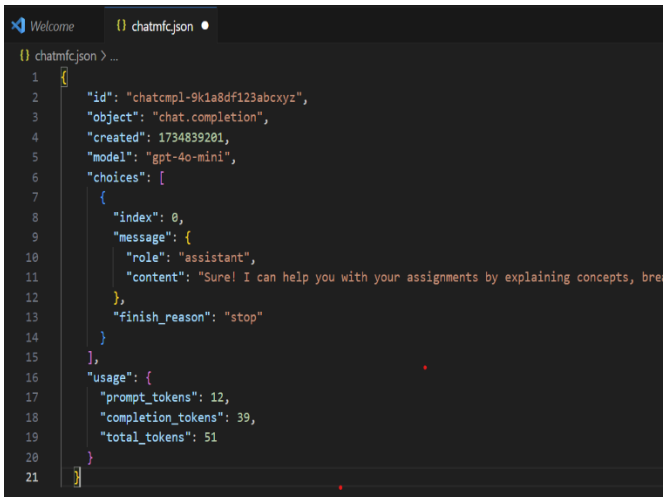


Fig. 6. NLP JSON response to Open-AI.

Upon receiving the model's inference, the framework formats and injects the generated text back into the editor in real time via synthetic keystroke generation. The Fig. 6 shows the JSON response from the NLP server.

Subsequently, the framework triggers a keystroke injection routine using platform-specific input simulation libraries (e.g., Win32 API SendInput) [3]. The output is programmatically written back into the user's active application window, preceded by the ">>" symbol to denote system-generated text. Upon completion of the full response rendering, the system appends a final status marker:

This marker serves as a visual delimiter to indicate the end of model-generated output, enhancing usability and separating AI responses from subsequent user input. In Fig. 7, the NLP response was directly key-stroked in Microsoft Windows Notepad.

A similar result can be achieved using any of the available platforms, such as Microsoft Word, sticky notes (as shown in Fig. 8), in-browser live chat, the Visual Studio Code editor, and many more.

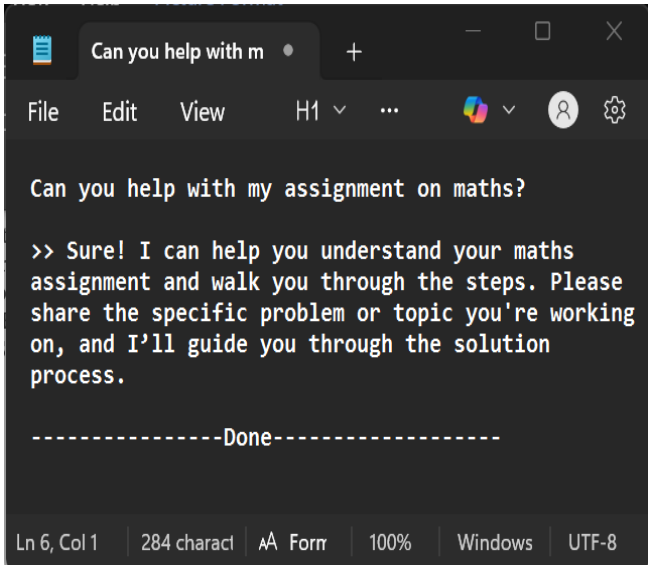


Fig. 7. NLP response in Notepad.

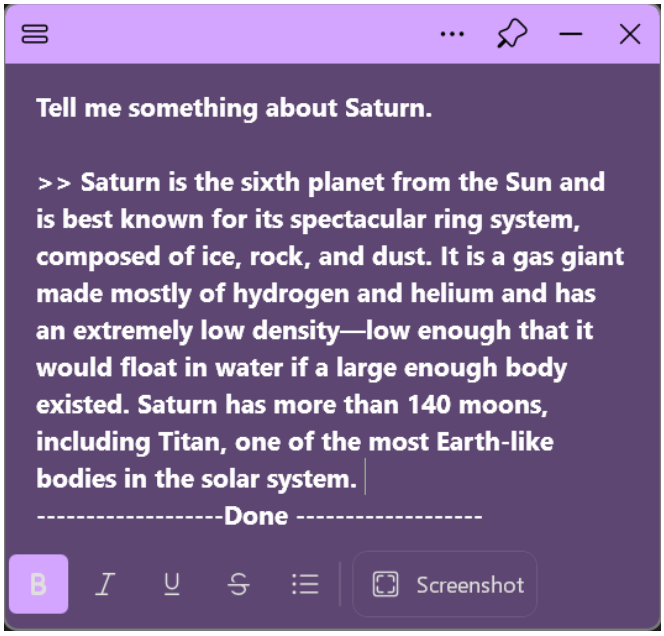


Fig. 8. NLP response in sticky notes.

Similarly, it works with any available text editor, even a browser. The user does not necessarily need to open a text editor; pressing keys on the keyboard can be registered and produce a response at the cursor point.

III. Multi-AI Orchestration via An Interface-Less Model

Within the interface-independent integration framework, you can run these heterogeneous AI assistants in parallel, combining the strengths of various natural language processing (NLP) models and task-specific intelligent agents into a single operational pipeline. Unlike existing NLP engines that focus on context-aware, text-based answers to users' queries, other AI

assistants enable procedural, action-oriented automation. These particular agents can undertake deterministic computations like intelligent scheduling, habit and routine orchestration, multi-calendar synchronization, and workflow optimization, using rule-based reasoning and machine learning-backed event prioritization [26].

By coordinating these capabilities without the constraints of a graphical interface, the system facilitates seamless interoperability, enabling composite AI-driven solutions that extend beyond conversational interaction into fully automated operational ecosystems. Several intelligent personal assistants have been developed in recent years, including OpenAI's ChatGPT [27], Google Gemini [28], Microsoft Copilot [29] and Amazon Alexa [30], Each is leveraging advanced artificial intelligence and natural language processing to interact with users and perform tasks efficiently.

A. Conventional Method to Assist the User

Earlier implementations have incorporated predefined, task-oriented commands to facilitate user interaction in general-purpose computing settings [31]. Available literature presents batch scripting techniques and Windows Command Prompt (CMD) utilities for performing context-specific operations, including opening applications, accessing files, and controlling the system [32]. This capability was implemented using a keyword-to-command mapping schema, in which user-entered input tokens were programmatically matched to shell commands stored in a persistent configuration file (as described in Section 1.2).

For example, the command mapping file may contain entries such as:

Open notepad == start notepad

Open chrome == start chrome

Open wordpad == start wordpad

Open file1 == start c:\file1.txt

shutdown == shutdown /s /t 0

shutdown_5 == shutdown /s /t 300

recordaudio == start SoundRecorder.exe

A user types a command string like "open chrome" into some interface, which the system captures the keystroke stream, parses the first keyword ("chrome"), and searches the mapping file for a corresponding match. The shell command retrieved is then run through the "system()" function of C++ and passed to the operating system's native command processor [33]. Such a methodology enables both immediate (in real time) and delayed (e.g., initiating a shutdown after 300 seconds upon detecting shutdown_5) operations.

In addition to launching applications, it supports auxiliary batch script utilities for hardware-level interfaces, like generating audio/video recording subsystems within Windows operating systems. For example, when RecordAudio is run, SoundRecorder.exe is launched to capture sound for a fixed period. This utility can be used differently depending on the operating system and programming language.

This traditional assistance framework expands its scope to include security-critical automation, enabling it to trigger keystrokes for specific protective actions. As shown in the authors' prototype, threat-detection keywords are stored in the command mapping file. When typed, either intentionally by the owner or accidentally by an unauthorized user, the system activates a security protocol that captures audio and video streams, encrypts or locks private directories, and executes a coordinated shutdown [34], [35]. Such mechanisms provide a low-latency, interface-independent safeguard against potential data breaches by leveraging event-driven command-execution pipelines without requiring GUI intervention.

B. AI Model-based Assistance in One Place

As outlined in Section I, state-of-the-art Natural Language Processing (NLP) models have been successfully implemented to provide domain-specific services across various application sectors [1], [2]. Building on the traditional keystroke-capture-and-trigger method discussed earlier, the framework combines multiple AI-driven service modules into a single, interface-independent environment. This design allows users to access various AI Assistance-based features simultaneously without being limited by application-specific GUIs or platform constraints. [2].

While NLP models primarily focus on generating semantically coherent responses to text-based queries, AI assistance agents are typically trained and configured to execute context-aware actions within browser environments, operating systems, or hybrid computing ecosystems [36]. For example:

1) Motion: An AI-driven scheduling engine for dynamic task reprioritization and meeting rescheduling based on real-time availability constraints [37].

2) Clockwise: A temporal resource optimization system designed to protect focus time and reduce meeting fragmentation [38].

3) Reclaim.AI: A buffer-creation and automated rescheduling assistant for time-block management [39].

4) Google: Workspace-based automation tools for integrated email handling, calendar synchronization, and cross-application task execution [40].

Whilst these tools will each provide a specific functionality in a proprietary or web-based environment, Open APIs (Application Programming Interfaces) offer a standardized path for integration. In the proposed system, API credentials and endpoint configurations are kept in encrypted configuration files (described in Section II). An identification keystroke triggers each AI assistant. The system captures an event when the user enters a pre-defined keystroke sequence, resolves the mapping to the corresponding API call, and transmits the request payload to the assistant's open API endpoint over HTTPS using RESTful API protocols [15], [16].

Direct task orchestration doesn't require an intermediary GUI. For example, typing in the command:

"Send a Thank You email to xyz at 13:00"

would push the corresponding AI assistant API to schedule execution, achieving prompt execution with asynchronous

processing guarantees and secure API key or OAuth 2.0 token authentication [41].

This centralized AI model orchestration framework combines multiple automation pipelines, such as scheduling, communication, and system-level operations, using a unified, low-latency, event-driven command-processing architecture. This utility can be set up in the same way as other commands. For example, the system utilizes AI-assisted scheduling via an open API to automate email scheduling. All it needs is to set up the user's email and other prerequisites. Then, this keystroke-based command can be sent to the open API of the AI assistance, and it will schedule the email [32].

The system enables the currently running AI assistant to schedule email transmission at a specified time using keystroke input. Another example is ordering dinner or telling the assistant to order a pizza from a particular restaurant in a specific size at a specific time. Now, using the method described in Section I, the User can switch the AI assistance model with key strokes and use different assistance services in real time, just like NLP models.

There are several AI assistants available that offer open APIs to help developers utilize them programmatically. Each assistant is specialized and already trained to perform different tasks efficiently. Details for these open APIs can be added to the settings file and easily triggered and used with a single keystroke combination.

To demonstrate the practical use of the proposed framework beyond productivity tasks, consider automating a food delivery request, like ordering a pizza. Usually, this process involves users navigating a vendor's website or mobile app, selecting items, entering delivery information, and completing the payment, or, if using AI assistance, opening their application or website, writing the prompt and details, waiting for the order response, and then switching back to their working window. With an AI assistant integrated via Open API endpoints, these steps can be performed programmatically with minimal user effort. After selecting the appropriate model via a combination of keystrokes, the user can type the prompt anywhere, and it will be executed.

In practice, the system leverages natural language parsing and intent recognition models to capture the user's request. For instance, a keystroke trigger such as:

"Order a large pepperoni pizza with extra cheese from Domino's at 7:30 PM."

Would be intercepted by the framework's command mapping module and mapped to the corresponding Domino's API (or an intermediary food delivery platform API such as Uber Eats or DoorDash. An AI assistance model will do that by itself in the background.

The AI agent translates this natural language request into a structured JSON payload containing attributes such as:

- restaurant_id (e.g., Domino's)
- item_id (e.g., large pepperoni pizza)
- customization (e.g., extra cheese)
- delivery_time (e.g., 19:30 hours)
- payment_token (securely stored OAuth 2.0 or PCI-DSS-compliant token).

The payload is then sent via a RESTful POST request to the vendor's ordering API. Once authenticated and validated, the system sends a confirmation response that the AI assistant may display to the user as a text notification. Then it can be generated at the user's cursor point via keystroke generation. (e.g., "Your pizza order has been scheduled for 7:30 PM").

From a systems perspective, this workflow illustrates that task-oriented AI assistance not only automates systems but also extends to delivering real-world, consumer-facing services. The same event-driven orchestration pipeline, keystroke capture, intent parsing, API mapping, response handling, and so on, simplifies automation of productivity tasks (such as calendar scheduling) and lifestyle tasks (like food delivery), highlighting the versatility and extensibility of the proposed system architecture [42], [43].

IV. Leveraging NLP in the Present Interface-Free System

A. Evolution from Static Mapping to AI-Driven Interaction

The ability of this model to adopt a specific pattern has been a consideration in earlier models, which explored rule-based mappings in which a predefined set of user inputs produces deterministic outputs, and this was the starting point of the proof-of-concept work [2]. Early systems in the field, while limited by scalability, were a good way to demonstrate keyboard-controlled automation of user assistance. Based on the framework mentioned above, this research encompasses next-generation transformer-based Natural Language Processing (NLP) architectures (GPT-2 and BERT, in particular) with open-access APIs and open-source libraries. This transition from hardcoded mappings to context-aware, generative AI architectures massively enhances the adaptability and, hence, robustness of human-computer interactions [8] [9].

Furthermore, the system now supports language inference via tokenized input processing, dynamic prompt engineering, and response conditioning, allowing it to handle open-ended tasks beyond fixed templates. These advancements facilitate the transition from static to semantic-level understanding, aligning the system's capabilities with modern NLP standards [44].

B. Augmented User-Centric Functionalities of MLP-based Frameworks

1) Offline response caching and persistent knowledge storage: One key enhancement in this research is the local caching of AI-generated responses for offline use. This capability addresses a critical limitation of cloud-based NLP models, which typically require uninterrupted internet access. The system enables selective storage of high-utility content, including commonly queried definitions, domain-specific formulas, unit conversions, standardized datasets, and language-dictionary-based questions, in JSON or CSV formats. If the User generated the prompt without internet access, it will look in the cache for similar keywords and create a response

that matches the best at the cursor position. (This procedure for the keyword matching and generating a response is already explained in reference [2].) These cached assets can be accessed via background threads or scheduled jobs, ensuring that offline inference remains feasible even without invoking remote APIs [45].

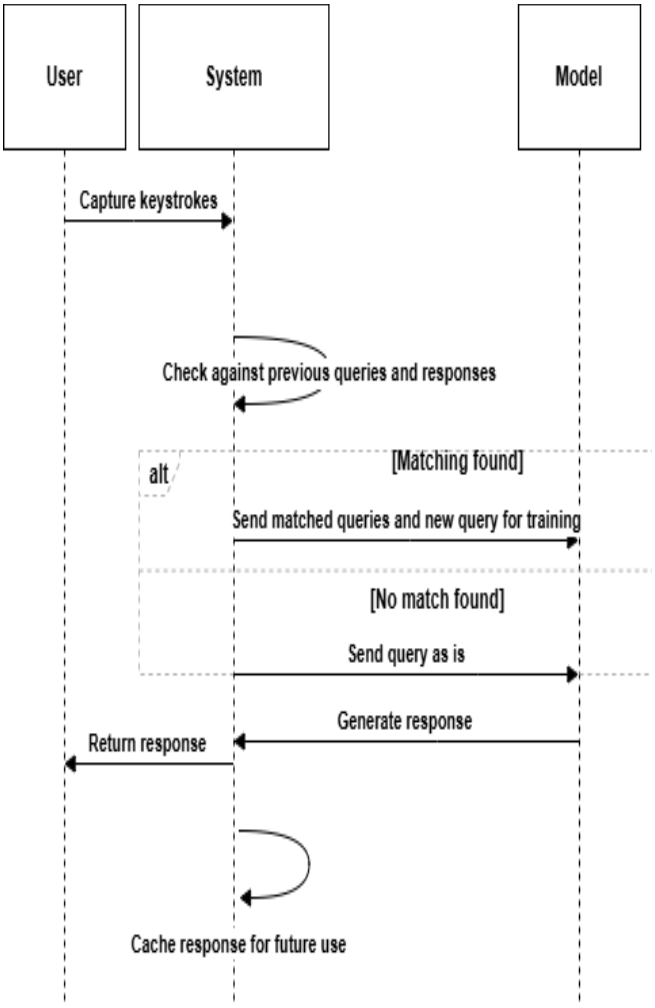


Fig. 9. Cached data to train the model.

This approach aligns with principles of edge AI, where inference and computation are offloaded to local devices to reduce latency and dependency on external networks, thereby enhancing both resilience and real-time performance [46].

2) Using the cached data to train the existing model: As discussed in Section II, when a special keystroke combination is met, the child thread starts the cached base operation. All caching will be stored on the User/Developer's local device and used only with authorized permission to generate a more accurate prompt via connected NLP models. It can be stopped at any time by pressing the special key combinations again.

It can be stored in the same format as a multi-dimensional array mentioned in Section II above, or in JSON format, depending on the programming language selected to develop the tool. Still, the flow of the operation will be the same.

Once the user starts using the tool, it begins capturing their keystrokes. When the hotkey is pressed, it registers the sequence of keystrokes as query sentences. It can submit queries to different open-source AI models and receive responses. As illustrated in Fig. 9, every time a query is made by pressing the Enter key, it is checked against previously cached questions (Only after the user activates the cache-based system via a special key combination, which is set at "start cached" and "stop cached", later users can change it via config file) and responses; if the sentence structure or keywords match, it is noted that this may be a similar request. (sentence matching algorithm is provided in [2]). Then the previous response will be sent along with the query to train the specific model to generate a response that matches the last successful pattern. If no match is found, the query will be sent to the open-source API as-is.

Once the framework's thread finds a matching cache activation combination, it will activate the program's other child thread(loop). This thread is responsible for matching strings from the previous response, sending the matched cache along with the new request, and storing the input and output. If the user presses the same combination again, the thread will stop this loop, and no cache will be sent or stored from that point.

3) Cross-application language support and system-wide paraphrasing: The framework also proposes application-agnostic language support, including real-time spelling correction, semantic synonym generation, language translation, and context-sensitive paraphrasing. Conventional implementations are for word processors (e.g., Microsoft Word or Google Docs, typically via plugins or macros). In comparison, this system works in the background at the OS level, where text is intercepted from any active application window and routed through NLP APIs [2], [47]. For example, users who write emails in Thunderbird and take notes in Sublime Text will receive AI-generated paraphrases or vocabulary recommendations without installing software-specific extensions. Like OS-level accessibility services, this architecture ensures uniform NLP augmentation across a broad spectrum of user applications and workflows.

4) Behavioral biometrics and typing pattern-based security: Beyond usability, the proposed system incorporates a lightweight, AI-driven keystroke dynamics monitoring module that passively analyzes a user's typing biometrics in real time. This component captures parameters such as keypress duration, inter-key latency, and frequency patterns to establish a unique behavioral fingerprint [48]. These learned patterns can be frequently used sentences and stored keywords from the cached memory and the config file. The child thread will continue matching the pattern once started via a special key combination. If anomalous input sequences deviate from the learned pattern, a security submodule is triggered, which may execute actions such as user alerts, session locking, intrusion logging, or system shutdown, as per Section III.

This implementation is inspired by research in behavioral user authentication and continuous identity verification, which have been applied in domains such as secure terminals, financial systems, and high-stakes editorial workflows [49]. Integrating

this with an NLP interface enables intelligent, non-intrusive surveillance, adding a security layer that is particularly beneficial to journalists, academic writers, and researchers handling sensitive or original content.

To mitigate the risk of capturing sensitive user inputs such as passwords or confidential credentials, the proposed framework incorporates application-aware filtering mechanisms that identify protected input fields based on standard UI attributes, including masked or secure text-entry indicators. Input interception is selectively suspended when interactions occur within recognized password or secure-entry contexts, ensuring that sensitive data is neither processed nor transmitted to external inference services. In addition to automated filtering, the framework provides a user-controlled pause and resume mechanism that allows monitoring to be temporarily disabled through predefined commands [2] (e.g., typing a pause trigger), enabling users to manually suspend input capture during interactions involving sensitive information or private credentials.

With respect to the behavioral protection layer, the biometric keystroke analysis module operates on anonymized keystroke dynamics features such as typing rhythm and latency patterns without retaining raw textual content. The performance of this layer is influenced by variability in individual typing behavior and environmental conditions, which may result in occasional false-positive or false-negative classifications during anomaly detection. In the current implementation, the biometric layer is intended to function as a supplementary risk-assessment mechanism rather than a primary authentication system. Further evaluation of classification thresholds and adaptive behavioral modeling is identified as an area for future work to improve detection accuracy and minimize unintended behavioral misclassification.

5) Additional applications and future-scalable extensions: To broaden the scope, several novel use cases are proposed or in development:

- AI-augmented Integrated Development Environments (IDEs): Automatic code snippet insertion, real-time explanation of programming constructs, and citation generation using NLP models like Codex or CodeBERT.
- Education Platforms: Adaptive learning assistants that dynamically rephrase explanations or suggest learning resources in response to student queries.
- Legal and Medical Transcription: Real-time conversion of typed input into domain-specific legalese or medical jargon using fine-tuned Large Language Models (LLMs).
- Multilingual Operating Systems: Automatic translation of input text into target languages on-the-fly for bilingual users, with support for right-to-left (RTL) and logographic scripts.

These future extensions illustrate the potential for a modular AI service layer across personal computing environments, where diverse domains can benefit from real-time, context-aware, NLP-powered interactions.

V. Current Limitations and Room for Improvement

Although the proposed method demonstrates how traditional user-selected IDEs can be enhanced with AI-driven intelligence through keystroke-level interactions, some technical issues remain. First, when generating synthetic keystrokes, words with repeated characters (like “small,” which has a double "l") can sometimes be misread by the keystroke emission module. In some cases, the system registers only one keypress event, resulting in outputs like “smal” instead of the correct “small.” This issue occurs due to event coalescing behavior when rapidly dispatching key events, a known challenge in low-level input simulation frameworks.

Second, if the NLP model produces a lengthy or complex response, the generated ASCII sequence might exceed the active processing buffer's capacity. This can cause the buffer to fill up, which may slow performance or, in severe cases, crash the main thread responsible for capturing system events and generating keystrokes. Similar buffer overflow and thread-blocking issues have been observed in asynchronous I/O systems and event-driven input pipelines.

Both of these limitations can be addressed in future versions of the system by implementing a segmented buffering strategy combined with asynchronous queuing algorithms for keystroke storage and controlled output. Using dynamic buffer allocation, non-blocking queues, and per-character acknowledgment techniques can help manage repeated-character sequences effectively and prevent overflow during significant responses.

Although operating system-level input hooks are supported across major platforms such as Windows, macOS, and Linux, their practical implementation differs due to variations in security policies, event-handling frameworks, and permission models. For instance, Windows environments allow programmatic access to input simulation through system calls such as SendInput(), whereas macOS requires the use of Quartz Event Services with additional accessibility permissions for event interception and synthesis. Similarly, Linux-based systems rely on mechanisms such as X11 or Wayland input subsystems, which may impose sandboxing or privilege restrictions. These platform-specific differences can influence the deployment and operational behavior of keystroke monitoring and simulation modules within the proposed framework. A comprehensive cross-platform implementation strategy, accounting for OS-level constraints and permission requirements, remains an area for future investigation to enhance the portability and robustness of the system.

VI. Future Scope

In the proposed framework, users can dynamically switch among multiple AI and NLP models to leverage their distinct functional advantages. Each model offers specialized capabilities optimized for specific linguistic or contextual tasks. Future enhancements aim to incorporate adaptive analytics modules that autonomously select the most appropriate model based on the user’s active application context and ongoing workflow. This adaptive orchestration will enable real-time utilization of heterogeneous AI agents, thereby enhancing overall system efficiency and contextual responsiveness.

Beyond its current capabilities with open-source NLP models, the proposed framework presents significant potential for the development of personalized AI assistants. Future extensions of this work will focus on enabling the assistant to learn individual user behavior, task preferences, and interaction patterns over time. By leveraging adaptive learning techniques, the system can evolve into a user-specific intelligent agent capable of autonomously handling routine daily activities, providing contextual recommendations, and optimizing task execution based on personal workflows.

Moreover, future research will explore the integration of behavior-aware security mechanisms within personalized assistants. By modeling normal user interaction patterns, the system can detect anomalous activities and initiate preventive actions, thereby strengthening protection against unauthorized access and data misuse. Such advancements will transform the framework into a comprehensive, user-centric AI ecosystem that combines intelligent automation, personalization, and proactive security.

VII. Conclusion

The proposed interface-agnostic AI integration framework allows diverse NLP models and AI-based assistants to work smoothly without any traditional graphical user interfaces. It demonstrates how intelligent automation can be integrated into user workflows through low-level input monitoring, standardized communication protocols, and secure API orchestration. This minimizes reliance on platform-specific plug-ins or standalone tools to optimize task execution and user productivity. Moreover, AI-assisted decision-making with context-aware keystroke interpretation introduces a new approach to adaptive computing that learns user intent and automatically summons the appropriate AI model for a given task. This architecture is modular, secure, and scalable enough to support ever-evolving AI ecosystems and open APIs. It's possible to further expand this framework with improved cross-device synchronization, advanced behavioral analytics, and self-optimizing task delegation to make it accessible across enterprise and personal computing environments in future iterations. This work adds a valuable insight to the ongoing evolution of intelligent human–computer interaction by bridging traditional input processing with contemporary AI-driven automation.

References

[1] J. Z. K. L. P. L. Pranav Rajpurkar, "SQuAD: 100,000+ Questions for Machine Comprehension of Text," Association for Computational Linguistics, p. 2383–2392, November 2016.

[2] D. H. Patel, "Real-time Interactive and Artificial Intelligence," International Journal of Engineering and Advanced Technology (IJEAT), vol. 8, no. 3, pp. 179-183, 2019.

[3] A. U. H. H. S. M. B. R. A. M. A. S. H. J. H. B. O. B. S. L. Jamil Hussain, "Model-based adaptive user interface based on context and user experience evaluation," Journal on Multimodal User Interfaces, pp. 1-16, 2018.

[4] E. a. T. H. Wiki, "Milestones: American Standard Code for Information Interchange (ASCII), 1963," ETHW, 2016.

[5] I. Corporation, "IBM Personal Computer Technical Reference," IBM, August 1981. [Online]. Available: https://bitsavers.org/pdf/ibm/pc/pc/6025008_PC_Technical_Reference_Aug81.pdf?utm_source=chatgpt.com. [Accessed 14 November 2025].

[6] Y. Yang and L. Li, "Turn Smartphones into Computer Remote Controllers," International Journal of Computer Theory and Engineerin, vol. 4, no. 4, pp. 561-564, 2012.

[7] M. Corporation, "Keyboard Input Overview," Microsoft, 2025.

[8] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in NAACL-HLT 2019, 2019.

[9] OpenAI, "GPT-2: Language Models are Unsupervised Multitask Learners," OpenAI, San Francisco, CA, 2019.

[10] M. Neumann, D. King, I. Beltagy and W. Ammar, "ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing," in Association for Computational Linguistics (ACL), Florence, 2019.

[11] M. N. Group, "MITIE: MIT Information Extraction," 03 February 2015. [Online]. Available: https://github.com/mit-nlp/MITIE. [Accessed 22 September 2025].

[12] S. Contributors, "spacy-nlp (Node.js wrapper for spaCy)," NPM, 2018.

[13] J. D. Weisz, S. Kumar, M. Muller, K.-E. Browne, A. Goldberg, E. Heintze and S. Bajpai, "Examining the Use and Impact of an AI Code Assistant on Developer Productivity and Experience in the Enterprise," arXiv, 2024.

[14] O. S. Foundation, "OpenSSL Cryptography and SSL/TLS Toolkit," OpenSSL Software Foundation, 2025. [Online]. Available: https://www.openssl.org/.

[15] A. Contributors, "Axios – Promise based HTTP client for the browser and node.js," Axios, 2024.

[16] T. C. Project, "libcurl – client-side URL transfer library," The curl Project, 2024. [Online]. Available: https://curl.se/libcurl.

[17] M. Corporation, "SetWindowsHookExA function (winuser.h)," Microsoft, 2023.

[18] X. Foundation, "Xlib – C Language X Interface," X.Org Foundation, 1985.

[19] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu and C. Xu, "Transformers: State-of-the-Art Natural Language Processing," in Association for Computational Linguistics (ACL), Online, 2020.

[20] E. International, "The JSON Data Interchange Syntax, ECMA-404, 2nd ed.," ECMA International, 2017.

[21] T. L. Foundation, "systemd.service — Service unit configuration," 2015–present. [Online]. Available: https://www.freedesktop.org/software/systemd/man/systemd.service.html.

[22] Microsoft, "Service Applications," 2025. [Online]. Available: https://learn.microsoft.com/en-us/windows/win32/services/service-applications.

[23] P. S. Foundation, "asyncio — Asynchronous I/O," Python 3 Documentation, [Online]. Available: https://docs.python.org/3/library/asyncio.html.

[24] I. C. Foundation, "std::thread — C++ standard threading library," [Online]. Available: https://en.cppreference.com/w/cpp/thread/thread.

[25] S. Bird, E. Klein and E. Loper, Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit, Sebastopol, CA: O'Reilly Media, 2009.

[26] S. M. H. Zaidi, R. Iqbal, A. Alharbi, H. H. Zuberi, A. Ahmed and M. U. Sheikh, "Intelligent Process Automation Using Artificial Intelligence to Create Human Assistant," International Journal of Information Technology and Decision Making, vol. 24, no. 5, pp. 1234-1256, 2025.

[27] OpenAI, "ChatGPT: Optimizing Language Models for Dialogue," OpenAI, San Francisco, CA, 2023.

[28] G. Developers, "Gemini is now accessible from the OpenAI Library," Google, Mountain View, CA, 2024.

[29] Z. Khan, "Microsoft 365 Copilot APIs: Unlocking enterprise knowledge for AI with the Retrieval API," Microsoft 365 Developer Blog, 2025.

[30] M. Yoshitake, "Introducing AI native SDKs for Alexa+," Amazon.com, Inc., 2025.

[31] B. W. Kernighan and R. Pike, "The UNIX Programming Environment," Englewood Cliffs, NJ, Prentice-Hall, 1984, pp. 71-99.

[32] D. Hennig, "Windows PowerShell: Batch Files on Steroids," Saskatoon, Stonefield Software Inc., 2019.

[33] Cppreference.com, "std::system," C++ Reference Documentation, 2024. [Online]. Available: https://en.cppreference.com/w/cpp/utility/program/system.

[34] D. H. Patel, "Spyware Triggering System by Particular String Value," International Journal of Engineering Research and Development, vol. 11, no. 09, pp. 32-36, 2015.

[35] A. Somayaji and S. Forrest, "Automated Response Using System-Call Delays," in IEEE Symposium on Security & Privacy, Berkeley, CA, 2000.

[36] H. Xueyu, X. Tao, y. Biao, W. Zishu , X. Ruixuan, C. yurun, y. Jiasheng, T. Meiling , Z. Xiangxin, Z. Ziyu, L. Yuhuai, X. Shengze, W. Shenzhi, X. Xinchen , Q. Shuofei, W. Zhaokai, k. Kun, Z. tieyong, W. Liang, l. Jiwei, j. Yuchen Eleanor, Z. Wanchunshu, W. Guoyin, Y. Keting, Z. Zhou, Y. hongxia, w. Fan, Z. Shengyu and w. Fei, "OS Agents: A Survey on MLLM based Agents for General Computing Devices Use," arXiv, 2025.

[37] Motion, "AI-Powered Calendar and Task Management Platform," Motion, Inc., 2025.

[38] Clockwise, "AI-Powered Time Management Calendar," Clockwise, Inc., 2025.

[39] Reclaim.ai, "AI Calendar for Work and Life," Reclaim.ai, 2025.

[40] Google Workspace, "Automation and Integration Tools for Google Workspace," Google Workspace Marketplace, 2025.

[41] D. Fett, R. Küsters and G. Schmitz, "A Comprehensive Formal Security Analysis of OAuth 2.0," arXiv, 2016.

[42] O. Kalkan, "Event Driven AI Workflows: Powering Real Time Automation," Gibion.ai Blog, 2025.

[43] IBM, "What is AI Orchestration?," IBM Think, 2025.

[44] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), 2019.

[45] T. Wolf, V. Sanh, L. Debut, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz and J. Brew, "Transformers: State-of-the-Art Natural Language Processing," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP): System Demonstrations, 2020.

[46] H. Lin and S. Zhu, "Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing," Proceedings of the IEEE, vol. 107, no. 8, pp. 1738-1762, 2019.

[47] Y. Hu, X. Jing, K. Youlim and R. T. Julia, "Misspelling Correction with Pre-trained Contextual Language Model," arXiv, 2021.

[48] P. S. Teh, A. B. J. Teoh and S. Yue, "A Survey of Keystroke Dynamics Biometrics," The Scientific World Journal, vol. 2013, 2013.

[49] F. Monrose and A. D. Rubin, "Keystroke Dynamics as a Biometric for Authentication," Future Generation Computer Systems, vol. 16, no. 4, pp. 351-359, 2000.

AUTHOR'S PROFILE



DIVIJ H PATEL was born in Nadiad, Gujarat, India, in 1995. He received the B.E. degree in Computer Engineering from Gujarat Technological University, Gujarat, India, in 2017. He has pursued the postgraduate certification in Information and Communication Technology Solutions for Small Businesses at Confederation College, Thunder Bay, ON, Canada.

During his undergraduate studies, he conducted research in several areas of computer science and engineering. He has published three research papers in international journals, including "Spyware Triggering System with the Particular String Value" and "Multipurpose Reminder System" in the International Journal of Engineering Research and Development, and "Copy-Paste Command with Additional Facilities" in the International Journal of Computer Applications, New York, USA.

He has filed three patents with the Indian Patent Office covering innovations in wearable devices and human–computer interaction. His patented works include the Smart Wearing Cap Device, a method for Assigning Shortcut Keys, and the Folding Calculator. His research and innovations have also been featured in newspapers and television media such as Gujarat Samachar, Sandesh News, Vichar Kranti News, DNN Link News Channel, and VTV News.

Best Practices to Train Accurate Deep Learning Models: A General Methodology

Alberto Nogales*, Ana M. Maitín, Álvaro J. García-Tejedor
Center for Studies and Innovation in Knowledge Management (CEIEC), Francisco de Vitoria University, 28223 Pozuelo de Alarcón, Spain

***Abstract*—In recent years, the field of computer science has experienced great changes due to the incredible advances in the field of artificial intelligence. Deep Learning models are responsible for most of them since the biggest milestones occurred in 2012 when AlexNet won the image classification challenge called ImageNet. These models have demonstrated great performance in different types of complex tasks like image restoration, medical diagnosis, or object recognition. Their disadvantages are related to their high data dependency, which forces experts in the field to follow a precise methodology to obtain accurate models. In this study, we describe a complete workflow that begins with the management of the raw data until the in-depth interpretation of the performance of the models. This should be taken as a high-level consultation document describing good practices that should be applied. Apart from the step-by-step methodology, we present different use cases that correspond to the two main problems of the field: classification and regression.**

***Keywords*—Methodology; artificial intelligence; deep learning**

I. Introduction

If we talk about what has been one of the biggest advances in recent history, we must talk about the revolution of artificial intelligence (AI). AI is the domain within computer science that analyzes and interprets the processes underlying intelligent actions in humans, aiming to replicate these actions in machines, not always employing the same mechanisms [1]. These advancements have been fostered by two main factors. First, the creation and availability of large amounts of data have never been seen before. Second, the affordable price of hardware to compute this data and train the AI models.

But if we must talk about the milestone in the modern AI era, we must talk about Deep Learning (DL) models. This is a family of models whose term was coined in 2006 by Hinton. They are defined as multiple-layer models using hierarchical Artificial Neural Networks (ANNs) with the capacity to progressively learn data representations, beginning with the input data and advancing to higher levels of abstraction [2]. Its biggest milestone occurred in 2012 when AlexNet was the first DL model that won an image classification challenge called ImageNet [3].

Deep Learning models have been demonstrated to have high performance in solving many different complex tasks. Nevertheless, some weaknesses face any expert in the field constantly, such as poor interpretability, data dependency, or lack of generalization in a broad range of datasets. The last two, apart from being interrelated, are common to all the implementation workflows of these models. Normally, large amounts of data are needed to train accurate models, but there is also a need to generalize and avoid overfitting. The latter is described in [4] as the situation in which the model learns characteristics associated with noise or variability within the data, rather than the fundamental distribution of the training data. So, avoiding overfitting is the main condition to obtain a model that performs well generalize.

Overfitting is tested at the end of the process of model training. At this point in the process, different measures of a metric are compared, corresponding these values to the different sets into which the data in origin is split. The success of the training process depends on different decisions to be taken, which are related to the type of problem to be solved, the nature of the data, and other conditions. The large number of decisions to take and the lack of knowledge about what they depend on have allowed the publication of papers where the overfitting of the models is noticeable. The motivation of this study is to compile and explain step-by-step how to perform accurate processes to train Deep Learning models.

The work aims to provide resource information for researchers so they can know which steps to apply and understand them. This will be very useful for scientists without a deep understanding of deep learning terms. In the case of experts in the field, they could have a guide of best practices to follow. In this sense, the main contribution is to compile in a single document the explanation of each step. It will not only describe the method step-by-step, but we will also provide why each step is necessary and how to apply it depending on several conditions, such as the use case to solve or the type of data. As we want this study to be accessible to any researcher interested in the field, we are providing a high-level explanation of terms, strategies, etc., with only basic mathematical formalization.

The rest of the study is organized as follows: Section II describes all the antecedents of the DL field, so a general idea of different terms could be obtained, with the possible applications DL could have. Section III explains step-by-step the proposed methodology to obtain accurate models. Section IV shows different use cases where the methodology is applied. Finally, Section V compiles different conclusions related to what has been explained in the work.

II. Antecedents

DL models are very diverse, with their performance tied to the nature and quality of the training data. This section explores the different models used in the field. Fig. 1 shows the hierarchical organization of these models depending on the way

*Corresponding author.

they learn. An in-depth description of the models will be given later.

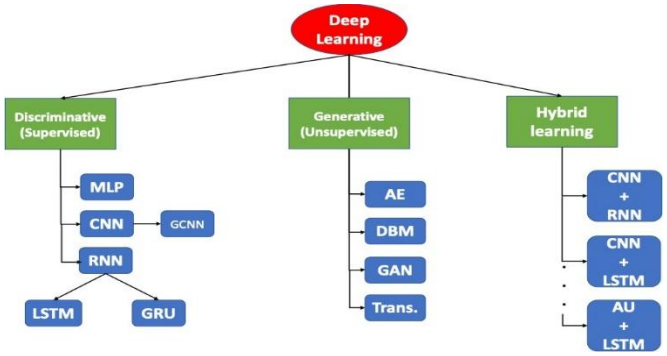


Fig. 1. Classification of Deep Learning models based on their different training strategies.

DL models are first categorized based on their approach to learning from data (training stage), which includes supervised and unsupervised methods. The first category corresponds to those using labelled data for training, where the model understands the relationship between input data and expected output, which is a discriminative task.

The Multilayer Perceptron (MLP) represents the most basic form of a DL model. Its architecture consists of an input layer, multiple hidden layers, and an output layer. It is typically used to predict a particular numeric value, but classification problems can also be solved, for example, [5]. Take the benefit of using MLPs to predict the density of 48 refrigerant systems.

Convolutional Neural Networks (CNNs) stand out as one of the most used models, particularly in computer vision applications [3]. Their primary strength lies in their capability to detect patterns in a spatially invariant manner. This characteristic enables them to recognize specific structures in an image, regardless of their location. More recently, there has been a development of a specialized type of CNN known as Graph Convolutional Neural Networks (GCNNs). Introduced by [6], GCNNs are designed to process graph-structured data. This model encodes both the structure of a graph and the features of its nodes. Examples of the application of these models can be found in the following papers [7], where CNNs are used for the automated assessment of damage type and severity in wooden structures [8], and a GCNN model is used for identifying essential genes in organisms.

Recurrent Neural Networks (RNNs), as defined by [9], employ an input vector of variable length and recursively apply a transition function to their internal hidden state vector h_t. They are particularly useful for time series data structures. Within the family of RNNs, there are two specific subtypes known as Long Short-Term Memory (LSTMs) and another called Gated Recurrent Units (GRUs). LSTMs, introduced by [10], were devised to effectively handle noisy or ambiguous input data. The study [11] is an example where RNNs are used for the task of stock prediction.

The other prominent set of models falls under the unsupervised training category. In this scenario, the data has no labels, and there is no prior knowledge regarding the results of performing generative tasks [12].

Deep Autoencoders (DAEs) apply unsupervised learning. First described by [13], they are characterized by having input and output layers of the same or similar size, along with two processing modules. The first structure, the encoder, takes the input data and compresses it to a smaller representation containing its essential features. The second structure, the decoder, aims to reconstruct the original input data by upsampling the compressed representation until it reaches the size of the input data.

Restricted Boltzmann Machines (RBMs) are regarded as a specific form of Autoencoder, initially proposed by [14], capable of learning probability distributions. RBMs have been employed to construct Deep Boltzmann Machines (DBMs), as stated by [15]. In [16], Autoencoders are used for segmenting teeth in Intra-Oral 3D Scans.

A particular and very well-known Autoencoder is Transformers, which are models that avoid recurrence and exclusively rely on an attention mechanism to establish global dependencies between input and output [17]. This type of model is used by [18] for the prediction of risk in cardiovascular diseases.

By combining the previous learning types, it creates a new category known as semi-supervised learning. Within this classification fall Generative Adversarial Networks (GANs). GANs comprise two neural models: the generator and the discriminator, operating in an adversarial training paradigm [19]. This architecture aims to understand and replicate a given data distribution. The generative model tries to generate synthetic instances of the input data, while the discriminator assesses these instances to determine their similarity to the input data. This adversarial process provides a probability of authenticity for each instance, distinguishing between input (authentic) and synthetic (generated) data. Through iterative refinement, the generator learns to produce data that increasingly resembles the input data. In [20], the authors implement a GAN model to translate RGB images into IR ones for fire forest monitoring.

In recent years, a trend has emerged within the field of deep learning: the development of hybrid models. These models are considered important for future advancements in DL. They combine two or more distinct architectures, such as CNN-LSTM or Autoencoder-LSTM, [21] and [22].

III. Introduction Methodology to Train Deep Learning Models

In this section, the methodology to obtain accurate DL models is described in depth. The process has been divided into different steps, adding the best practices and an explanation about how they should be applied. It begins with having the dataset raw related to a use case to solve with it. Then, deciding which DL model fits better until the obtention of a set of accurate metrics that avoid generalizes. Finally, ending with an interpretation of the models using other metrics and making a comparison with some baselines.

A. Deciding What Deep Learning Model to Use

The first decision to be taken when solving a DL problem corresponds to choosing a suitable model. Considering what has

been described in Section II, the different DL models that can be applied depend on the nature of the data and the use case to be solved. If we talk about the different use cases, there are also many applications, but all of them can be classified into two main types: classification and regression. Belonging to one group or the other will have consequences, as will be described later, in the metrics used to measure the performance of the model. Classification problems consist of building a model that links the attributes of the instances, typically represented as attribute-value pairs, to the corresponding class labels [4]. Regression problems aim to predict the value of a dependent variable based on one or more independent variables, [4]. Typical classification problems in DL are image classification or disease diagnosis. Within the regression problems group, we have time series prediction and object detection.

Due to the wide range of models and use cases, this decision has multiple solutions. We provide a general vision about how to choose the proper DL model. Table I compiles information on the different models, the type of data used to train them, and the typical use cases to solve.

TABLE I. SUMMARY OF DL MODELS USED, TYPE OF DATA AND APPLICATIONS.

Model type	Nature of data	Use case
MLP	Tabular data	Prediction of a continuous value
CNNs	Image data with spatial relationships	Image Classification, Object Detection, Segmentation
GCNNs	Data represented as graphs or networks	Graph-structured Data, Social Network Analysis
RNNs	Temporal sequences, sequential data	Sequential Data Modelling, Time Series Prediction
Autoencoders	Unlabelled images	Data Compression, Denoising, Anomaly Detection
Transformers	Sequential data, particularly text data	Language Translation, Text Generation
GANs	Often images, but can be applied to other domains	Image Generation, Style Transfer, Data Augmentation
DRL	Sequential decision-making tasks with rewards	Game Playing, Robotics, Autonomous Systems
Hybrid models	Mixed data types or tasks	Multiple types of data/modalities or tasks

B. Splitting the Dataset into Train, Validation, and Test

As has been said before, to obtain a Deep Learning model that performs well, we need to avoid overfitting. Model testing, which consists of checking how a trained model performs with new data never seen before, is the way to prevent it. For this purpose, first, there is a need to split the data into three sets: training, validation, and test.

The training set shows instances of the model so it can be adjusted to generalize its distribution. The validation set measures the accuracy of the model based on the errors obtained. The test set is used to check the capacity of the model to generalize by using data never seen before.

So, the first decision to be taken is the percentages used to create the three sets. A typical criterion is based on the Pareto principle, which states that for many phenomena in nature, about 80% of the consequences are produced by 20% of the causes [23]. In this way, a good choice is to split the training and test by 80% and 20%, respectively. Then, split the training set into 80-20 for training and validation. Anyway, this is not an exact rule, and depending on the size of the original dataset, these percentages could vary. For small datasets, it could be 70-30, and for big datasets, 90-10, as done in [24].

The second decision to take related to the training, validation, and test sets is the strategy to follow during the split. Typically, a randomization is applied to avoid patterns depending on the order of the data. This prevents potential biases by ensuring that there is enough diversity representing the complete dataset in the three subsets. However, depending on the type of data and its organization, randomizing it is not always the best option. It is very important to avoid instances from the test set being introduced during training, which is called data leakage. A typical data leakage case occurs when using frames of videos in a model. If we split the sets of subsequent frames, almost equal frames could be part of the training and test sets. To avoid this, it is better to split the dataset into full sequences.

Finally, a problem could occur in classification problems with imbalanced classes. This could lead to a training or test set where a class has more presence compared to the other. The strategy that is applied to avoid this is called stratification. This will ensure that each subset maintains an identical class distribution.

C. Feature Scaling

Apart from the obvious transformation of data into continuous values, as input data in DL models is always numeric data, feature scaling is a best practice to apply just before starting to train the models. This term is defined in [25] as transforming all features to a uniform range, avoiding the influence of large-scale features on classification models or other feature-dependent processes. Feature scaling also has different benefits. For example, they can accelerate the process of gradient-based optimization algorithms, as in DL models, which leads to faster training times. Also, by avoiding the influence of features over the rest, the model performance tends to be improved. Finally, calculating distances is more accurate after feature scaling, as this process is part of the algorithms used in DL, for example, L1 regularization.

In this sense, there are two main strategies: scaling and normalization. The former transforms features to have a mean of zero and a standard deviation of one, which prevents certain features from dominating others during training. The latter scales features to a range between 0 and 1 or -1 and 1, which is particularly useful for algorithms sensitive to the magnitude of features, such as distance-based algorithms. There are many different strategies to apply scaling and normalization, but the following should be highlighted.

In the case of scaling, most well-known is Min-Max. This strategy rescales the features to a fixed range, typically [0, 1], by subtracting the minimum value from each feature and then dividing by the range (the maximum value minus the minimum value). Eq. (1) formalizes this process:

$$X_{MinMax} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

The main strategies in normalization are Euclidean Norm and Z-Score. Z-score transforms the features to have a mean of 0 and a standard deviation of 1. This strategy is depicted by Eq. (2):

$$Z_{Score}=\frac{X-\mu}{\delta} \quad (2)$$

In the equation, μ is the mean, and δ is the standard deviation.

D. Choosing the Metrics Depends on the Problem to Solve

The way to measure the performance of a DL model is by using metrics. The aim is to obtain the values that ensure that the model is well-trained. As a first step, we must choose which generic metric to use depending on the problem to solve. As we have said before, DL problems can be framed into two big classes: classification and regression.

In classification problems, the main metric is called accuracy. This is defined as the proportion of the samples that the model has predicted correctly among the total number of instances. Eq. (3) describes this metric:

$$Accuracy=\frac{TP+TN}{TP+FN+FP+TN} \quad (3)$$

In the equation above, True Positive (TP) refers to cases where the actual sample class is positive, and the model's prediction is also positive. True Negative (TN) denotes instances where the actual sample class is negative, and the model's prediction is also negative. False Positive (FP) signifies scenarios where the actual sample class is positive, yet the model predicts a negative outcome. False Negative (FN) represents situations where the actual sample class is negative, but the model predicts a positive result. The accuracy measures a percentage in ranges of 1, so the best value is the nearest to 1.

For regression problems, the Mean Squared Error (MSE) is usually used. It is defined in [26] as the mean of the squared differences between the predictions and the actual values. The following Eq. (4) describes this metric:

$$MSE=\frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2 \quad (4)$$

In the previous equation, N corresponds to the number of instances of the dataset, yi is the real value, and $\hat{y}i$ is the prediction made by the model. When using the MSE, the aim is to obtain a value as close to 0 as possible.

These metrics will be used as a first step to measure the performance of the model during training, validation, and test, or check if underfitting or overfitting is happening. For a deeper analysis of how models work, we will use other metrics that will be introduced in Section III H.

E. Using k-Fold Validation During the Training Stage

K-fold validation is a typical strategy used when data is scarce. Apart from that, it should be used to validate the models more exhaustively. As DL models are initialized by randomization of their weights, this initialization could benefit from the data split in how the model performs. To mitigate this issue, k-fold cross-validation is employed.

In this strategy, the training set is divided into k smaller subsets or "folds". Subsequently, a loop is executed for each of the k folds. Then, the model is trained using k-1 folds as the training data and validated (evaluated for its performance) using the remaining fold. The numeric measure reported by k-fold cross-validation is the average of the values calculated across the entire loop. Also, a standard deviation is given that will be useful to know if the performance is worse in any of the folds. Considering these factors, a good practice is using k-fold cross-validation with k values set to either 5 or 10. Empirical evidence suggests that these values yield test error rate estimates that achieve an appropriate balance, mitigating common Deep Learning pitfalls as described in [27].

F. Hyperparameter Optimization

Deep Learning models depend on multiple configurable variables that set its behavior and performance. Thus, it is essential to find the best set of hyperparameters to obtain an optimal model. There are different strategies to achieve this task, but we can mention the following: grid search, random search, and Bayesian Optimization. First, one optimizes the hyperparameters by exploring and combining various ranges of possible values to find the best configuration [28]. Random search is another simple and efficient hyperparameter optimization method that randomly samples parameter combinations, offering better performance than grid search in high-dimensional spaces [29]. The use of one or another depends on the model to tune, grid search ensures that all hyperparameter combinations are tested, but needs from a lot of computational resources, so it is better for models where the hyperparameter search space is small. However, random search is more efficient in computational terms, but does not explore all the search space of hyperparameters, so it should be used when the search space is wide. Finally, we have Bayesian Optimization defined by [30] as a probabilistic surrogate model (often a Gaussian process) to represent prior beliefs about the objective function and iteratively updates this model using new data. By balancing exploration and exploitation through acquisition functions, it efficiently selects the next sample points to find the global optimum with fewer evaluations.

G. Avoiding Underfitting and Overfitting

As we have said before, the main aim when training a DL model is avoiding overfitting/underfitting. Apart from that, it is also important that it can perform better than a human doing the specific task for which we are training it. To accomplish this, we will introduce the concept of trade-off between bias and variance [31]. This term addresses the balance between model complexity and its ability to generalize to unseen data (test set). In DL models, this trade-off plays a crucial role in determining model performance and robustness. The bias of a model refers to its ability to capture the underlying patterns in the input data, while variance refers to its sensitivity to small fluctuations during training.

The aim of the bias is accomplished when the metric at the training stage improves how the human performs this task. For example, if we are training a model for the diagnosis of Parkinson's Disease with Magnetic Resonance, we must obtain a better accuracy than that obtained by a physician. A model with low bias may simplify the underlying patterns in the data, leading to underfitting and poor performance on both the training and test sets. Apart from the bias, we use the concept of

variance, which is the difference between the metrics at the training and test stages. A model with high variance may capture noise or irrelevant patterns in the training data, leading to overfitting and poor generalization of unseen data. There is no exact value for this, but having a difference greater than 10% of the value between the training and stages seems too much. In the case of underfitting, the metrics during the training stage could be higher than during the test. When overfitting occurs, the training is lower than the test. In conclusion, the aim is to obtain a high bias and low variance.

To avoid underfitting in DL models, several strategies can be employed. Firstly, increasing model complexity by adding more layers, neurons, or parameters can help capture underlying patterns in the data. Additionally, experimenting with more complex architectures such as CNNs, RNNs, or Transformers tailored to the nature of the data and task can enhance model performance. Another option is training for longer by increasing the number of epochs or iterations, which allows the model to learn intricate patterns for more time.

To prevent overfitting in DL models, several strategies can be employed. Firstly, regularization techniques like dropout and batch normalization. The former consists of temporarily disconnecting several neurons from the network, including all its incoming and outgoing connections [32]. The latter normalizes the inputs of the layers. Another technique is early stopping, where training is stopped when the validation metric stops improving. We can also use data augmentation techniques, such as rotation, translation, scaling, and adding noise, to enhance training diversity. Finally, increasing the size of the training dataset or simplifying the model by reducing layers, neurons, or parameters can also mitigate overfitting.

H. Adding New Metrics for an In-Depth Interpretation of the Model

In Section III D, we talked about the main metrics that could be used for classification and regression problems. These metrics have been used to measure the performance of the model to find a bias-variance trade-off. As a first approach, these metrics are enough, but in some cases, more complex metrics should be used. For example, if the developed model is used in the field of medicine, metrics considering FP and FN should be used. This will let us interpret the models more deeply. Let's consider a model for cancer diagnosis, which is not the same in that it has problems with FN (healthy people diagnosed with cancer) than FP (people with cancer who have been considered as healthy). In the last case, this is a problem that a model should minimize. So, considering all these points, we are introducing more metrics.

In the case of classification problems, metrics like specificity and sensitivity (sometimes called recall) are very useful. Specificity calculates the ratio of true negatives to the sum of true negatives and false positives, while sensitivity performs a similar calculation, changing true negatives for true positives and false positives for false negatives. Finally, precision is computed by dividing the number of true positives by the sum of true positives and false positives. These three equations [Eq. (5) to Eq. (7)] are formalized as follows:

$$\text{Specificity} = \frac{TN}{TP + FN} \quad (5)$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (6)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

The interpretation of the metrics above can be done as follows: Specificity is used so the model can distinguish healthy individuals from those without cancer, crucial for minimizing unnecessary treatments or interventions. In the case of sensitivity, the model can identify most cases of cancer, reducing the risk of treating cases of disease and applying an accurate intervention and treatment, minimizing unnecessary stress for patients. Finally, with precision, we ensured to identify cases of cancer and apply an accurate intervention and treatment. Apart from that, there are other more complex metrics like F1-Score, Area Under the Curve (AUC), and Receiver Operating Characteristic (ROC) Curve.

If we talk about regression problems with DL models, some points should be considered. If the use case consists of predicting a particular value, like a time series problem for predicting the value of a company´s stock, the concepts of FN and FP do not apply. Anyway, there are more metrics like MSE, such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), or R-squared (also Coefficient of Determination) that can be used similarly. But this is not the only case of regression problems that can be solved with DL models; we also have object detection, audio restoration, and others. In these cases, the concepts of FP and FN can be used. Therefore, there exist metrics for this purpose, for example, Intersection over Union (IoU) and Dice score.

IoU computes the ratio of the overlapping area between the predicted (output image of the model) and ground truth (image used in the output layer during training) regions to the total area covered by both regions [33]. It ranges from 0 to 1, where 0 signifies no overlap and 1 denotes perfect spatial alignment between the regions. So, it is defined as the ratio of the area of intersection between the predicted image and ground truth to the area of their union. Eq. (8) formalizes it mathematically:

$$\text{IoU} = \frac{|Predicted \cap Ground\ Truth|}{|Predicted \cup Ground\ Truth|} \quad (8)$$

In the equation above, |Predicted ∩ Ground Truth| represents the area of intersection between the predicted and ground truth regions. Then, |Predicted ∪ Ground Truth| represents the area of union between the predicted and ground truth regions.

Dice score is a metric based on IoU and calculates the ratio of twice the overlapping area to the sum of the sizes of the predicted and ground truth regions [34]. It also ranges from 0 to 1, having the same interpretation as IoU. Eq. (9) describes it:

$$\text{Dice} = \frac{2 \times |Predicted \cap Ground\ Truth|}{|Predicted| + |Predicted|} \quad (9)$$

Regarding the equation above, the numerator is the same as in IoU. Then, |Predicted| represents the size of the predicted region, and |Ground Truth| represents the size of the ground truth region.

I. Making a Comparison with a Baseline

Using a baseline is essential for evaluating the performance of a Deep Learning model for several reasons. Firstly, it provides a point of reference against which the model's performance can be compared. This is crucial in understanding whether the model is learning meaningful patterns in the data or if its performance is merely a matter of luck. Secondly, baselines help in setting realistic expectations for the model's performance. By comparing against a simple baseline, we can measure how much improvement DL techniques offer over simpler approaches. This helps in assessing whether the additional complexity of applying DL techniques is justified. Moreover, using a baseline can also highlight potential shortcomings or biases in the dataset. If a DL model struggles to outperform a simple baseline, it may indicate issues with the quality or representativeness of the data, prompting further investigation and data preprocessing. Overall, employing a baseline is a fundamental step in the evaluation process of Deep Learning models, providing context and insights that are essential for making informed decisions about model performance and deployment.

IV. Use Cases

In this section, we are exploring some use cases where the methodology explained above has been explained in detail, step-by-step. These examples try to cover most of the situations that could happen during the implementation of a DL model, considering the decisions to take. The first use case is a model that could diagnose breast cancer using thermographic images [35]. In the second use case, a diagnostic model for epilepsy using electroencephalographs [36]. Finally, the third use case solves the problem of small brain tumor detection in Magnetic Resonances [37]. In the following Table II, Table III, and Table IV (one for each use case), we summarize each step of the methodology, adding the decision taken and its explanation.

TABLE II. Methodology Applied to Breast Cancer Diagnosis

Step	Decision	Explanation
3.1.	Classification problem with coloured images solved with CNNs	The dataset consists of thermographic images of healthy people and people with breast cancer.
3.2.	Randomise 80-20 split	The number of images is enough, and the images between patients are very different.
3.3.	Scaling: Min-Max	Typical decision when working with images.
3.4.	Accuracy metrics	In a diagnosed problem, we need to measure the number of hits.
3.5.	K-fold validation with k=5	Usual decision
3.6.	Bias-variance trade-off. Train: 87.67%±6.65, Val: 86.67%±12.47, Test: 89.23%±5.85	Bias is correct as human performance accuracy, assisted by a CAD, is near 83%, (Keyserlingk et al. 2020). The variance is also correct.
3.7.	Other metrics. Specificity: 88%, Sensitivity: 90.53% and Precision: 88.91%	These metrics are around the same values which means the model is very stable
3.8.	Baselines. VGG16: 68%. VGG19: 51.33%, Inception 84%	The proposed model performs better than the baseline

TABLE III. Methodology Applied to Epilepsy Diagnosis

Step	Decision	Explanation
3.1.	Classifying chunks of an EEG using a CNN	The dataset is chunked into timestamps that are modelled as graphs represented in images. The CNN model discriminates between stages of a seizure.
3.2.	80-20 split between patients	As brain states in a person do not change a lot from a particular moment and the following, the split is done considering the EEGs of the patients.
3.3.	Scaling: does not apply	As images are in greyscale, they are in the same ranges.
3.4.	Accuracy metrics	For a classification problem, accuracy should be used.
3.5.	K-fold validation with k=5	Usual decision
3.6.	Bias-variance trade-off. Model 1: 93.6%, 88.2%, 87.2%. Model 2: 91.9%, 86.8%, 81.3%	Bias cannot be measured as there are no official studies about the performance of physicians in distinguishing seizure states. The variance is correct.
3.7.	Other metrics. Specificity: 94%, Sensitivity: 94.1% and Precision: 93.2%	These metrics are around the same values, which means the model is very stable.
3.8.	Baselines: nothing.	There is no baseline for being a very specific use case.

TABLE IV. Methodology Applied to Tumor Detection

Step	Decision	Explanation
3.1.	Small brain tumor detection in MR using a U-Net	The dataset is pairs of MRs of the brain and the mask that isolates the brain tumour in this MR
3.2.	Randomise 80-20 split	The number of MRs is enough and they are very different between them.
3.3.	Scaling: Not needed	The images are in greyscale.
3.4.	Loss metrics: MSE	In object detection, we need to know how the predicted mask resembles the ground truth image that represents the mask.
3.5.	K-fold validation with k=5	Usual decision.
3.6.	Bias-variance trade-off. Train:10% ± 0.5%, Validation: 11.1% ± 0.5%, Test: 14%	Human performance goes from 28 and 44% error (the model performs better), and the variance is not high.
3.7.	Other metrics: Dice score. Train: 96.3% ± 0.8%. Validation: 92% ± 1%. Test: 91.6%.	Dice score considers FP and FN.
3.8.	Baselines. Dice score: 45%	The proposed model performs better than the baseline.

V. Conclusion

This study presents a step-by-step methodology to train DL models accurately. Apart from that, it presents the best practices to do so, and complete information is provided during the process. The manuscript should be taken as an information resource for researchers who are not experts in DL and want to use these techniques in their field, so they can understand the main terms and strategies to apply and why. In case of having a deeper understanding of DL models, this could be used as a guide for best practices.

The methodology is divided into eight steps. First, tries to explain which DL models should be used depending on the nature of the data and the use case to solve. Second, describe how to split training, validation, and test sets considering the distribution of the dataset and its similarities. Third, compile different strategies to apply feature scaling, which is related to the value ranges of the different features of the data. The fourth shows which general metrics should be used depending on whether the problem to solve is regression or classification. The fifth describes the validation strategy called k-fold cross-validation. Sixth, introduces the concept of bias-variance trade-off that shows if there is any underfitting or overfitting. Seventh tries to understand in-depth how the model performs by introducing metrics that consider false positives or false negatives. Eighth, evaluates the proposed model against a baseline, so it can be demonstrated that it is better than what can be found in the scientific literature.

Finally, three different use cases have been compiled where we have provided information for each step of the method, pointing out the decision that has been taken and an explanation about it, if possible.

AUTHOR'S CONTRIBUTION

A.N. and A.G.T.: Conceptualization, formal analysis, experimental design, and methodology. A.N.: Manuscript writing, figures, and tables. All authors reviewed the manuscript. A.G.T. and A.M: Approval of the final version of the manuscript.

REFERENCES

[1] S. J. Russell and P. Norvig, "Artificial intelligence: a modern approach. Malaysia," 2016, Pearson Education Limited London, UK:

[2] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Adv. Neural Inf. Process. Syst., vol. 25, pp. 1097–1105, 2012.

[4] C. Sammut and G. I. Webb, Encyclopedia of machine learning. Springer Science \& Business Media, 2011.

[5] H. Li and A. Baghban, "Insights into the prediction of the liquid density of refrigerant systems by artificial intelligent approaches," Sci. Rep., vol. 14, no. 1, p. 2343, 2024.

[6] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2016.

[7] R. Ehtisham et al., "Computing the characteristics of defects in wooden structures using image processing and CNN," Autom. Constr., vol. 158, p. 105211, 2024.

[8] W. Hu, M. Li, H. Xiao, and L. Guan, "Essential genes identification model based on sequence feature map and graph convolutional neural network," BMC Genomics, vol. 25, no. 1, p. 47, 2024.

[9] J. L. Elman, "Finding structure in time," Cogn. Sci., vol. 14, no. 2, pp. 179–211, 1990.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[11] M. Lu and X. Xu, "TRNN: An efficient time-series recurrent neural network for stock price prediction," Inf. Sci. (N Y)., vol. 657, p. 119951, 2024.

[12] R. Sathya, A. Abraham, and others, "Comparison of supervised and unsupervised learning algorithms for pattern classification," International Journal of Advanced Research in Artificial Intelligence, vol. 2, no. 2, pp. 34–38, 2013.

[13] D. H. Ballard, "Modular learning in neural networks.," in AAAI, 1987, pp. 279–284.

[14] P. Smolensky, "Chapter 6: information processing in dynamical systems: foundations of harmony theory," Parallel distributed processing: explorations in the microstructure of cognition, vol. 1, 1985.

[15] R. Salakhutdinov and G. Hinton, "Deep Boltzmann Machines," in Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics, D. van Dyk and M. Welling, Eds., in Proceedings of Machine Learning Research, vol. 5. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, 2009, pp. 448–455. [Online]. Available: https://proceedings.mlr.press/v5/salakhutdinov09a.html

[16] A. Almalki and L. J. Latecki, "Self-Supervised Learning With Masked Autoencoders for Teeth Segmentation From Intra-Oral 3D Scans," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 7820–7830.

[17] A. Vaswani et al., "Attention is all you need," Adv. Neural Inf. Process. Syst., vol. 30, 2017.

[18] A. U. Rahman, Y. Alsenani, A. Zafar, K. Ullah, K. Rabie, and T. Shongwe, "Enhancing heart disease prediction using a self-attention-based transformer model," Sci. Rep., vol. 14, no. 1, p. 514, 2024.

[19] I. Goodfellow et al., "Generative adversarial nets," Adv. Neural Inf. Process. Syst., vol. 27, 2014.

[20] S. P. H. Boroujeni and A. Razi, "IC-GAN: An Improved Conditional Generative Adversarial Network for RGB-to-IR image translation with applications to forest fire monitoring," Expert Syst. Appl., vol. 238, p. 121962, 2024.

[21] P. zhao, D. Ding, K. Li, Y. Li, and G. Jin, "A tunable diode laser absorption spectroscopy (TDLAS) signal denoising method based on LSTM-DAE," Opt. Commun., vol. 567, p. 130327, Sep. 2024, doi: 10.1016/j.optcom.2024.130327.

[22] F. Zhou et al., "Unified CNN-LSTM for keyhole status prediction in PAW based on spatial-temporal features," Expert Syst. Appl., vol. 237, p. 121425, 2024.

[23] R. Dunford, Q. Su, and E. Tamang, "The pareto principle," 2014.

[24] N. Maray, A. H. Ngu, J. Ni, M. Debnath, and L. Wang, "Transfer Learning on Small Datasets for Improved Fall Detection," Sensors, vol. 23, no. 3, p. 1105, Jan. 2023, doi: 10.3390/s23031105.

[25] D. Protić et al., "Numerical Feature Selection and Hyperbolic Tangent Feature Scaling in Machine Learning-Based Detection of Anomalies in the Computer Network Behavior," Electronics (Basel)., vol. 12, no. 19, p. 4158, Oct. 2023, doi: 10.3390/electronics12194158.

[26] A. Jadon, A. Patil, and S. Jadon, "A Comprehensive Survey of Regression-Based Loss Functions for Time Series Forecasting," 2024, pp. 117–147. doi: 10.1007/978-981-97-3245-6_9.

[27] M. Kuhn and K. Johnson, "Over-Fitting and Model Tuning," in Applied Predictive Modeling, New York, NY: Springer New York, 2013, pp. 61–92. doi: 10.1007/978-1-4614-6849-3_4.

[28] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," Adv. Neural Inf. Process. Syst., vol. 24, 2011.

[29] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization.," Journal of machine learning research, vol. 13, no. 2, 2012.

[30] H. J. Kushner, "A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise," Journal of Basic Engineering, vol. 86, no. 1, pp. 97–106, Mar. 1964, doi: 10.1115/1.3653121.

[31] M. Belkin, D. Hsu, S. Ma, and S. Mandal, "Reconciling modern machine-learning practice and the classical bias--variance trade-off," Proceedings of the National Academy of Sciences, vol. 116, no. 32, pp. 15849–15854, 2019.

[32] S. K. Srivastava, S. K. Singh, and J. S. Suri, "Healthcare Text Classification System and its Performance Evaluation: A Source of Better Intelligence by Characterizing Healthcare Text," J. Med. Syst., vol. 42, no. 5, p. 97, May 2018, doi: 10.1007/s10916-018-0941-6.

[33] M. A. Rahman and Y. Wang, "Optimizing Intersection-Over-Union in Deep Neural Networks for Image Segmentation," 2016, pp. 234–244. doi: 10.1007/978-3-319-50835-1_22.

[34] H. Chegraoui et al., "Object Detection Improves Tumour Segmentation in MR Images of Rare Brain Tumours," Cancers (Basel)., vol. 13, no. 23, p. 6113, Dec. 2021, doi: 10.3390/cancers13236113.

[35] A. Nogales, F. Pérez-Lara, and Á. J. García-Tejedor, "Enhancing breast cancer diagnosis with deep learning and evolutionary algorithms: A comparison of approaches using different thermographic imaging treatments," Multimed. Tools Appl., vol. 83, no. 14, pp. 42955–42971, Oct. 2023, doi: 10.1007/s11042-023-17281-x.

[36] A. Nogales, Á. J. García-Tejedor, J. Serrano Vara, and A. Ugalde-Canitrot, "eDeeplepsy: An artificial neural framework to reveal different brain states in children with epileptic spasms," Epilepsy & Behavior, vol. 154, p. 109744, May 2024, doi: 10.1016/j.yebeh.2024.109744.

[37] A. Nogales, C. de la Pinta, Á. J. García-Tejedor, E. García-Barriocanal, D. Guadalupe, and M.-A. Sicilia, "Evaluating the Detection of Small Brain Lesions in Magnetic Resonances Using Deep Learning," 2025. doi: 10.2139/ssrn.5364234.

A Constraint-Driven Conversational Architecture for Staged Task Interaction Using Large Language Models

Joseph Benjamin Ilagan, Jose Ramon Ilagan
Ateneo Business Insights Laboratory for Development (BUILD)
Ateneo de Manila University, Quezon City, Philippines 1108

***Abstract*—Large language models (LLMs) enable flexible conversational interfaces, but remain difficult to deploy in structured, staged tasks that require controlled progression, bounded information disclosure, and task validity. Prompt-based control is inherently probabilistic and has been shown to degrade under multi-turn interaction, leading to premature solution disclosure, stage skipping, loss of state coherence, and constraint violations. This study presents a constraint-driven conversational architecture that separates probabilistic language generation from deterministic task governance. An external control layer manages dialogue state, stage transitions, and constraint enforcement, while a simulation layer represents task logic independently of the LLM. We instantiate the architecture in the context of customer discovery tasks to illustrate how staged processes and bounded disclosure can be operationalized without embedding task logic directly into prompts. This work focuses on architectural design and control mechanisms rather than outcome evaluation, offering a reusable architectural pattern for LLM-driven conversational systems that must preserve staged progression, enforce constraints, and prevent premature disclosure during multi-turn interaction.**

***Keywords*—Large language models; conversational systems; task-oriented dialogue; deterministic control; constraint enforcement; staged interaction**

I. Introduction

Large language models (LLMs) have enabled conversational interfaces for tasks that previously relied on scripted workflows or rule-based dialogue systems. However, when tasks require staged progression and procedural integrity, prompt-based control remains unreliable. Common failures include premature solution disclosure, stage skipping, uncontrolled transitions, loss of state coherence, and domain constraint violations. These failures limit use in settings where the process itself must be preserved.

Prior work has improved multi-turn interaction through prompting techniques and learned policies, but task governance is still often mediated through model behavior. This makes guarantees on state, constraints, and stage boundaries difficult to obtain.

This study proposes a constraint-driven conversational architecture that separates probabilistic language generation from deterministic task governance. An external control layer maintains dialogue state, regulates stage transitions, and enforces constraints, while a simulation layer represents task logic independently of the LLM to support runtime checks and bounded information disclosure. We instantiate the architecture in customer discovery tasks to illustrate staged interaction without embedding task logic directly into prompts. This work focuses on architectural design and control mechanisms rather than outcome evaluation.

It is important to clarify the nature of the contribution made in this work. The proposed architecture does not claim novelty in staged task representations, finite-state control, or deterministic dialogue management as isolated techniques. Related ideas appear in classical dialogue system architectures and in hierarchical control formulations for sequential decision-making [1], [2].

The contribution instead lies in repositioning deterministic task governance as a first-class architectural layer in LLM-driven conversational systems, where language generation is probabilistic and state coherence cannot be assumed. In this setting, staged progression, constraint enforcement, and bounded disclosure are treated as external guarantees instead of emergent properties of prompt design or model reasoning. This distinction is central to the architectural argument advanced in this study.

A. Research Questions

This study addresses the following research questions:

- RQ1. What system architecture can support structured, staged task interaction with LLM-based conversational interfaces, while preserving natural language flexibility?
- RQ2. How can dialogue state, stage transitions, and permissible system actions be represented and governed deterministically and not through prompt interpretation?
- RQ3. What runtime mechanisms enable enforceable constraints and bounded information disclosure, including integration of an external task logic or simulation layer independent of the LLM?

B. Contributions

This study makes the following contributions:

- We present a constraint-driven architecture for LLM-based conversational systems that separates language generation from deterministic task control for staged interaction.

- We define a dialogue state and control model that governs stage transitions, permissible system actions, and bounded information disclosure outside the LLM.
- We show how the architecture can be operationalized with an external simulation/control layer, using customer discovery as an illustrative domain.

II. Review of Related Literature

This section reviews prior work related to: 1) controlling large language model (LLM) behavior in multi-turn settings, 2) maintaining explicit state and constraints in structured tasks, and 3) architectural patterns that separate probabilistic language generation from deterministic task governance. The review is organized around control mechanisms (prompting and agents), externalization of state (memory, simulators, controllers), and system-level designs for constrained interaction.

A. Prompt Engineering for Task Control

Prompt engineering is a widely used mechanism for steering large language models (LLMs) in conversational systems without modifying model parameters. Surveys of prompting methods document a range of strategies, including instruction prompts, role prompting, and structured prompt templates intended to improve task adherence across diverse settings [3].

Several prompt engineering techniques have emerged to increase reliability in multi-turn interaction. Common approaches include explicitly stated behavioral constraints, stepwise instructions, and structured response formats that narrow the space of acceptable outputs. In practice, these methods remain inherently probabilistic because task logic and constraints are interpreted as opposed to enforced, and can degrade across longer conversations or under distribution shift [3].

Few-shot learning extends prompt-based control by providing a small number of exemplars that demonstrate the expected behavior [4]. In staged tasks, exemplars can illustrate valid responses and interaction norms, but they do not provide explicit guarantees on state persistence, bounded information disclosure, or correct stage transitions, particularly when user behavior deviates from the examples.

A related technique known as *chain-of-thought* (CoT) prompting encourages models to generate intermediate reasoning steps prior to producing a final answer [5]. CoT prompting can improve multi-step reasoning performance, but the reasoning trace remains part of the model output and not an external representation of task state. As a result, CoT does not by itself enforce task constraints or regulate progression in structured, staged interactions.

Overall, prompt-centric approaches provide flexible and low-overhead control of LLM behavior, but task control, constraint enforcement, and state management are encoded implicitly in natural language prompts. This implicit encoding makes it difficult to ensure consistent adherence to staged task structures or to prevent failure modes such as premature information disclosure and uncontrolled stage transitions in conversational systems.

B. Agentic and Tool-Augmented LLM Systems

Tool-augmented and agentic systems extend LLMs with actions such as retrieval, API calls, and tool usage. ReAct couples reasoning traces with action selection in an interleaved loop [6]. Such approaches improve task completion on interactive problems, but the policy is typically still mediated through model generations, not an external controller.

Toolformer proposes self-supervised learning of tool usage, allowing models to invoke tools through simple APIs [7]. These frameworks motivate modularity, yet many implementations still treat constraints, stage boundaries, and permissible transitions as soft rules embedded in prompts or tool schemas, not explicit governance layers.

C. State, Memory, and Environment-Mediated Interaction

Long-horizon task interaction requires stable representations of state, goals, and constraints. Existing work spans symbolic dialogue management, hybrid neural-symbolic systems, and environment-mediated learning, where an external environment provides dynamics and feedback [2]. Related lines of work use simulators for training, evaluation, or controlled interaction loops [8]. A key distinction for this study is treating simulation/control as a persistent runtime layer that governs progression, not only as a training or evaluation artifact.

D. Architectural Patterns for Conversational Systems

Architectural perspectives on dialogue systems emphasize modular separation of components such as NLU, dialogue state tracking, dialogue policy, and NLG [1]. These patterns motivate separation of concerns, but they do not fully anticipate LLM-first dialogue generators where the generator can blur boundaries among state, policy, and surface realization.

E. Synthesis and Gap

Across the strands reviewed in this section, task governance is predominantly implicit and probabilistic, even when external tools or environments are introduced. Constraints, stage boundaries, and permissible transitions are typically specified as prompt rules, learned behaviors, or conventions mediated through language generation, rather than enforced through explicit deterministic control structures.

Although deterministic state machines and hand-crafted control policies are well-known in task-oriented dialogue more broadly, these designs largely predate LLM-first conversational architectures and are rarely positioned, in the LLM-centric literature reviewed here, as the primary mechanism for guaranteeing staged progression, bounded disclosure, and constraint satisfaction during runtime interaction.

This work responds to this gap by treating deterministic task governance as a first-class architectural abstraction. The proposed constraint-driven conversational architecture explicitly separates probabilistic language generation from deterministic task control, elevating stage representation, enforceable transition logic, and constraint checks to core components that regulate what the LLM may say and when (see Table I).

TABLE I. COMPARISON OF TASK GOVERNANCE APPROACHES IN LLM-BASED CONVERSATIONAL SYSTEMS

Property	Prompt-centric	Agentic	Proposed (Constraint-driven)
Stage enforcement	Probabilistic	Emergent	Deterministic
State representation	Implicit	Partial	Explicit
Disclosure control	Prompt-level	Tool-schema	Controller-enforced
Transition authority	Model	Model	Controller
Recovery on violation	None/retry	Model-driven	Deterministic fallback
Task logic location	In prompt	Mixed	External layer
Cross-domain reuse	Low	Moderate	High (config-driven)

III. ARCHITECTURE

A. Conceptual Model for Deterministic Task Governance

The architecture proposed in this study is guided by a simple conceptual model that distinguishes between *probabilistic language generation* and *deterministic task governance*. While large language models (LLMs) are well suited for producing fluent, context-sensitive responses, they do not provide intrinsic guarantees on task progression, state persistence, or constraint satisfaction across multi-turn interactions. These properties must, therefore, be enforced outside the language model.

Procedural integrity, as used in this study, refers to the property that stage position, transition validity, and disclosure boundaries are maintained by deterministic system components throughout the interaction lifecycle, independent of model behavior.

At the core of the model is the treatment of task state as a first-class system artifact. Dialogue state, stage membership, and permissible actions are represented explicitly and persist independently of conversational turns. Instead of inferring state from natural language alone, the system maintains an external representation of where the interaction currently resides within a staged task structure. This enables consistent reasoning about what information may be disclosed, which actions are valid, and which transitions are allowed at any point in the interaction.

A second principle is the separation of task logic from surface realization. Task constraints, progression rules, and evaluation conditions are encoded in a simulation or control layer that operates deterministically. The role of the LLM is restricted to generating natural language realizations that are conditioned on, and bounded by, decisions made in this control layer. This separation prevents task logic from being embedded implicitly in prompts, where it would otherwise be interpreted probabilistically and remain difficult to enforce over extended conversations.

Finally, grounding is achieved through controlled exposure to task-relevant context in place of generic conversational guidance. System responses are generated with reference to the current task state and its associated constraints, allowing feedback and guidance to be tailored to the user's position within the staged process. This design supports specificity and consistency in guidance while preserving the flexibility of natural language interaction.

Together, these principles define a constraint-driven conversational model in which LLMs function as flexible linguistic interfaces governed by explicit, deterministic task structures. This conceptual framing underpins the architectural design presented in the following section.

B. Architectural Contribution Clarification

The architecture proposed in this study should be understood as a governance pattern and not a new dialogue algorithm. Its primary contribution is the explicit separation of conversational fluency from procedural authority in LLM-based systems. Unlike prompt-centric or agentic designs, where task control is mediated through model interpretation or reasoning traces, the proposed architecture assigns exclusive authority over task state, stage transitions, and disclosure boundaries to deterministic system components. This framing addresses failure modes that arise specifically in LLM-first conversational settings and are not adequately handled by classical dialogue pipelines or workflow engines when language generation is open-ended and user-driven.

C. Overall System Architecture

This subsection provides a high-level view of the proposed conversational architecture, outlining the primary components and their interactions (see Fig. 1). The design emphasizes a clear separation between probabilistic language generation and deterministic task governance to support structured, staged interaction.

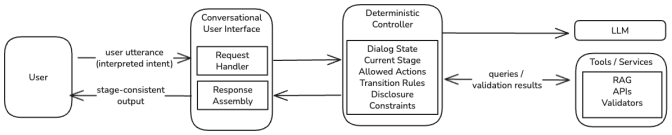


Fig. 1. Constraint-driven conversational architecture with deterministic task governance. The controller governs dialogue state, stage transitions, and disclosure constraints, invoking the LLM and external tools as bounded services.

D. Dialogue State Representation

Dialogue state is represented as an explicit, structured data object maintained by the control layer across conversational turns. The state encodes the current task stage, completed and pending artifacts, validated inputs, and flags relevant to progression and constraint enforcement. This representation persists independently of the conversation transcript and is not inferred solely from natural language interaction.

State updates occur through deterministic events, not through linguistic interpretation alone. User inputs may trigger validation checks or proposed updates, but acceptance of new state values depends on evaluation against task specifications defined in the control or simulation layer. This prevents ambiguous or partial inputs from implicitly advancing the task.

Separating dialogue state from conversational history enables consistent reasoning about task position even when the

interaction includes digressions, reformulations, or clarification requests. It also supports recovery from invalid inputs without relying on the language model to reconstruct prior context.

Dialogue state is treated as a persistent system artifact that may be stored across turns or sessions using any suitable state storage mechanism (e.g., in-memory stores, databases, or session services), which is intentionally left abstract in this architecture.

E. Stage Transitions and Control Logic

Structured tasks are modeled as staged processes governed by explicit transition rules rather than inferred conversational flow. Each task defines a finite set of stages, allowable transitions between stages, and conditions under which transitions may occur. These elements are treated as part of the task specification and are evaluated by a deterministic control layer at runtime.

Stage definitions are external to the language model and may be configured per task or use case. A stage specification encodes the set of valid stages, permitted successor stages, entry and exit conditions, and constraints on permissible system actions and disclosures at each stage. This allows the same conversational runtime to support different staged tasks without modifying the underlying control logic.

Transition evaluation is performed exclusively by the control layer. User inputs and LLM-generated responses may propose or suggest progression, but transitions are only executed when explicit transition conditions are satisfied. These conditions may include completion of required task elements, validation of user inputs against the task logic or simulation layer, or satisfaction of domain-specific constraints.

Invalid transitions, such as premature advancement or attempts to bypass required stages, are blocked by design. When a transition request is rejected, the system remains in the current stage and generates responses that are consistent with the active stage's disclosure and action constraints. This ensures that task progression remains valid even under adversarial, ambiguous, or unexpected user behavior.

By separating transition authority from language generation, the architecture enforces staged progression deterministically while preserving conversational flexibility. The LLM is responsible for surface realization and interaction quality, whereas the control layer maintains procedural integrity throughout the task lifecycle.

F. Constraint Enforcement and Bounded Disclosure

Constraint enforcement is implemented through stage-specific policies managed by the control layer. These policies define which system actions are permitted, which information may be disclosed, and which responses are prohibited at a given point in the interaction.

While avoiding encoding constraints as prompt instructions alone, the controller regulates what contextual information is provided to the language model and which response types are allowed. Future-stage artifacts, evaluative conclusions, or solution templates are withheld until the task reaches the appropriate stage, ensuring that the model cannot disclose information it has not been authorized to access.

When a user request violates active constraints, the controller blocks the corresponding action and triggers a recovery response. The recovery response is generated within the bounds of the current stage and redirects the interaction without advancing state or revealing restricted information. This preserves procedural integrity under adversarial, ambiguous, or premature user requests.

G. Integration of the Simulation or Task Logic Layer

Task logic is implemented in a simulation or rule-based layer that operates independently of the language model. This layer encodes domain-specific requirements such as completeness conditions, validity checks, dependency constraints, and progression criteria.

User inputs requiring structured interpretation are routed to the simulation layer for evaluation. The simulation returns deterministic signals such as completion flags, constraint violations, or required revisions. These signals are consumed by the controller and translated into bounded response contexts for the language model.

This separation allows domain logic to evolve without modifying prompt templates or conversational behavior. It also reduces reliance on the language model for correctness judgments, limiting its role to explanation and feedback consistent with the simulation's outputs.

H. Interaction Flow

Each interaction cycle follows a fixed control sequence. User input is first received and recorded without immediate state mutation. The controller evaluates whether the input proposes a state transition, requests restricted information, or requires validation against task logic.

If validation is required, the input is passed to the simulation layer. The resulting signals are incorporated into the dialogue state and used to determine the next permissible system action. Only after this decision is made does the controller construct a bounded response context for the language model.

The language model generates a natural language response within the constraints imposed by the controller. Conversational output does not directly modify task state. State updates occur only through explicit controller approval, ensuring that language generation cannot unilaterally advance task progression.

Algorithm 1 summarizes the deterministic runtime loop used to govern staged progression and disclosure.

The CHECKRESPONSE step in Algorithm 1 verifies that the LLM-generated response r_t does not reference disallowed disclosures or propose actions outside $\mathcal{A}_t$. Verification may be implemented through keyword filtering, structural pattern matching, or a secondary classifier. When a violation is detected, RECOVERYRESPONSE generates a stage-consistent fallback that redirects the user without modifying state or revealing restricted information.

Algorithm 1 Deterministic Controller Loop for Staged Conversational Tasks

Inputs: user message u_t, current state s_t, task spec $\mathcal{T}$, validator/simulator $\mathcal{V}$, LLM $\mathcal{M}$
Output: response text r_t, updated state s_{t+1}

$i_t \leftarrow \textsc{ClassifyIntent}(u_t, s_t, \mathcal{T})$ ▷ e.g., ask, revise, request_restricted, propose_transition
$x_t \leftarrow \textsc{ExtractStructuredInputs}(u_t, s_t, \mathcal{T})$ ▷ optional parsing
$v_t \leftarrow \mathcal{V}(x_t, s_t, \mathcal{T})$ ▷ returns completion, violations, missing fields
$\mathcal{A}_t \leftarrow \textsc{ComputeAllowedActions}(i_t, v_t, s_t, \mathcal{T})$
$\mathcal{D}_t \leftarrow \textsc{ComputeAllowedDisclosures}(s_t, \mathcal{T})$
$c_t \leftarrow \textsc{BuildBoundedContext}(s_t, v_t, \mathcal{A}_t, \mathcal{D}_t)$
$r_t \leftarrow \mathcal{M}(c_t)$
$ok \leftarrow \textsc{CheckResponse}(r_t, \mathcal{A}_t, \mathcal{D}_t)$
if $ok =$ **false then**
  $r_t \leftarrow \textsc{RecoveryResponse}(s_t, v_t, \mathcal{A}_t)$
end if
$s_{t+1} \leftarrow \textsc{CommitStateUpdate}(s_t, u_t, v_t, r_t, \mathcal{T})$
return r_t, s_{t+1}

IV. Illustrative Instantiation: Customer Discovery as a Staged Conversational Task

This section presents an illustrative instantiation of the proposed architecture using customer discovery as an example of a staged conversational task. The purpose is to demonstrate how deterministic task governance, bounded disclosure, and stage-aware interaction are operationalized in a concrete domain, not to evaluate task outcomes or user performance.

A. Task Structure and Stages

Customer discovery is modeled as a staged conversational task in which progression is constrained by explicit requirements at each stage. The staged structure reflects the need to prevent premature solution formulation while ensuring that users engage with each step of the discovery process in sequence (see Fig. 2).

In the illustrative instantiation, the task is decomposed into a finite set of stages, such as problem articulation, hypothesis formulation, interview planning, evidence gathering, and reflection. Each stage is associated with a well-defined set of permissible system actions, allowable disclosures, and completion conditions. These definitions are external to the language model and are maintained by the control layer as part of the task specification.

Stage progression is not inferred from conversational cues alone. Instead, advancement occurs only when the control layer determines that the conditions for exiting the current stage have been satisfied. For example, a transition from problem articulation to hypothesis formulation may require that the user has provided a sufficiently specific problem statement, as validated by the task logic or simulation layer. Attempts to advance without meeting these conditions are rejected, and the system remains within the current stage.

Treating stages as explicit system entities as opposed to conversational conventions ensures that task structure is preserved across multi-turn interaction. This approach allows conversational flexibility within a stage while preventing stage skipping, uncontrolled transitions, or disclosure of information that is inappropriate for the user's current position in the task.

TABLE II. Example Stage Specification for Customer Discovery under Deterministic Governance

Field	Specification (illustrative)
Stages	S1 Problem articulation; S2 Hypothesis formulation; S3 Interview planning; S4 Evidence gathering; S5 Reflection
State variables	`stage_id`; `artifacts` (problem_statement, hypotheses, interview_questions, notes); `flags` (validated, ready_to_advance); `violations`
Required artifacts	S1: problem statement draft; S2: testable hypothesis; S3: interview guide; S4: evidence log; S5: updated belief + next actions
Allowed actions	Ask clarification; critique current artifact; request missing fields; summarize progress; propose next micro-step within stage; attempt transition (controller decides)
Disallowed disclosures	No downstream templates or "answers" for future stages; no solution proposals before S5; no auto-advancing to S3+ without required artifacts
Exit criteria	S1: problem statement meets specificity checks; S2: hypothesis is falsifiable and linked to problem; S3: interview guide covers objectives and avoids leading questions; S4: evidence log meets completeness; S5: reflection includes decision and rationale
Validator outputs	`completion_flag`; `missing_fields`; `violations`; `allowed_actions`; `allowed_disclosures`

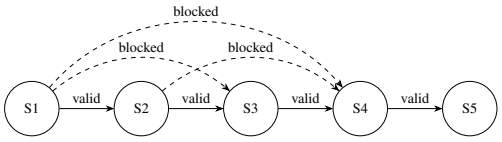


Fig. 2. State transition diagram for the customer discovery instantiation. Solid arrows indicate controller-approved transitions; dashed arrows indicate blocked skip attempts.

Table II provides an illustrative stage specification that externalizes exit criteria, allowed actions, and disclosure limits.

B. Grounded Feedback and Coaching Behavior

The architecture supports coaching behavior by making feedback generation stage-aware and state-dependent. Rather than prompting the model to "behave" as a coach across all turns, the control layer determines the allowable feedback type at the current stage and provides the LLM with stage-bounded context and permissible action options.

At runtime, the controller constructs a constrained response context that includes: 1) the current stage identifier, 2) relevant state variables and artifacts collected so far (e.g., problem statement drafts, hypotheses, interview questions), and 3) an explicit policy describing what the system may and may not disclose in the current stage. The LLM then produces natural language realizations of actions selected or permitted by the controller, such as asking a clarifying question, requesting missing inputs, offering critique on a draft artifact, or summarizing progress.

Grounding is operationalized through controlled access to task artifacts and simulator outputs. When user input requires validation or structured interpretation, the controller routes it to the task logic or simulation layer, which returns deterministic signals such as completion flags, constraint violations, or required revisions. The LLM is not expected to infer these signals from the conversation transcript. Instead, the controller passes the relevant outcomes as response constraints, ensur-

Worked trace: staged customer discovery with bounded disclosure (illustrative).

State (before): `stage_id=S1 (Problem articulation)`, `artifact.problem_statement=underspecified`, `ready_to_advance=false`.

User: "Can you help me write interview questions now so I can test this quickly?"
Intent: `request_future_stage_artifact (S3)`.
Validator: `completion_flag=false`; `missing_fields={target user segment, situation, observable pain}`; `violations={premature stage transition}`.

Allowed actions: ask clarification; critique current problem statement; request missing details.
Disallowed disclosures: interview question templates; hypothesis phrasing; discovery call scripts.

System (LLM realization within bounds):
"Before we design interviews, we need to be precise about the problem. Who specifically experiences this issue, in what recurring situation, and what concrete cost or friction do they face? Try rewriting the problem in one sentence using those elements."

State (after): `stage_id=S1`; `pending={target user segment, situation, pain}`; `ready_to_advance=false`.

Fig. 3. Illustrative trace showing deterministic blocking of premature interview design during customer discovery.

ing that coaching feedback remains consistent with the task specification.

This design yields two practical effects. First, feedback remains aligned with stage intent, since the controller restricts the response space to coaching behaviors appropriate to the current stage. Second, bounded disclosure is preserved, since the LLM does not receive privileged information or future-stage hints unless explicitly authorized by the controller. The result is a conversational experience that remains flexible in language while remaining disciplined in instructional sequencing.

Fig. 3 shows a concrete turn-level example of how stage skipping and premature disclosure are blocked at runtime. This example reflects early-stage customer discovery, where interview design is intentionally blocked until the problem is explicitly articulated.

C. Observed Failure Modes and Mitigations

The architecture is motivated by failure modes commonly observed in prompt-governed multi-turn interaction for staged tasks. In prompt-centric implementations, the model is expected to remember stage position, respect procedural boundaries, and withhold information until the appropriate time. In practice, these behaviors are unreliable under long-horizon conversation, ambiguous user prompts, or user attempts to accelerate the process.

A frequent failure mode is premature solution disclosure, where the model provides downstream answers (e.g., proposed solutions, scripts, or conclusions) before earlier stages have been completed. Another is stage skipping, where the model implicitly advances the task after a single user message, compressing intermediate stages or omitting required artifacts. Conversational drift is also common, where the interaction gradually shifts toward generic advice that is weakly related to the staged task, reducing coherence and diminishing the value of accumulated state. Finally, constraint violations can occur when the model produces outputs that conflict with domain rules, such as recommending actions that the task specification explicitly forbids at a given stage.

The proposed architecture mitigates these failure modes through deterministic governance. Premature disclosure is controlled through stage-specific disclosure policies that limit which information and actions are available to the LLM at each stage. Stage skipping is prevented by enforcing transition checks in the controller, so that advancement can only occur when explicit completion conditions are met. Drift is reduced by anchoring each turn to an explicit task state representation and by routing key evaluations to the simulation layer instead of relying on narrative consistency in model outputs. Constraint violations are handled through runtime checks and rejection of invalid actions, followed by controlled recovery responses that redirect the interaction without changing the active stage.

These mitigations are not framed as guarantees of correctness in the model's linguistic output. Rather, they provide guarantees about procedural integrity: stage position, transition validity, and bounded information disclosure are governed by deterministic system components, while the LLM remains responsible for conversational naturalness within the permitted space.

V. Discussion

A. Design Trade-Offs

The proposed architecture also differs from conventional workflow and orchestration systems commonly used for multistage task execution. Workflow engines and automation platforms are designed to coordinate deterministic system-to-system processes, typically assuming that task progression is driven by predefined triggers, API responses, or execution outcomes in place of open-ended human interaction.

In contrast, the architecture presented in this study addresses staged task progression in the presence of natural language interaction, where user input and model-generated responses introduce uncertainty and variability. The challenge is not merely sequencing actions, but governing conversational progression, state coherence, and information disclosure under probabilistic language generation.

While workflow systems excel at executing predefined pipelines, they do not address failure modes specific to LLM-driven interaction, such as premature solution disclosure, stage skipping, or conversational drift. The proposed control layer operates at a different abstraction level, mediating between conversational inputs and task logic to ensure that staged processes remain valid even when interaction is negotiated through language as opposed to enforced through execution constraints alone.

B. Relation to Prompt-Centric and Agentic Approaches

Recent agentic approaches to large language model (LLM) systems emphasize autonomous planning, action selection, and tool use driven largely by model-generated reasoning traces. In such systems, the LLM is often responsible for determining when to advance between steps, invoke tools, or terminate a task, with control logic embedded implicitly in prompts or emergent reasoning behavior.

The architecture proposed in this study adopts a deliberately different stance on agency. Rather than treating the LLM as an autonomous task manager, the model is positioned as a bounded conversational service whose outputs are mediated by an external control layer. While the LLM may propose actions or suggest progression, authority over dialogue state transitions, stage advancement, and permissible disclosures resides exclusively in deterministic system components.

This design choice reflects a prioritization of procedural integrity over autonomous flexibility. By externalizing task governance, the architecture avoids relying on probabilistic reasoning traces or prompt adherence to enforce staged progression. As a result, conversational fluency is preserved while guarantees over task structure, bounded disclosure, and valid transitions are maintained independently of model behavior.

The proposed approach is not intended to replace agentic systems in domains where autonomous planning is desirable. Instead, it targets settings where the integrity of staged processes is critical and where conversational interaction must remain flexible without allowing the model to unilaterally control task execution.

The distinction between agentic autonomy and deterministic governance is intentional. The proposed architecture does not attempt to improve the model's ability to plan or reason about task execution. Instead, it removes responsibility for procedural correctness from the model altogether. This design choice reflects a different objective than that of agentic systems. The goal is not autonomous task completion, but preservation of staged interaction integrity under conversational uncertainty. This difference in objective leads to a different allocation of responsibility across system components.

C. Generality and Transferability

The proposed architecture is not specific to customer discovery tasks. For example, an entrepreneurship education scaffold could gate hypothesis formulation until a problem statement meets specificity criteria, while a customer validation workflow could restrict solution proposals until sufficient interview evidence has been logged — both mirroring the staged structure demonstrated in Section IV.

Transferability is achieved through task configuration in place of architectural modification. Stages, constraints, and validation logic are defined externally and can be replaced without changing the conversational runtime. This allows the same system to support multiple task types with differing structures and domain rules.

The architecture therefore functions as a reusable governance layer for conversational systems instead of a domain-specific application design.

VI. Limitations and Future Work

A. Current Limitations

The architecture introduces additional system complexity relative to prompt-centric approaches. Explicit controllers, state representations, and simulation layers require greater upfront design effort and tighter integration among system components.

The approach also assumes that task structure can be sufficiently formalized. Domains with highly subjective evaluation criteria or weakly defined progression rules may require hybrid designs that combine deterministic checks with probabilistic judgment.

The architecture does not address optimization of language quality or persuasive effectiveness. Its focus is procedural integrity as opposed to conversational richness.

The deterministic governance model may also introduce rigidity in domains where task progression is inherently exploratory or where valid sequences are not fully specifiable in advance. Creative workflows, open-ended research tasks, or domains with subjective completion criteria may require hybrid designs in which constraint strictness is relaxed or dynamically negotiated.

B. Future Extensions

Future work may explore declarative task specification languages that allow stages, constraints, and transition logic to be defined without custom code. This would reduce the effort required to author new staged tasks.

Empirical evaluation of user experience and learning outcomes under deterministic governance is another natural extension. Comparative studies could examine differences between prompt-centric, agentic, and constraint-driven architectures.

Adaptive control policies may also be investigated, allowing constraint strictness or feedback style to vary based on observed user behavior while preserving deterministic stage boundaries.

VII. Conclusion

This study presented a constraint-driven conversational architecture for supporting structured, staged task interaction with large language models. The proposed design separates probabilistic language generation from deterministic task governance by introducing an external control layer responsible for dialogue state management, stage transitions, and constraint enforcement. By externalizing task logic and progression rules, the architecture avoids relying on prompt-based control alone to preserve procedural integrity during multi-turn interaction.

Through an illustrative instantiation in customer discovery tasks, the study demonstrated how staged processes and bounded information disclosure can be operationalized without embedding task logic directly into language prompts. The architecture enables conversational flexibility while maintaining explicit guarantees over task progression, permissible actions, and disclosure boundaries.

Rather than focusing on outcome evaluation, this work contributes an architectural pattern for LLM-driven conversational

systems that require structured interaction and enforceable task constraints. The proposed approach is intended to be reusable across domains where the integrity of staged processes is critical, and it provides a foundation for future work that may explore empirical evaluation, task configuration languages, or adaptive control policies built atop the same deterministic governance framework.

The central contribution of this work is the articulation of deterministic task governance as an architectural requirement for LLM-driven conversational systems, instead of a behavioral property to be induced through prompting or autonomous reasoning.

FUNDING

This work was supported by the Ateneo de Manila University Research Council under the University Research Council (URC) grant program, Grant No. URC-2026-05, awarded to Joseph Benjamin Ilagan.

ACKNOWLEDGMENT

The authors thank Maria Mercedes T. Rodrigo, Ph.D., the Department of Information Systems and Computer Science (DISCS) of the Ateneo de Manila University, the John Gokongwei School of Management (JGSOM), the Ateneo Business Insights Laboratory for Development (BUILD), and the Ateneo University Research Office (UREO) for institutional support and administrative assistance related to this work.

DECLARATION ON GENERATIVE AI

Generative artificial intelligence tools were used to support language refinement and drafting during manuscript preparation. All scientific content, system design decisions, arguments, and conclusions were developed by the human authors. The authors critically reviewed, edited, and take full responsibility for the accuracy, originality, and integrity of the manuscript.

REFERENCES

[1] D. Jurafsky and J. H. Martin, "Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition," 2009, publication Title: Prentice Hall Series in Artificial Intelligence.

[2] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum, "Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation," *Advances in neural information processing systems*, vol. 29, 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/hash/f442d33fa06832082290ad8544a8da27-Abstract.html

[3] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, Sep. 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3560815

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and A. Askell, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html?utm_source=transaction&utm_medium=email&utm_campaign=linkedin_newsletter

[5] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

[6] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "React: Synergizing reasoning and acting in language models. arXiv 2022," *arXiv preprint arXiv:2210.03629*, vol. 10, 2023.

[7] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, E. Hambro, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," *Advances in Neural Information Processing Systems*, vol. 36, pp. 68 539–68 551, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html

[8] Z. Zhang, D. Zhang-Li, J. Yu, L. Gong, J. Zhou, Z. Hao, J. Jiang, J. Cao, H. Liu, and Z. Liu, "Simulating classroom education with llm-empowered agents," in *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, 2025, pp. 10 364–10 379. [Online]. Available: https://aclanthology.org/2025.naacl-long.520/

Measurement of Dataset Quality and Computational Quality of Graph Neural Networks on Analog Integrated Circuit Recognition System

Arif Abdul Mannan[1], Koichi Tanno[2]
Faculty of Engineering, University of Miyazaki, Miyazaki, Japan[1]
Department of Electrical Engineering, Brawijaya University, Malang, Indonesia[1, 2]

***Abstract*—To meet the needs of AI design in analog IC design automation and especially in big data applications, the application of sensitive and precise Graph Neural Network (GNN) recognition will become increasingly necessary. This sensitive GNN recognition is needed to accommodate the highly stringent trade-offs on analog IC design requirements, combined with the increasingly large learning data requirements. Furthermore, more precise GNN recognition will be a fundamental requirement for a more sensitive system to accommodate noise in floating point (FP) calculations, namely inaccuracies and imprecision in IEEE 754 FP calculations. In this study, by refining a previously proposed method that uses the output vector representation (OVR) of the untrained Graph Neural Network (GNN), and by exploiting the numerical reproducibility error in FP calculations, a method for measuring the sensitivity and precision of GNNs for use in analog IC design recognition has been proposed. The results of the sensitivity and precision measurements of the proposed method show a complex combination of effects between dataset quality (the presence or absence of data duplication), the sensitivity of the GNN to distinguish each feature in the dataset, the complexity of the calculations that occur in the GNN, and the level of quality of the FP calculations performed by the processing unit related to the non-ideal FP calculations. With certain GNN configurations, the proposed method also succeeded in measuring the difference in FP calculation quality of Central Processing Unit (CPU) and Graphics Processing Unit (GPU), where, for big data applications, from the tests carried out, the maximum amount of data that can be distinguished by the CPU is 25 to 100 times more than with the GPU. Because it only uses untrained GNN OVR and does not involve any training process in obtaining results, the proposed method is still unable to obtain a correlation with the final value of GNN performance after training is complete. The measurement method using GNN OVR involving the learning process is future work.**

***Keywords*—Big data; Graph Neural Network; Artificial Intelligence; calculation quality measurement; analog circuit design**

I. Introduction

Artificial intelligence (AI) is highly needed in the world of Integrated Circuit (IC) design. However, the application of AI in analog IC design is still very limited. The main reason for the slow implementation of AI in analog IC design is the very stringent trade-offs inherent in analog IC design and the high dependence on the knowledge of engineers. Small parameter changes in an analog IC design can cause significant changes in the behavior and output of the analog IC being designed [1], [2], [3], [4], [5]. Therefore, to design an AI system for analog IC design applications, a neural network (NN) system capable of distinguishing even the smallest changes in analog circuit design parameters is required [6], [7]. In other words, in big data applications, AI applied to analog IC design must have a high level of sensitivity [8], [9], [10], [11].

The application of NN, particularly graph neural networks (GNNs), to analog IC design has been discussed previously [1], [8], [12]. The application of GNNs such as Graph Convolutional Network (GCN) [13], Graph Isomorphism Network (GIN) [14], Graph Attention Network V2 (GAT) [15], and GraphSage (GSG) [16] has also been studied. All these GNNs utilize floating-point (FP) calculations. As we know, FP calculations are inherently inaccurate and imprecise due to the internal processes within the Central Processing Unit (CPU) or Graphics Processing Unit (GPU). As long as FP calculations in CPUs/GPUs still use IEEE 754 [17], these inaccuracies and imprecisions are inevitable [18], [19], [20], [21], [22], [23]. Therefore, all GNNs applied to analog IC design must not only be able to distinguish very small parameter changes (high sensitivity) but also possess high precision, allowing them to tolerate the inaccuracies and imprecisions inherent in the floating-point calculation system within the CPU/GPU.

The ability to measure the sensitivity and precision of a neural network (NN) [24] used in analog IC design would be extremely useful for simplifying the AI design process, if feasible. The application of GNN output calculations to detect data duplication in a dataset for analog IC design, to measure the sensitivity of NN inputs, has been successfully implemented using a floating-point digit rounding system [8]. However, this proposed implementation in [8] has a weakness: determining the exact value for the floating-point digit rounding in data duplication detection remains uncertain. This proposed implementation only uses exact values based on best-case trials, leaving the challenge of measuring the precision of floating-point calculations in NNs unresolved.

In this study, a new method for determining the exact value of floating-point digit rounding in measuring the level of data duplication in datasets, specifically for the application of GNNs in analog IC design, is proposed. This new idea is based on exploiting the inaccuracy and imprecision of floating-point calculations within the CPU/GPU itself. Therefore, in addition to obtaining the ability to measure the level of duplication and/or measure the level of GNN's ability to distinguish each data in the dataset (GNNs sensitivity) more precisely, with the

technique proposed in this study, the ability to measure the level of precision of the CPU/GPU (or other NN processors, including the possibility of FPGA [25]) in processing GNN calculations can also be obtained. Therefore, the use of 32-bit, 64-bit, or even the possibility of using 16-bit FP in [26] can be determined. Although this measurement method is intended for analog IC design, the use of this proposed method can be applied directly to other applications, whether using GNN or other artificial NN (ANN), as long as they have the same OVR structure.

This study consists of the following sections: The method for obtaining calculation quality measurements from GNNs is described in Section II. Simulations and discussions are presented in Section III, while conclusions and future work are presented in Section IV.

II. GNN Calculation Quality Measurement

In this study, the measurement of GNN calculation quality is obtained by optimizing and developing a previous method that also uses an untrained GNN output vector representation (OVR) in [8]. The use of untrained GNN OVR is very useful because it is expected that the process of measuring GNN performance (sensitivity and precision) can be done even before the training process occurs. Before discussing the method of measuring the calculation quality of GNNs, graph datasets and GNN output will first be explained.

For a graph-based dataset, consider a dataset that contains $\mathcal{V}$ number of graph information as $\mathcal{G}(\mathcal{V},\mathcal{E})$, with additional information that can also be added, such as the input features $\{x_v, \forall_v \in \mathcal{V}\}$, have variate number of vertex (node v) and edge (ε) array length on every graph. The GNN OVR z_v is obtained from every neighborhood vector $h_v^{(l)}$ for all $v \in \mathcal{V}$ as shown in Eq. (1):

$$z_v = h_v^{(L)} \quad ,\forall v \in \mathcal{V} \tag{1}$$

As explained in the previously proposed method, in this study, the same GNN OVR z_v structure, as shown in Eq. (2), will be used as the basis for the calculation. This results in variations in $\mathcal{V}$ and $\mathcal{E}$ being negligible and resulting in data with a fixed C dimensional output (C is the number of classes in the dataset) being generated.

$$z_v = \left(z_{v_1}, z_{v_2}, z_{v_3}, \cdots z_{v_c}\right), \forall v \in \mathcal{V} \tag{2}$$

Based on the output value of GNN OVR z_v, $prpDist$ parameter and $prDiff$ parameter will be obtained [8]. The $prpDist$ is the pseudo distance parameter, calculated simply by using the sum of the absolute value of every axis's scalar value difference between two points, as shown in Eq. (3):

$$prpDist_p = \begin{cases} \sum_{i=1}^{C} \left| z_{r_i} - z_{p_i} \right| & at\ r \neq p \quad r,p \in \mathcal{V} \\ 0 & at\ r = p \quad r,p \in \mathcal{V} \end{cases} \tag{3}$$

With z_r and z_p is vector value of the reference point and point under test in the C -dimension output, respectively. The $prDiff$ is the "difference" parameter, calculating the relative difference between two points of OVR data on the C dimensional space into one scalar number, as shown in Eq. (4):

$$prDiff_p = \sum_{i=1}^{C} (z_{r_i} - z_{p_i}) n_{r_i} \tag{4}$$

With n_{r_i} is obtained from the i-th element of N_r, the normalized reference point array (NRA), consists of fractional number with range from $0 \leq n_{r_i} \leq 1$ and calculated using Eq. (5):

$$N_r = \frac{Z_r - z_{r_{min}}}{z_{r_{max}} - z_{r_{min}}} \tag{5}$$

A. Twice GNN OVR Generation

Techniques for measuring the imperfections of FP calculations using shadow execution [18], [19], and incorporating external factors into the calculation [20] have been proposed previously. In this "calculation quality measurement," the measurement is obtained by calculating the difference between two GNN OVR results performed using the same environment and configuration. Ideally, these two GNN OVR results should be the same, but due to the numerical reproducibility error in FP calculations [21], [22], [23], the two GNN OVR results will differ. Simply put, in this study, the measurement method is carried out by exploiting numerical reproducibility errors in FP calculations. As shown in Algorithm 1, these two (or more) GNN OVR outputs can be easily obtained by using one of the following three methods:

1) Identical input generation: Recognize the untrained GNN twice using the same input dataset. Based on Eq. (1) and Eq. (2), this identical input generation method can be described as Eq. (6):

$$\begin{aligned} z1_v &= \left(z1_{v_1}, z1_{v_2}, z1_{v_3}, \cdots z1_{v_c}\right) = h_v^{(L)}, \forall v \in \mathcal{V} \\ z2_v &= \left(z2_{v_1}, z2_{v_2}, z2_{v_3}, \cdots z2_{v_c}\right) = h_v^{(L)}, \forall v \in \mathcal{V} \end{aligned} \tag{6}$$

2) Reversed input generation: Recognize the untrained GNN once using the initial input dataset, then perform a second recognition using the initial dataset with its graph sequence reversed. The GNN OVR output from the second recognition is then reversed to match the order of the first recognition results. Based on Eq. (1) and Eq. (2), this reversed input generation method can be described as Eq. (7):

$$\begin{aligned} z1_v &= \left(z1_{v_1}, z1_{v_2}, z1_{v_3}, \cdots z1_{v_c}\right) = h_v^{(L)}, \forall v \in \mathcal{V} \\ z_j &= \left(z_{j_1}, z_{j_2}, z_{j_3}, \cdots z_{j_c}\right) = h_j^{(L)}, \forall j \in J \end{aligned} \tag{7}$$

With $J = \mathcal{V}[::-1]$ and $z2_v = z_j[::-1]$. The $[::-1]$ is the operation of reverse order of the array data.

3) Randomized input generation: Recognize the untrained GNN once using the initial input dataset, then perform a second recognition using the initial dataset with its graphs sequence randomized. The GNN OVR output from the second recognition is then reordered (in the reverse order of the

previous randomization) to match the order of the first recognition results. Based on Eq. (1) and Eq. (2), this randomized input generation method can be described as Eq. (8):

$$z1_v = \left(z1_{v_1}, z1_{v_2}, z1_{v_3}, \cdots z1_{v_c}\right) = h_v^{(L)}, \forall v \in \mathcal{V}$$
$$z_m = \left(z_{m_1}, z_{m_2}, z_{m_3}, \cdots z_{m_c}\right) = h_m^{(L)}, \forall m \in M \quad (8)$$

With $M = R(\mathcal{V})$ and $z2_v = R^{-1}(z_m)$. The $R()$ is the operation or the process of randomly arranging the array sequence based on a relationship table that has been created randomly before recognition is carried out, and the $R^{-1}()$ is the process of returning an array order from random to normal based on the same table.

Algorithm 1: Two GNN OVR generator

```
PROGRAM OVRMatrixGenIden()
INPUT h_v
OUTPUT z1_v, z2_v
Compute
for v in V do
z1_v ← recog(h_v)
end for
for v in V do
z2_v ← recog(h_v)
end for
return z1_v, z2_v

PROGRAM OVRMatrixGenReversed()
INPUT h_v
OUTPUT z1_v, z2_v
Compute
for v in V do
z1_v ← recog(h_v)
end for
h_j ← h_v [::-1]    // h_j is h_v in reverse order
for j in J do
temp_j ← recog(h_j)
end for
z2_v ← temp_j [::-1]
return z1_v, z2_v

PROGRAM OVRMatrixGenRandom()
INPUT h_v
OUTPUT z1_v, z2_v
Compute
dataSuffleIndex ← RandomList(0, len(h))
for v in V do
z1_v ← recog(h_v)
end for
for index, data in dataSuffleIndex
h_m_index ← h_v_data  // h_m is h_v in random order
end for
for m in M do
temp_m ← recog(h_m)
end for
for index, data in dataSuffleIndex
z2_data ← temp_index
end for
return z1_v, z2_v
```

To produce the two GNN OVR outputs, only one of the three existing methods is required. In this study, the randomized input generation (with the highest probability of numerical reproducibility error in FP calculations) will be used as a reference for the simulation in Section III.

After the two GNN OVRs are calculated, $prpDist$ and $prDiff$ will be calculated using Eq. (3) and Eq. (4). Each GNN OVR will produce one pair of variables $prpDist$ and $prDiff$. Therefore, the variables $prpDist1$ and $prDiff1$ will be obtained from the OVR $z1_v$, while $prpDist2$ and $prDiff2$ will be obtained from the OVR $z2_v$ calculation.

"**Important!** Starting from $prpDist$ and $prDiff$ calculations onward, using more accurate and precise floating-point calculations is crucial. If the GNN OVR calculation uses 32-bit FP, then calculating $prpDist$ and $prDiff$ requires at least 64-bit FP to obtain a precise measurement of the GNN calculation quality."

B. Calculation Quality Measurement

In this study, two variables will be determined in the GNN calculation quality measurement results: the calculation quality percentage (CQP) and the calculation quality digit (CQD). The CQP is defined as the percentage of data certainty (still able to ascertain its precision) that is still unaffected by floating point inaccuracy and imprecision at each decimal point. Meanwhile, the calculation quality digit is defined as the lowest digit value in a FP calculation result that can still be ascertained for its precision before errors from floating point inaccuracy and imprecision affect it. Examples of GNN calculation quality measurements are presented in TABLE I and Fig. 1. TABLE I shows that up to 7 digits after the decimal point, 100% of the data can be ascertained for its precision. Meanwhile, for up to 8 digits after the decimal point, the data with ascertained precision drops to 83%, indicating that 17% of the data has experienced precision degradation due to non-ideal FP calculations. Therefore, the CQD based on the example of TABLE I is 7.

TABLE I. EXAMPLE OF GNN CALCULATION QUALITY RESULT

Decimal Places	*Calculation Quality Percentage (CQP)*	*Remarks*
0	100%	
1	100%	
2	100%	
3	100%	
4	100%	
5	100%	
6	100%	
7	100%	<= CQD
8	83%	
9	20%	
10	15%	

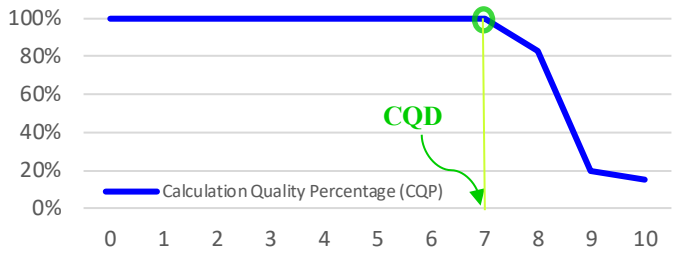


Fig. 1. The example of GNN calculation quality chart.

By using the two pairs of $prpDist$ and $prDiff$ generated in Section II-A, the quality measurement of the GNN calculations can be performed. Assuming that the FP calculations in Section II-A are ideal, then all the results of $prpDist1$ will be the same as $prpDist2$. However, floating point calculations on CPU/GPU are not ideal (floating point inaccuracy and imprecision), so there will be a slight difference between $prpDist1$ and $prpDist2$ as shown in Eq. (9):

$$prpDist1_p \approx prpDist2_p \quad ,\forall p \in \mathcal{V} \tag{9}$$

If we describe the exact value of $prpDist2$ results, we will obtain Eq. (10). From Eq. (9) and Eq. (10), if applied to prdiff1 and prdiff2, then Eq. (11) will be obtained.

$$\epsilon Dist_p = prpDist1_p - prpDist2_p \quad ,\forall p \in \mathcal{V} \tag{10}$$

$$\epsilon Diff_p = prDiff1_p - prDiff2_p \quad ,\forall p \in \mathcal{V} \tag{11}$$

where, $\epsilon Dist$ and $\epsilon Diff$ are the error values resulting from the inaccuracy and imprecision of FP numbers.

By obtaining the $\epsilon Dist$ value, the quality of the GNN calculation will be determined based on the exponent value of the most significant digit in $\epsilon Dist$ compared to the exponent value of the most significant digit in $prpDist1$ or $prpDist2$. The same applies to $\epsilon Diff$. The exponent value of the most significant digit can be obtained using Algorithm 2.

Algorithm 2: Get Exponential Scale

```
PROGRAM GetExpScale()
INPUT x_v
OUTPUT t_v
Compute
for v in V do
t_v ← truncate(log(x_v))
end for
return t_v
```

By obtaining the highest exponent value of $prpDist1$ (one can be selected between $prpDist1$ and $prpDist2$), calculation Eq. (12) can be performed. Therefore, the relative error value $ErDist_p$ of $\epsilon Dist$ based on $prpDist1$ can be obtained from Eq. (13):

$$tDist1_p = GetExpScale(prpDist1_p) \tag{12}$$

$$ErDist_p = \epsilon Dist_p \odot 10^{tDist1_p} \quad ,\forall p \in \mathcal{V} \tag{13}$$

The same is true for $prDiff$, so that Eq. (14) and Eq. (15) can be obtained:

$$tDiff1_p = GetExpScale(prDiff1_p) \tag{14}$$

$$ErDiff_p = \epsilon Diff_p \odot 10^{tDiff1_p} \quad ,\forall p \in \mathcal{V} \tag{15}$$

Based on the results from Eq. (13) and Eq. (15), the CQP calculation process can be carried out using Algorithm 3. Algorithm 3 is only able to calculate the CQP value at one decimal place. So if a CQP with a decimal point value ranging from 0 to 10 is required (as in the example in TABLE I), then 11 calculations using Algorithm 3 are required.

Algorithm 3: Get Single Quality Percentage

```
PROGRAM GetQualityPercentage()
INPUT ErDist_p, ErDiff_p, DecimalTest
OUTPUT Qp
Compute
for p in V do
tempDist ← ErDist_p * 10^DecimalTest
resDist_p ← |tempDist| < 1
tempDiff ← ErDiff_p * 10^DecimalTest
resDiff_p ← |tempDiff| < 1
end for
resAll = logical_and (resDist_p, resDiff_p)
resGood = nonzero (resAll )
Qp = 100 * len(resGood) / len(resAll )
return Qp
```

C. Special Value Decimal Places

With the $\epsilon Dist$ and $\epsilon Diff$ values obtained from Eq. (10) and Eq. (11), and based on the calculations performed in Algorithm 3, the decimal digit values that can be entered do not have to be integers. The digit values entered can be fractional numbers. In this study, the digit values with values $x.3$ and $x.6$ are chosen to increase the resolution of the CQP results. The digit value $x.3$ is chosen because the value of $10^{-0.3}$ has a value close to 0.5 (the exact value used in the actual calculation as shown in Algorithm 3 is $10^{x.30103}$; or the value 10^{x} multiplied by 2), and the digit value $x.6$ is chosen because $10^{-0.6}$ has a value close to 0.25 (the exact value used in the actual calculation as shown in Algorithm 3 is $10^{x.60206}$; or the value 10^{x} multiplied by 4). Example of rounding down using 2, 2.3, 2.6 and 3 decimal places after the comma is shown is TABLE II.

TABLE II. EXAMPLE OF ROUNDING DOWN USING 2, 2.3, 2.6 AND 3 DECIMAL PLACES

Real Numbers	Rounding (down) n-digits after the decimal point			
	2.0	2.3	2.6	3.0
3.1214	3.1200	3.1200	3.1200	3.1210
3.1231	3.1200	3.1200	3.1225	3.1230
3.1274	3.1200	3.1250	3.1250	3.1270
3.1287	3.1200	3.1250	3.1275	3.1280
3.1330	3.1300	3.1300	3.1325	3.1330

III. SIMULATION AND DISCUSSION

In the simulation in this study, the simulation results of dataset duplication detection from [8] will be included. The

results of dataset duplication detection will be named the GNN & Dataset Compatibility (GDC) variable, which consists of: GNN & dataset compatibility map (GDCM) and GNN & dataset compatibility value (GDCV). Similar to CQP, which has its own value at each decimal point, GDCM shows the combined value of GNN sensitivity and dataset quality (dataset uniqueness; the presence or absence of data duplication) at each decimal point. Meanwhile, GDCV is the value of GNN & Dataset Compatibility exactly at the decimal point indicated by CQD.

The main purpose of measuring the calculation quality and compatibility of GNN & Dataset is to help the design of GNN that will be used in analog IC design automation system. In this study, for analog IC design applications, the dataset to be used, extracted using the method in [27], has the features as shown in TABLE III. In this simulation section, this dataset will be referred to as "MOSFET".

TABLE III. MOSFET DATASET FEATURE

Datasets Feature	Quantity
Number of data	2,115
Number of classes	121
Min data per class	1
Max data per class	795
NETLIST length	5,214,505
NETLIST number of line	95,615
Number of MOSFET(s)	74,626
Number of Power Supply(s)	4,234
Number of Resistors(s)	7,804
Number of LC(s)	11

For all simulations measuring the quality of GNN calculations and dataset quality, a personal computer with a 10850K CPU, 64GB RAM, an RTX 4070 Ti GPU, a Windows 10 system, and a Python environment with PyTorch for neural network computations were used. To obtain GNN OVR data, simulations were performed with both the CPU and or GPU in 32-bit FP format. Meanwhile, calculations starting from $prpDist$ and $prDiff$ up to CQP, CQD, GDCM, and GDCV were performed purely on the CPU with a higher FP accuracy configuration of 64-bit FP format.

A. Measurement Results of 1-Node Multi-Edge and Net Connection Abstraction MOSFET

In this simulation section, the quality of GNN calculations and the dataset quality of MOSFET with 1-Node Multi-Edge & Net Connection (1NMC) abstraction in [12] is measured. The measurement results of CQP, CQD, GDCM and GDCV for 2115 data (121 classes) in 1NMC abstraction are shown in TABLE IV and Fig. 2. This simulation was performed using GIN (there is no specific reason for choosing GIN; GCN, GAT, GSG, or other GNNs could also be used), 128-dim input layer dimension, no hidden layer, 128-dim output layer, and GPU calculation as the Processing Unit (PU) to generate GNN OVR. SDD in TABLE IV represents "successfully differentiated data", indicating the number of unique data that are successfully differentiated by GNN, while DIN (Data integrity number) represents the number of data that are not affected by FP calculation noise (inaccuracy and imprecision of FP calculation).

TABLE IV. GIN 128-DIM, NO HIDDEN LAYER, 128-DIM, GPU

Decimal Places	SDD	GDCM	DIN	CQP	Remarks
0.0	23	1.10%	2115	100.00%	
0.3	77	3.60%	2115	100.00%	
0.6	183	8.70%	2115	100.00%	
1.0	404	19.10%	2115	100.00%	
1.3	741	35.00%	2115	100.00%	
1.6	889	42.00%	2115	100.00%	
2.0	1036	49.00%	2115	100.00%	
2.3	1140	53.90%	2115	100.00%	
2.6	1211	57.30%	2115	100.00%	
3.0	1299	61.40%	2115	100.00%	
3.3	1309	61.90%	2115	100.00%	
3.6	1310	61.90%	2115	100.00%	
4.0	1312	62.00%	2115	100.00%	
4.3	1312	62.00%	2115	100.00%	
4.6	1312	62.00%	2115	100.00%	<= CQD
5.0	1316	62.20%	2108	99.70%	
5.3	1334	63.10%	2069	97.80%	
5.6	1376	65.10%	1967	93.00%	
6.0	1516	71.70%	1556	73.60%	
6.3	1665	78.70%	1161	54.90%	
6.6	1806	85.40%	760	35.90%	
7.0	1945	92.00%	215	10.20%	
7.3	2029	95.90%	62	2.90%	
7.6	2074	98.10%	18	0.90%	
8.0	2101	99.30%	6	0.30%	
8.3	2110	99.80%	1	0.00%	
8.6	2114	100.00%	1	0.00%	
9.0	2115	100.00%	1	0.00%	
9.3	2115	100.00%	1	0.00%	
9.6	2115	100.00%	1	0.00%	
10.0	2115	100.00%	1	0.00%	

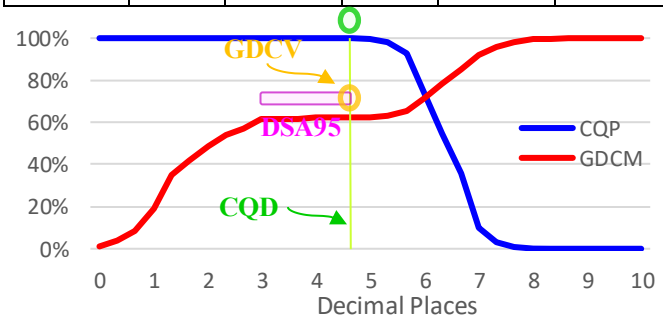


Fig. 2. Dataset and calculation quality measurement for 1NMC abstraction MOSFET using GIN 128-dim, GPU.

From the measurement results outlined in TABLE IV and Fig. 2, the following points can be observed:

1) The CQD value of the 1NMC abstraction MOSFET is 4.3 decimal places. This indicates that the inaccuracy and imprecision of the FP calculation begin to affect the GNN output starting at 4.6 decimal places.

2) The GDCV for the 1NMC abstraction MOSFET is 1312/2115, or 62.0%, at 4.3 decimal places. This indicates that the GNN settings in this measurement failed to distinguish 803 from 2115 data points, even though the GNN calculation was not yet affected by the imperfections of the FP calculation. It can be concluded that the GNN quality of 62.0% is due to either the dataset containing duplicate data or the GNN's insufficient power to distinguish between the data points.

3) If the Data Stability Area of 95 (DSA95) is defined as the area where the GNN can distinguish at least 95% of the data from the GDCV value and has a sensitivity of 100 decimal places, then the DSA95 of a 1NMC abstraction MOSFET is 1.6. This means the GNN can distinguish more than 58.9% of the data (95% from the GDCV of 62.0%) to 1.7 decimal places before the imperfections in the FP calculation begin to degrade the calculation's precision. Basically, the imperfection resulting from the FP calculation can be called noise, producing a random effect on the GNN calculation. In other words, DSA95 can be equated with the Signal to Noise Ratio (SNR) in the Analog IC world, where the noise floor starts from 4.6 decimal places (95% was chosen as a consideration that 95% is included in the stable category in the author's view; if a minimum of 98% is desired, it will be DSA98; the number can be anything, from 0% to 100%).

B. Optimized Dataset Duplication Detection

The GDC quality measurement results obtained can be generated from one or both of the dataset duplication and the GNN's ability to distinguish each data. To be able to obtain a more accurate level of dataset duplication, measurements can be carried out using various methods so that the effect of the GNN's inability to distinguish each data can be minimized. Methods that can be done include increasing or reducing the number of GNN hidden layers, changing the dimensions of the GNN layers, using various types of GNN, and using different PU (CPU / GPU or other NN processors). Of these various methods, the highest GDCV results can be said to be the value closest to the actual dataset duplication.

TABLE V. GIN 128-DIM WITH 0, 1, 2, AND 3-HIDDEN LAYER; GPU

GNN type	n-dim	n-hidden layer	CQD	GDCV	DSA95	PU
GIN	128	0	4.6	62.0%	1.6	GPU
GIN	128	1	4.3	62.0%	1.3	GPU
GIN	128	2	4.3	62.1%	1.3	GPU
GIN	128	3	4.6	62.1%	1.6	GPU

Using the 1NMC abstraction MOSFET dataset, the measurement results of the quality of GNN calculations with several kinds of hidden layers, various kinds of layer dimensions, various types of GNNs, and the use of CPU and GPU processors are shown in TABLE V and Fig. 3, TABLE VI and Fig. 4, TABLE VII and Fig. 5, and TABLE VIII and Fig. 6, respectively.

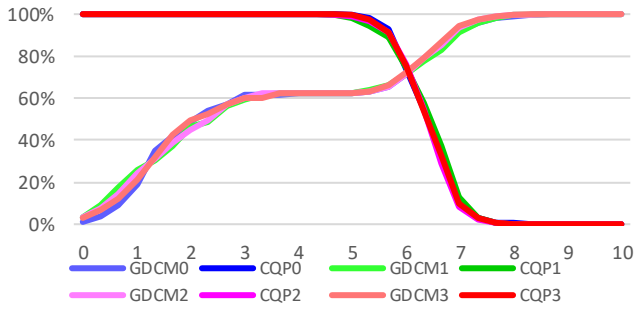


Fig. 3. Dataset and calculation quality measurement for 1NMC abstraction MOSFET using GIN 128-dim, multiple hidden layer, GPU.

TABLE VI. GIN 2, 4, 8, AND 32-DIM WITH 1-HIDDEN LAYER; GPU

GNN type	n-dim	n-hidden layer	CQD	GDCV	DSA95	PU
GIN	2	1	4.6	47.9%	2.3	GPU
GIN	4	1	4.6	62.0%	2.3	GPU
GIN	8	1	4.6	62.0%	2.3	GPU
GIN	32	1	5.0	62.0%	2.0	GPU

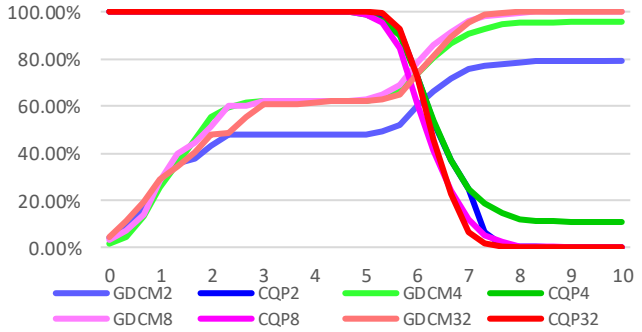


Fig. 4. Dataset and calculation quality measurement for 1NMC abstraction MOSFET using GIN multiple dim, GPU.

TABLE VII. GCN, GIN, GAT, GSG 128-DIM WITH 2-HIDDEN LAYER; GPU

GNN type	n-dim	n-hidden layer	CQD	GDCV	DSA95	PU
GCN	128	2	6.3	61.9%	1.7	GPU
GIN	128	2	4.6	62.1%	1.3	GPU
GSG	128	2	6.3	61.9%	1.7	GPU
GAT	128	2	6.6	62.0%	2.0	GPU

Based on the measurement results as described in TABLE V, TABLE VI, TABLE VII, and TABLE VIII, the highest GDCV value is 1314/2115 (62.1%). This value can be concluded as the closest value to the actual total data duplication (there is still a possibility that all GNN configurations that have been carried out have the same weaknesses and fail to distinguish certain data). Or if measurements are carried out with the aim of designing GNN in analog IC design automation, the GNN configuration that is most sensitive to the 1NMC abstraction MOSFET dataset is the GNN with a configuration that has a GDCV of 62.1%.

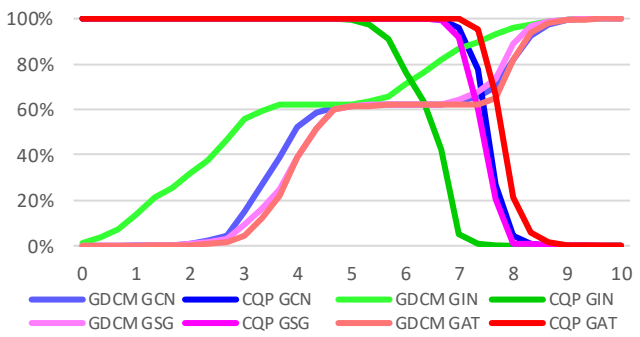


Fig. 5. Dataset and calculation quality measurement for 1NMC abstraction MOSFET using GCN, GIN, GAT and GSG on GPU.

TABLE VIII. GCN, GIN, GAT, GSG 128-DIM WITH 2-HIDDEN LAYER; CPU

GNN type	n-dim	n-hidden layer	CQD	GDCV	DSA95	PU
GCN	128	2	7.3	62.0%	2.3	CPU
GIN	128	2	5.3	62.1%	2.3	CPU
GSG	128	2	6.6	61.9%	2.0	CPU
GAT	128	2	6.6	62.0%	1.6	CPU

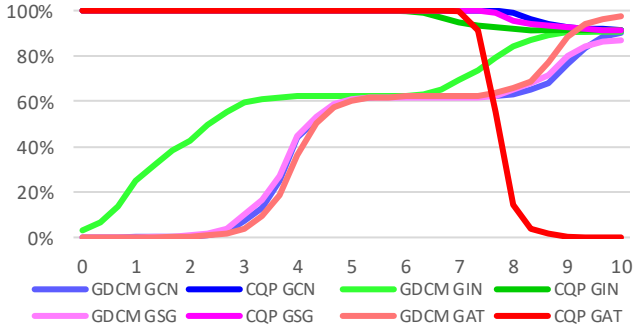


Fig. 6. Dataset and calculation quality measurement for 1NMC abstraction MOSFET using GCN, GIN, GAT and GSG on CPU.

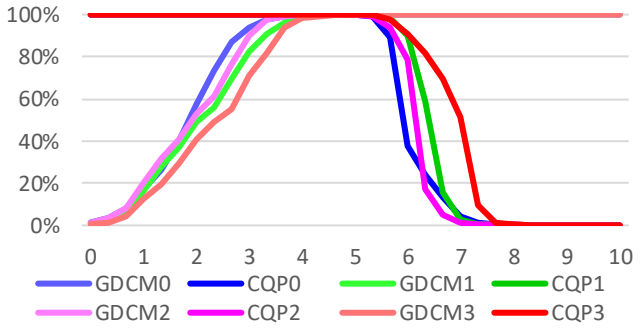


Fig. 7. Dataset and calculation quality measurement for 4NC abstraction MOSFET with multiple hidden layer.

As a comparison of the 1NMC abstraction MOSFET dataset, the measurement process with various GNN configurations was also carried out on the 4-nodes & Net Connection (4NC) abstraction MOSFET dataset in [12] with one of the measurement results shown in Fig. 7. The best GDCV value obtained by this 4NC abstraction MOSFET dataset is 2114/2115 (99.95%). Therefore, compared with the 1NMC abstraction, based on the measurement results obtained, the decision to use the 4NC abstraction is more recommended for use in designing AI for analog IC designs.

C. CPU and GPU 32-Bit FP Performance

The GNN calculation quality measurement in this study can also be used to compare the calculation quality of the processing unit used (such as CPU or GPU) in applying the desired GNN & dataset configuration. The method used in comparing this processing unit calculation quality is by generating one "untrained GNN" (as an initial example, generated on the CPU), and performing recognition (to generate the GNN OVR). Then this "untrained GNN" is transferred to another processing unit (for example, GPU) and performing recognition with the same dataset. The GNN OVR generation process does not have to be only once, in this simulation, each processing unit, four sets of GNN OVR are generated.

The performance comparison of this processing unit calculation quality will be proportional to the difference in CQP and CQD obtained. As an example case, the difference in calculation quality between CPU and GPU for the 1NMC abstraction MOSFET dataset using a 32-dimensional 1-hidden-layer GIN is shown in Fig. 8 and TABLE IX. Both CPU and GPU in this test use 32-bit FP in their calculation process.

TABLE IX. CPU VS. GPU CALCULATION QUALITY MEASUREMENT

GNN type	n-dim	n-hidden layer	CQD	GDCV	DSA95	PU
GIN	32	1	5.3	62.0%	2.0	GPU
GIN	32	1	5.3	62.0%	2.0	GPU
GIN	32	1	5.3	62.1%	2.0	GPU
GIN	32	1	5.0	62.0%	1.7	GPU
GIN	32	1	6.0	62.0%	2.7	CPU
GIN	32	1	6.0	62.0%	2.7	CPU
GIN	32	1	6.0	62.0%	2.7	CPU
GIN	32	1	6.0	62.0%	2.7	CPU

Based on Fig. 8, several results are obtained. First, when the CQP is 100% (without any influence from the FP calculation imperfections, confirmed up to 5.0 decimal places), the 4 sets of GDCMs from the CPU and 4 sets of GDCMs from the GPU have identical results, shows that the results of the dataset quality measurement (GDC) are independent from the PU being used. Second, after the FP calculation imperfections begin to take effect, in the CQP of the GPU calculation results, 4 out of 4 measured sets vary from each other, while in the CQP of the CPU calculation results, 4 out of 4 measured sets are all identical. This shows that FP calculations in CPUs are more repeatable, more capable of producing the same output values when using the same input and calculation formulas compared to GPUs. And third, based on the measurement results shown in Fig. 8 and Table IX, it is observed that the CQD of the GPU is 0.7 to 1.0 decimal places lower than that of the CPU. Since the calculation quality measurement uses two variables $prpDist$ and $prDiff$, for applications in the big data field, the maximum amount of data in the dataset that can be differentiated (utilizing the entire $prpDist$ and $prDiff$ range that is not affected by the imperfections of the FP calculation) by the CPU will be 25 to 100 times greater than that of the GPU.

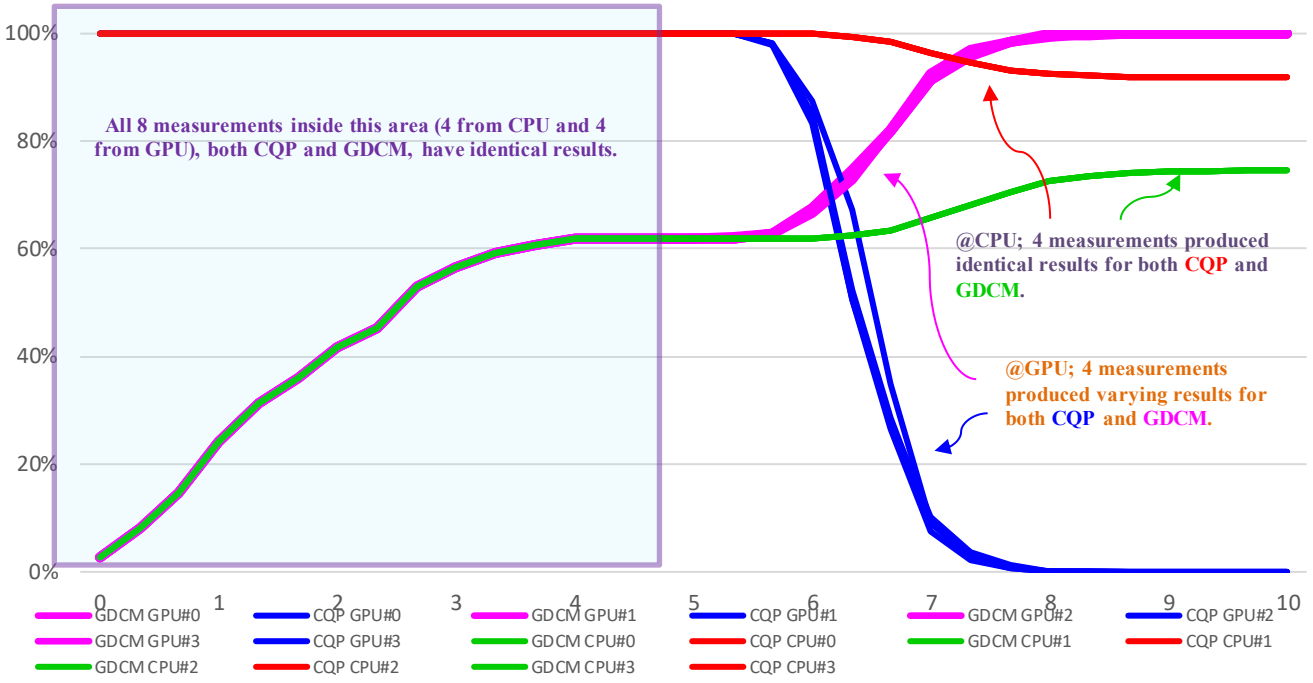


Fig. 8. CPU vs. GPU calculation quality measurement for 1NMC abstraction MOSFET using GIN.

Note: This comparison involves the effects of the programming language used, compiler version, operating system version, device driver version, device microcode version, and other internal processing unit factors. Currently, in this study, these factors cannot be separated and measured individually, so they will be represented in this study as a single variable: the CPU or GPU. If GNN OVR can be generated by applying variations of these factors, then individual measurements for these factors will be possible. Another important thing has also been observed from this comparison of the quality of CPU and GPU calculations: the difference in GNN calculation performance between CPU and GPU is also significantly influenced by GNN & dataset configuration factors, particularly the GNN type (GCN, GIN, GAT, GSG, or other GNN types), as observed in Fig. 5 and Fig. 6. Therefore, the results of this GNN calculation quality measurement can be used as a new opportunity in benchmarking the performance of GNN algorithms.

D. Limitation

The GNN calculation quality measurement in this study has limitations related to the precision of the measurement results. In conducting the measurements, the data used as the basis for calculating CQP, CQD, GDCM, and GDCV, are generated from the recognition of untrained random weight GNN. Therefore, the result of each random weight initialization (the process of creating an untrained GNN) at the beginning of the measurement will produce a small random effect on the measurement results (observed to have very significant random results in certain extreme configurations). The measurement results for 4 times initialization of untrained random weight GNN using the same configuration, namely GIN, 32-dim layer dimension, 1 hidden layer, GPU, are shown in Fig. 9. As well as the example for the measurement results for extreme case conditions using the configuration GIN, 2-dim layer dimension, 1 hidden layer, GPU, are shown in Fig. 10. One example of an extreme configuration that was found to produce very significant random output in this measurement of the quality of GNN calculations is a very low GNN layer dimension configuration.

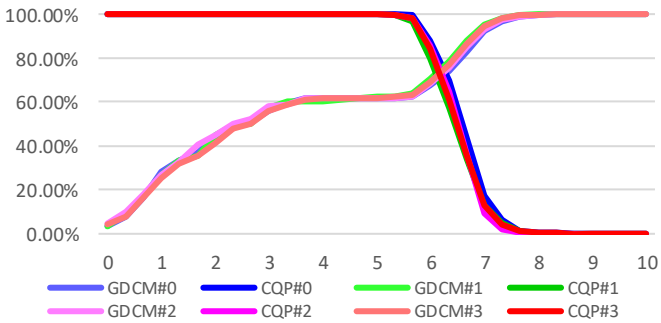


Fig. 9. Dataset and calculation quality measurement for 4N abstraction MOSFET with 32-dim.

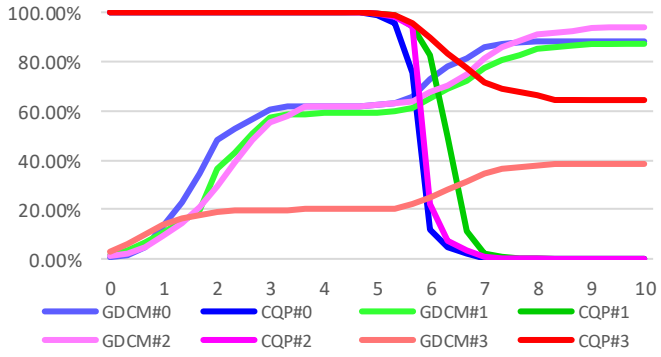


Fig. 10. Dataset and calculation quality measurement for 4N abstraction MOSFET with 2-dim.

IV. CONCLUSION

In this study, a method for measuring dataset quality and GNN calculation quality has been proposed. The measurement of dataset quality and GNN calculation quality is carried out by utilizing the previously proposed $prpDist$ and $prDiff$ calculations, calculating the $prpDist$ and $prDiff$ variables twice (exploiting the numerical reproducibility error in FP calculations) and comparing the differences between the two results to determine the quality of the GNN calculation. There are five variables generated from this proposed method. These variables are CQP, CQD, GDCM, GDCV, and DSA95. CQP and CQD are the results of measuring the quality of the GNN calculation, influenced by a combination of the quality of the FP calculation of the PU being used and the complexity of the GNN algorithm (GNN type) being applied. GDCM and GDCV are the results of a combination of dataset quality (the presence or absence of data duplication) and the sensitivity of the GNN to be able to distinguish each feature contained in the dataset. Meanwhile, DSA95 is the result of a combination of all factors that influence the previous four variables.

By combining multiple GNN configurations in the measurement, the data duplication value in the dataset can be more accurate than previously proposed methods. By combining certain GNN configurations, a comparison of the calculation capabilities between CPU and GPU can also be obtained. This comparison can be obtained based on measuring the initial location of the FP calculation noise after the decimal point. By using GIN and the 1NMC abstraction, calculations using the CPU are observed to be more precise and repeatable compared to using the GPU, so that the maximum amount of data that can be distinguished by GNN in a dataset for big data applications is 25 to 100 times larger.

Since all measurements were taken before the training process began (all measurements used a GNN with random weights that had not been trained and did not involve the training process and weight changes), and the measurement process did not utilize class information in the dataset, a correlation between the GNN calculation quality measurements and the GNN's performance after the training process (training accuracy and validation accuracy) could not be observed. Studying the behavior of measurement results for extreme configurations that can produce very significant random measurements is also the subject of future work.

ACKNOWLEDGMENT

We would like to express our gratitude to Mrs. Toyama for all the support she provided, including the research environment.

REFERENCES

[1] J. Lu, et all, “Automatic Op-Amp Generation From Specification to Layout,” in IEEE Transactions on Computer-Aided Design ff Integrated Circuits And Systems, vol. 42, no. 12, December 2023.

[2] Z. Zhao, L. Zhang, “Analog Integrated Circuit Topology Synthesis With Deep Reinforcement Learning,” IEEE Transactions on Computer-Aided Design ff Integrated Circuits And Systems, vol. 41, no. 12, December 2022.

[3] Z. Zhao, J. Luo, J. Liu and L. Zhang, "Signal-Division-Aware Analog Circuit Topology Synthesis Aided by Transfer Learning," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 42, no. 11, pp. 3481-3490, Nov. 2023.

[4] I. Guven, M. Parlak, D. Lederer and C. De Vleeschouwer, "AI-Driven Integrated Circuit Design: A Survey of Techniques, Challenges, and Opportunities," in IEEE Access, vol. 13, pp. 167364-167389, 2025.

[5] K. Arimura, R. Miyauchi and K. Tanno, “Systematic Offset Voltage Reduction Methods Using Half-Circuit of Input Stage in the Two-Stage CMOS Operational Amplifiers and Comparators,” IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2025

[6] F. Jiao, S. Montano, C. Ferent, A. Doboli, and S. Doboli, “Analog circuit design knowledge mining: Discovering topological similarities and uncovering design reasoning strategies,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 34, no. 7, pp. 1045–1058, Jul. 2015.

[7] H. Li, X. Liu, F. Jiao, A. Doboli, and S. Doboli, “InnovA: A cognitive architecture for computational innovation through robust divergence and its application for analog circuit design,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 37, no. 10, pp. 1943–1956, Oct. 2018.

[8] A. A. Mannan, K. Tanno, “Graph Neural Network Output for Dataset Duplication Detection on Analog Integrated Circuit Recognition System,” International Journal of Advanced Computer Science and Applications, vol. 16, no. 5, 2025.

[9] L. Ozyilmaz, T. Yildirim, “Sensitivity analysis for conic section function neural networks,” Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000.

[10] T. Xu, A. L. Cox and S. Rixner, "Design and accuracy trade-offs in Computational Statistics," 2025 IEEE International Symposium on Workload Characterization (IISWC), Irvine, CA, USA, 2025.

[11] Tiong-Hwee Goh, "Semantic extraction using neural network modelling and sensitivity analysis," Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan), Nagoya, Japan, 1993.

[12] K. Hakhamaneshi, M. Nassar, M. Phielipp, P. Abbeel and V. Stojanovic, “Pertaining Graph Neural Network for Few-Shot Analog Circuit Modeling and Design,” IEEE Transactions On Computer-Aided Design of Integrated Circuits and Systems, Vol. 42, NO. 7, JULY 2023.

[13] T. N. Kipf, M. Welling, “Semi-supervised classification with graph convolutional networks,” 5th International Conference on Learning Representations (ICLR), 2017.

[14] K. Xu, W. hu, J. Leskovec and S. Jegelka, ”How Powerful are Graph Neural Networks?,” International Conference on Learning Representations (ICLR) 2019.

[15] S. Brody, U. Alon and E Yahav, “How Attentive are Graph Attention Networks?,” The Tenth International Conference on Learning Representations (ICLR), 2022.

[16] W. L. Hamilton, R. Ying and J. Leskovec, “Inductive representation learning on large graphs,” 31st Annual Conference on Neural Information Processing Systems, NeurIPS, 2017, pp. 1025–1035.

[17] "IEEE Standard for Floating-Point Arithmetic," in IEEE Std 754-2019 (Revision of IEEE 754-2008) , pp.1-84, 22 July 2019.

[18] S. Chowdhary, S. Nagarakatte, “Parallel shadow execution to accelerate the debugging of numerical errors,” Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, pp.615–626, 2021.

[19] S. Chowdhary, S. Nagarakatte “Fast shadow execution for debugging numerical errors using error-free transformations,” Proceedings of the ACM on Programming Languages 6 (OOPSLA2), pp. 1845–1872, 2022.

[20] Y. Tan, et all, “Computing Floating-Point Errors by Injecting Perturbations,” arXiv:2507.08467v1 [cs.SE] 11 Jul 2025.

[21] P. Langlois, R. Nheili and C. Denis, "Numerical reproducibility: Feasibility issues," 2015 7th International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 2015.

[22] S. Shanmugavelu, et all, “Impacts of floating-point non-associativity on reproducibility for HPC and deep learning applications,” SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis, Atlanta, GA, USA, pp. 170-179, 2024.

[23] S. F. J. Apostal, D. Apostal and R. Marsh, "Containers and Reproducibility in Scientific Research," 2018 IEEE International

Conference on Electro/Information Technology (EIT), Rochester, MI, USA, 2018.

[24] H. Al-Rikabi, B. Renczes, "Floating-Point Roundoff Error Analysis in Artificial Neural Networks," 25th IMEKO TC-4 International Symposium on Measurement of Electrical, Brescia, Italy / September 12-14, 2022.

[25] T. Hadjam, A. M. Salah, M. B. Nedjma, M. Abdelmadjid and H. Hamil, "FPGA implementation of a Convolutional Neural Network for image classification," 2022 2nd International Conference on Advanced Electrical Engineering (ICAEE), Constantine, Algeria, 2022.

[26] J. Yun, B. Kang and Z. Fu, "The Hidden Power of Pure 16-bit Floating-Point Neural Networks," arXiv:2301.12809v2 [cs.LG] 3 May 2024.

[27] A. A. Mannan, K. Tanno, "Netlist Feature Extraction for CMOS Analog Circuit Design Warning System," 23rd International Conference on Machine Learning and Cybernetics (ICMLC), pp. 127-132, 2024.

AUTHORS' PROFILE



Arif Abdul Mannan was born in Malang, Indonesia, on January 23, 1988. Currently, he is a PhD student in University of Miyazaki, Japan. He received S.T degrees from the Faculty of Engineering, Brawijaya University in 2010. He also received master's degrees from Double-Degree Program of Faculty of Engineering in Brawijaya University and Miyazaki University for his M.T and M.E degree in 2013. His research interests are circuit analysis on digital circuits (microprocessor design and application, digital CMOS integrated circuit design, FPGA & VHDL), and analog circuit (analog CMOS integrated circuit design including multi-valued logic). He can be contacted at email: arifabdulmannan@ub.ac.id.



Koichi Tanno was born in Miyazaki, Japan, on April 22, 1967. He received B.E. and M.E. degrees from the Faculty of Engineering, University of Miyazaki, Miyazaki, Japan, in 1990 and 1992, respectively, and Ph. D degree from the Graduate School of Science and Technology, Kumamoto University, Kumamoto, Japan, in 1999. From 1992 to 1993, he joined the Microelectronics Products Development Laboratory, Hitachi, Ltd., Yokohama, Japan. He contributed to the research on low-voltage and low-power equalizers for reading channel LSI of hard disk drives. In 1994, he joined the University of Miyazaki, where he is currently a Professor in the Faculty of Engineering, and a Vice-president (Collaborative Research and Community Cooperation). His main research interests are in analog integrated circuit design, nano-mist sprayers, and its application. Dr. Tanno is a senior member of IEEE. He can be contacted at email: tanno@cc.miyazaki-u.ac.jp.

LayCoder: UI Layout Completion with an Encoder-Only Transformer and Layout Tokenizer

Iskandar Salama, Luiz Henrique Mormille, Masayasu Atsumi
Soka University, Tokyo, Japan

***Abstract*—The growing complexity of user interface (UI) design calls for effective methods to understand, complete, and refine layout structures. While prior work has focused predominantly on generating UI layouts from scratch, completing partially designed interfaces is equally critical, particularly in iterative design workflows and scenarios involving incomplete prototypes. In this study, we address the UI layout completion task for mobile app screens using an encoder-only transformer architecture with masked modeling and a layout tokenizer. By representing UI elements as discrete tokens, we formulate layout completion as a sequence prediction problem that leverages global context to infer missing components. We evaluate our approach on subsets of the RICO dataset designed with varying constraints on UI element types and spatial overlap, and report results using standard layout metrics: Coverage, Intersection over Union (IoU), Max IoU, and Alignment. The experiments demonstrate that the proposed method achieves substantial improvements over LayoutFormer++, a widely adopted baseline in UI layout generation, particularly in Coverage, and in several cases, IoU, Max IoU, and Alignment. Additional experiments on noise-reduced subsets reveal that dataset curation can enhance spatial consistency but may also reduce Coverage, reflecting an inherent trade-off between completeness and structural precision. These findings highlight both the effectiveness and the limitations of encoder-only masked modeling for layout completion, and underscore the importance of balancing model design with dataset construction when tackling complex UI design tasks.**

***Keywords*—Layout completion; deep learning; encoder-only transformers; masked language modeling; tokenization**

I. Introduction

In the ever-evolving landscape of mobile applications, user interfaces (UIs) play a pivotal role in determining user engagement and satisfaction. As mobile apps have evolved, the design of user interfaces has become more sophisticated, focusing on enhancing user experience through intuitive and visually appealing layouts. This shift towards more user-centric designs matters significantly because studies have shown that user-friendly interfaces can lead to better user retention and increased app usage [1].

Consequently, the demand for efficient and innovative UI design techniques has increased, prompting the integration of generative AI (Gen-AI) into the design process [2], [3]. Generative models offer substantial benefits in UI development, including improved speed and automation [4], reducing the time and effort required for manual layout design.

Historically, research in automatic UI layout modeling has focused primarily on generating complete layouts from scratch [5]. However, the ability to complete partially designed interfaces is equally important—particularly in scenarios that involve iterative refinement or require automatic completion based on existing components [6].

Formally, we define UI layout completion as a conditional structured prediction problem under partial observability. Given a partially observed layout L_{obs}, where a subset of UI elements are missing or masked, the objective is to predict the missing elements L_{miss}. The completed layout $\hat{L}$ should remain spatially and semantically consistent with realistic UI design patterns.

This task differs from unconditional layout generation, as the model must preserve existing components while inferring plausible missing structure conditioned on the visible layout context. This formulation reflects practical early-stage design scenarios, where interfaces are incomplete and require context-aware completion to support iterative refinement.

To address this problem, we investigate the potential of encoder-only transformer models to infer missing components from partial UI layouts. Transformers [7] have become a powerful tool across multiple domains, notably in natural language processing and, more recently, layout modeling tasks. Their ability to model long-range dependencies and process sequences in parallel makes them particularly effective for capturing the structure of complex inputs such as UI layouts.

Encoder-only transformers are particularly well-suited for tasks requiring contextual understanding. In layout completion, they can process all visible components and their interrelationships in a single pass, enabling the model to predict missing elements based on global layout context. This capability aligns closely with masked language modeling (MLM) [8], where missing tokens are reconstructed from surrounding information.

Although our formulation is inspired by MLM, LayCoder is not a straightforward adaptation of BERT to discretized layout tokens. Unlike natural language, UI layouts are inherently spatial and structured, requiring joint reasoning about geometry, element semantics, and layout coherence. In particular, each UI component cannot be represented as a single token, but rather as a set of correlated spatial and semantic attributes.

To enable Transformer-based modeling of UI layouts, we therefore convert each layout into a structured sequence of discrete tokens using a custom tokenization mechanism, which we term the *Layout Tokenizer*. The first token represents the *layout category*, and each UI element is encoded through multiple attribute tokens—namely, *centerX*, *centerY*, *width*, *height*, and the *component label*. These attributes are individually mapped into discrete token IDs from a predefined vocabulary.

This representation allows LayCoder to perform masked

completion at the level of structured UI elements, where each missing component corresponds to multiple correlated spatial and semantic attributes. This element-level grouping distinguishes LayCoder from standard MLM models by enabling masked prediction over multi-attribute UI components rather than single discrete symbols.

The Layout Tokenizer operates on a predefined vocabulary consisting of 675 tokens. This layout-specific tokenization transforms the structured UI into a flat sequence of tokens, enabling Transformer models to process and complete UI layouts in a manner similar to how language models predict missing words [9].

Our approach is evaluated using the RICO dataset [25], one of the most comprehensive resources for mobile UI layouts. It contains over 66,000 UI screens and more than 3 million annotated elements, making it a suitable benchmark for layout modeling and completion.

Our main contributions are summarized as follows:

- We propose LayCoder, an encoder-only transformer framework that formulates UI layout completion as masked structured prediction over partially observed mobile app layouts.
- We introduce a custom Layout Tokenizer, which represents UI elements as discrete sequences of spatial and semantic tokens. This design enables Transformer models to leverage global context for inferring missing components in incomplete layouts.
- Through experiments on subsets of the RICO dataset, we demonstrate that the proposed method achieves substantial improvements over LayoutFormer++ [24], a previously leading approach in UI layout generation, with consistent gains in Coverage and notable improvements in IoU and Alignment across multiple test scenarios. The empirical results underscore the efficacy of encoder-only masked modeling in capturing spatial and semantic dependencies within layout structures.
- We further analyze the role of dataset curation by evaluating on noise-reduced subsets. The findings reveal that filtering improves spatial consistency but can reduce Coverage, highlighting the observed trade-off between layout completeness (Coverage) and structural precision (IoU and Alignment) in UI layout completion.

II. Related Work

The automated modeling of UI layouts has attracted significant research attention, with approaches spanning various architectural paradigms and task formulations. To provide a comprehensive overview of this rapidly evolving field, we organize prior work along two primary dimensions: 1) task type, distinguishing between layout generation (synthesizing complete layouts from scratch) and layout completion (predicting missing elements in partial layouts); and 2) architectural approach, encompassing GANs [10], VAEs [11], diffusion models [12], and transformer-based methods [13]. Within transformer-based approaches, we further distinguish between encoder-only, decoder-only, and encoder-decoder variants [14], as these architectural choices significantly impact model capabilities and computational efficiency.

A. Layout Generation Approaches

Layout generation has been addressed through various architectural approaches. GAN-based methods like Akin [15] and LayoutGAN [16] utilize adversarial training to synthesize realistic UI wireframes and graphic layouts by modeling geometric relations of 2D elements.

VAE-based approaches, including LayoutVAE [17], LayoutVQ-VAE [18], and VTN [19], offer robust frameworks for generating diverse layouts by learning latent representations and capturing high-level relationships between layout elements.

Recent advancements in layout generation have seen the emergence of diffusion models as a powerful approach for synthesizing structured layouts with high quality and diversity through discrete denoising processes. Notable examples include LayoutDM [20], [22], LayoutDiffusion [21], and layout decoupled diffusion [23].

Transformer-based models have emerged as a powerful approach for layout generation. Gupta et al. [5] introduced LayoutTransformer, a framework that uses self-attention mechanisms to model contextual relationships between layout elements across various domains, enabling the generation of novel scene layouts from an empty set or an initial seed of primitives.

While these layout generation methods have demonstrated impressive capabilities in synthesizing full-screen UI designs, they primarily address the creation of complete layouts from scratch. However, in many practical design scenarios, designers work with partially specified interfaces that require completing existing structures rather than generating entirely new ones.

B. Layout Completion Approaches

A smaller but growing body of work targets layout completion, which is more aligned with iterative design workflows. Li et al. [6] proposed a model for the auto-completion of user interfaces, focusing on assisting the design process by predicting the remaining UI elements in a partial layout. This approach utilizes Transformer-based tree decoders—Pointer and Recursive Transformer—representing a decoder-only architecture to handle the hierarchical structures and 2-dimensional placements of UI elements. Experiments on a public dataset show the effectiveness of these models, which significantly ease the effort of UI designers and developers. The proposed metrics for measuring tree prediction accuracy further contribute to deep learning research in layout prediction.

LayoutFormer++ [24] provides a unified framework for both generation and completion by serializing user constraints and employing a transformer encoder–decoder model. It introduces a constraint serialization scheme to represent various user constraints as token sequences in a predefined format and applies a decoding space restriction strategy to prune predictions that violate user constraints or yield low-quality layouts. This ensures that the generated results adhere to user requirements without sacrificing visual quality.

In contrast to these decoder- and encoder–decoder-based approaches, our work explores an encoder-only transformer formulation for the layout completion task. This design emphasizes parallel processing and self-supervised masked modeling to infer missing elements efficiently. Further details of the proposed method are described in Section III.

While LayoutFormer++ introduces constraint serialization and decoding space restriction to enforce validity during autoregressive generation, LayCoder adopts a fundamentally different approach. Instead of restricting the decoding space step-by-step, our encoder-only formulation relies on global contextual inference through masked modeling. This removes the need for sequential constraint enforcement but places greater emphasis on learning structural regularities directly from data.

Similarly, tree-decoder approaches explicitly preserve hierarchical structure during generation, whereas our flat tokenization strategy prioritizes modeling flexibility and parallel processing. This design choice trades explicit hierarchical inductive bias for scalability and simpler inference, allowing LayCoder to complete layouts in a single forward pass.

III. Proposed Method

Our proposed method section consists of three main components: 1) the definition of key concepts that characterize UI elements, 2) the tokenization strategy that converts structured UI data into a sequential format, and 3) the masked language modeling approach used to train the transformer.

An overview of the system architecture is provided in Fig. 1.

A. Key Concepts

A user interface (UI) consists of a hierarchy of elements, such as buttons, text boxes, images, and containers. These elements are organized as a tree, where some elements may contain others. We classify UI elements into two categories:

- Leaf elements: are elements that do not contain any child elements. Examples include buttons, text inputs, icons, and images.
- Non-leaf elements: are elements that contain one or more child elements. Examples include containers, lists, modals, and cards.

Each element, whether a leaf or non-leaf, is characterized by the following attributes:

- Spatial Coordinates (Bounds): Each element is represented by four values: x_1, y_1, x_2, y_2, denoting the top-left and bottom-right corners of its bounding box on an original screen with a resolution of 2560×1440.
- Element Label (Component Label): A categorical identifier that represents the type of the UI component, such as Text, Image, Container, etc.

In addition, each UI layout is associated with a global attribute:

Layout Category: A label indicating the overall purpose or category of the UI screen (e.g., Login, Registration, Catalog).

These attributes form the foundation for representing UI elements in our model and are critical to both the dataset construction and the tokenization process described in the following sub-sections.

B. Tokenization

We first construct a fixed vocabulary consisting of discrete token values used to encode spatial and semantic attributes of UI elements. For spatial attributes, we discretize normalized coordinate values in the range of 0 to 640, resulting in 641 unique tokens. For semantic attributes, we include 25 tokens representing the Component Labels (e.g., Text, Image), and seven tokens corresponding to distinct Layout Categories identified from the Akin framework [15]: Login, Splash, Registration, Catalog, Product page, Cart, and No-category. Additionally, we define two special tokens—one for separating between elements and one for masking elements during training. This results in a total vocabulary size of 675 tokens.

Spatial discretization to the range [0, 640] provides a consistent vocabulary but introduces bounded quantization error. The maximum absolute discretization error per coordinate is 0.5 units in the normalized scale, corresponding to less than 0.08% of the full spatial extent. While empirically sufficient for preserving layout structure, this discretization limits sub-pixel precision and should be interpreted as an approximation of continuous geometry.

Using this predefined vocabulary, we apply our Layout Tokenizer to discretize the numerical attributes and categorical labels into tokens. Each numerical attribute (e.g., centerX or width) is mapped to a corresponding token ID selected from the constructed vocabulary.

Each UI element is represented by five tokens: four for the spatial attributes (*centerX*, *centerY*, *width*, *height*) and one for the *Component Label*. Each UI layout is then represented as a sequence of tokens, referred to as the *Layout Token Sequence*. As shown in Fig. 2, this sequence starts with the *Layout Category* token, followed by a separator token, and then the *centerX*, *centerY*, *width*, *height*, and *Component Label* tokens for each UI element. Tokens from distinct elements are separated using the special separator token. Consequently, the total length of the sequence is *6N+2*, where N denotes the number of elements in the UI.

A practical aspect of this serialization is the ordering of UI elements. In our implementation, elements follow the order obtained from traversing the hierarchical RICO JSON tree. Although the rendered layout itself is permutation-invariant, Transformer encoders process sequences and are therefore sensitive to element order. We use this traversal order as a deterministic scheme to ensure reproducibility.

Importantly, this representation allows LayCoder to perform masked completion at the level of structured UI elements, where each missing component corresponds to multiple correlated spatial and semantic attributes. This element-level grouping distinguishes the task from a naive application of token-level masked language modeling, as reconstruction operates over structured components rather than independent tokens.

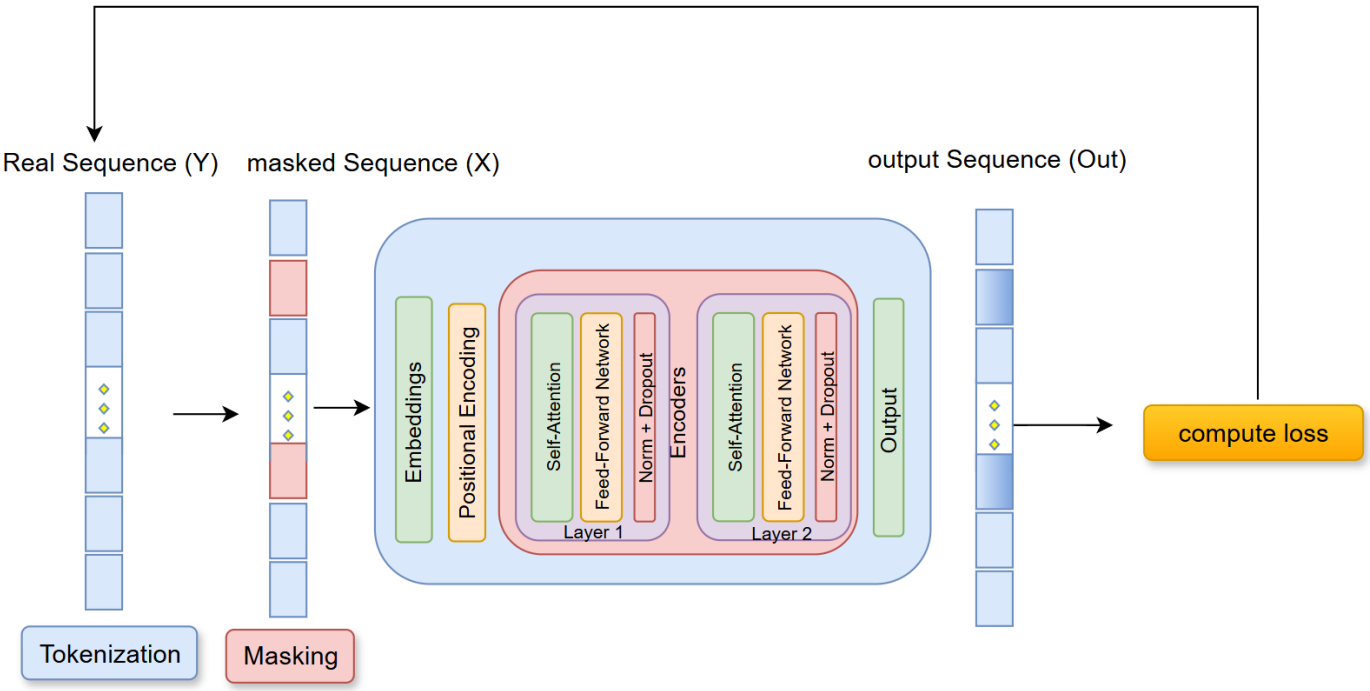


Fig. 1. System architecture: Detailed illustration of the proposed method, including layout tokenization, the masking mechanism, and the two-layer Transformer encoder used for UI layout completion. The architecture highlights how the input token sequence is processed through the encoder to predict the masked UI elements.

This layout-specific tokenization transforms structured UI layouts into flat token sequences that can be processed by the Transformer architecture and can be applied to other datasets with similar structural representations.

C. Encoder Architecture and Masked Training

The architecture follows a standard Transformer encoder stack. Each layer consists of multi-head self-attention, residual connections with layer normalization, and a position-wise feed-forward network. Self-attention enables each token to attend to all other tokens in the layout sequence, capturing long-range spatial and semantic dependencies across UI elements in a single forward pass.

Our approach utilizes an encoder-only transformer architecture, which consists of 2 stacked Transformer layers, each with a hidden size of 512. Within each layer, multi-head self-attention is implemented using 8 attention heads, and the feedforward sublayer has an intermediate dimension of 1024. Layer normalization is applied before each sublayer, and a dropout rate of 0.5 is used within each encoder block to mitigate overfitting. Input tokens are embedded into 512-dimensional vectors using a trainable embedding layer, and positional encodings are added via sinusoidal functions followed by dropout (p = 0.1). The final output is passed through a linear projection layer that maps from the hidden dimension back to the vocabulary size.

We adopt an encoder-only architecture rather than an encoder–decoder model for two reasons. First, masked modeling allows parallel reconstruction of missing elements without autoregressive decoding, reducing inference latency. Second, completion in our setting operates over a partially observed structured sequence, making bidirectional context modeling more natural than left-to-right generation. This design choice favors global contextual reasoning over sequential constraint enforcement.

We incorporate masked language modeling as a core technique of our training objective. It is applied by randomly selecting certain tokens within the layout token sequence and replacing them with our special mask token. During training, this process helps the model learn contextual dependencies between UI elements and their attributes.

Specifically, in the training phase, a random percentage k of tokens from each layout sequence is masked, where $k \in [5\%, 40\%]$. The masked tokens include UI component labels and spatial attributes, while category tokens and separator tokens are excluded to preserve structural consistency. The exact value of k is resampled at the beginning of each epoch. This strategy introduces variability and prevents the model from overfitting to a fixed masking pattern. An example of this token-level masking is illustrated in Fig. 3.

However, during inference, we adopt a masking strategy aligned with LayoutFormer++ [24]. Instead of masking individual tokens independently, we mask complete UI elements—i.e., an element and all of its associated attributes (*Bounds* and *Component Labels*). A fixed masking ratio is applied at the element level rather than the token level. For instance, given a UI layout with 5 elements and a masking ratio of 40%, we mask 2 entire elements, including all tokens representing their attributes. This approach is illustrated in Fig. 4.

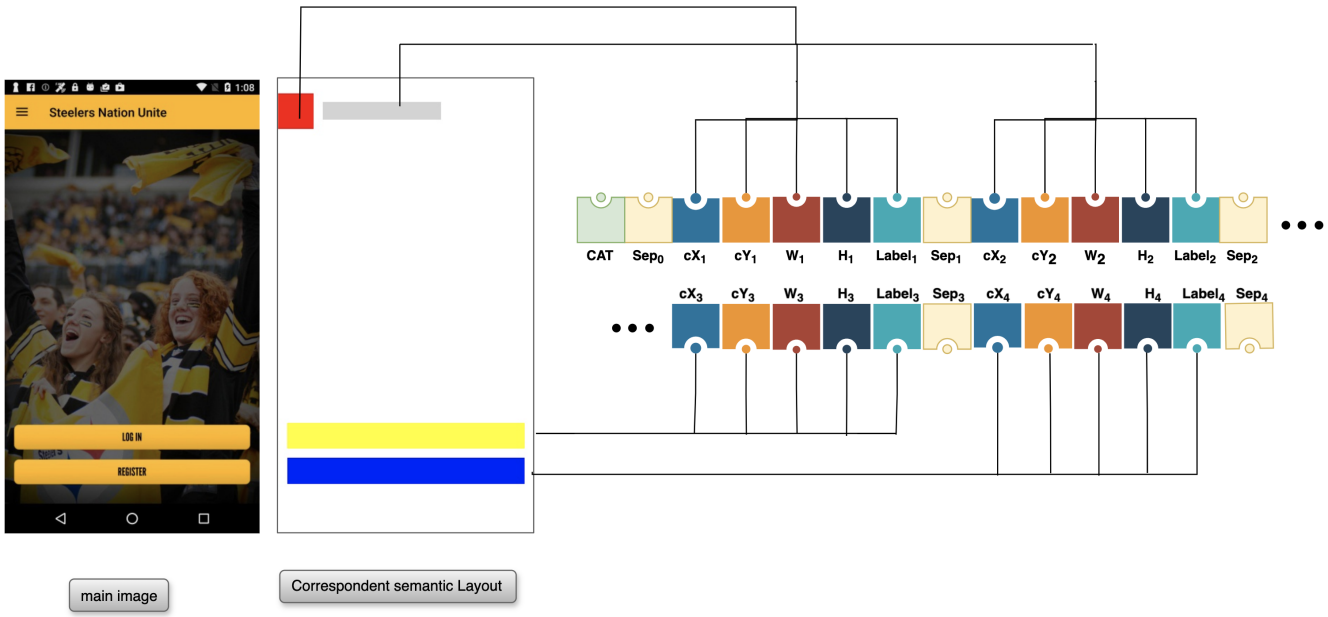


Fig. 2. Overview of the tokenization process using UI as an example with four elements. The Layout Token Sequence begins with the layout category token (CAT), followed by a separator token (Sep). For each UI element, five tokens are appended to the sequence: CenterX (cx), CenterY (cy), width (W), height (H), and the component label (Label). Although the illustration shows four elements, the same procedure applies to layouts with any number of elements.

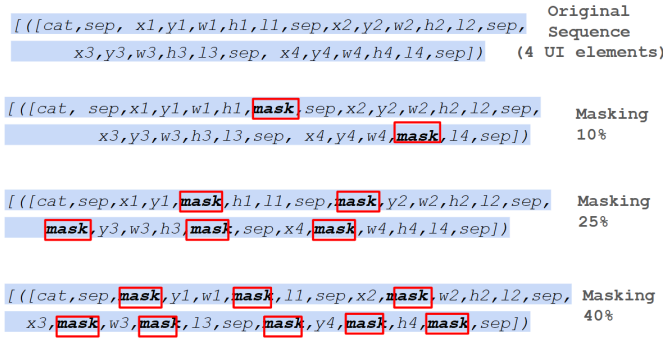


Fig. 3. The masking procedure is applied randomly to each sample in every epoch using a special mask token. The layout category (CAT) and separator (Sep) tokens are never masked; only element attributes are subject to masking. For element i, x_i, y_i, w_i, h_i, and l_i denote centerX, centerY, width, height, and component label, respectively.

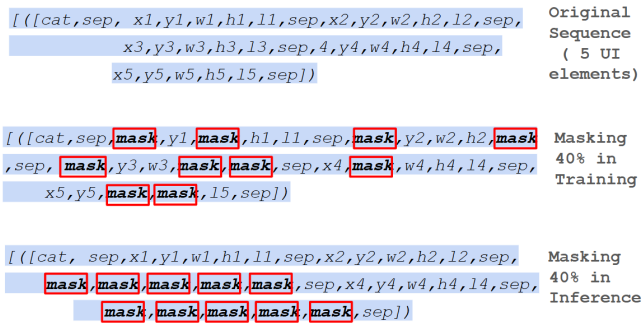


Fig. 4. Comparison of masking strategies between training and inference. During training, masking is applied at the token level, where individual coordinate or label tokens may be masked. During inference, masking is applied at the element level, where all tokens corresponding to a UI element (its coordinates and label) are masked together.

The reason we adopted a different masking strategy during training, as compared to the inference stage, is to learn fine-grained relationships between UI elements, while ensuring that the inference masking setting remains aligned with downstream layout completion scenarios.

The total number of trainable parameters in our configuration is approximately 4.55 million. For a sequence length T, the computational complexity of each encoder layer is $O(T^2)$ due to self-attention. This remains manageable in our setting since layouts are restricted to at most 20 elements (i.e., $T \leq 6N + 2$), enabling efficient training and inference on a single GPU.

IV. Dataset

In this research, we use the RICO dataset [25], a large-scale collection of over 66,000 UI screen images and more than 3 million individual UI elements, collected from a wide variety of Android mobile applications. RICO is one of the most comprehensive and diverse benchmarks for mobile user interfaces, providing rich resources for UI layout generation and completion design tasks.

Each UI screen in the dataset is represented in three formats: 1) a high-resolution screenshot, 2) a corresponding *semantic layout image* that visualizes the structure of the UI using colored bounding boxes (where each color represents a different UI element type, and the size of each box matches



Fig. 5. UI labels color palette: All UI component labels and their corresponding RGB color assignments.

the element's actual size on the original screen), and 3) a JSON file containing detailed metadata. This metadata includes attributes, bounding box coordinates, Android class names, and the hierarchical relationships among elements.

To ensure consistency across all visualizations, the *semantic layout images* use a predefined *UI color palette*, in which each component label is assigned a unique RGB color. Fig. 5 presents the complete set of component labels used in this research together with their corresponding color codes.

A. Re-organizing the RICO Dataset

Following prior work [24], [5], [15], we do not use the RICO dataset in its raw form. Instead, we construct subsets specifically designed to meet the requirements of our proposed method. This involves re-organizing the dataset structure and preprocessing the raw UI layout data accordingly.

As previously described, each UI screen in RICO is accompanied by a JSON file that defines its layout as a hierarchical tree of UI elements. By recursively traversing this tree, we determine the depth of each element and classify them into two categories: leaf elements (elements with no child nodes) and non-leaf elements (elements with one or more children), as described in Section III-A. These distinctions play a central role in how we construct dataset variants for our experimental setups.

To ensure fair comparison with prior work, we build upon the same processed LayoutFormer++ [24] dataset and derive two subset variants, each tailored to a specific experimental objective.

- DS1: This subset corresponds to the version used by LayoutFormer++ during its preprocessing pipeline, in which all UI layouts with more than 20 elements were removed. The result is a curated dataset where each instance contains at most 20 elements. *DS1* consists of 31,694 training samples (221,858 elements), 3,729 test samples, and 1,865 validation samples.
- DS2: This subset is a refined version of *DS1*. Two additional filters are applied: 1) only *leaf elements* are retained, while all *non-leaf elements* are discarded, and 2) any layouts containing spatially overlapping elements are excluded to ensure clean, non-overlapping compositions.

Since our experiments are built upon the same processed subset and data-splitting protocol introduced in LayoutFormer++, the train/test separation follows the established benchmark setting. Layouts are split at the screen-instance level such that no identical UI structures appear across DS1 training and DS2 testing.

However, because RICO contains many visually similar screens within the same application, this protocol does not fully eliminate potential app-level similarity. Consequently, the reported results should be interpreted as generalization across screen instances under the standard benchmark split, rather than strict generalization across entirely unseen applications.

The additional filtering applied in DS2 is motivated by reducing what we refer to as layout noise, namely structural artifacts that may obscure clear spatial patterns. Such noise may arise both from inherent layout complexity and from dataset construction artifacts. In our setting, this includes overlapping elements and non-leaf elements, which can introduce redundancy or hierarchical complexity that is not essential for the layout completion task. By filtering out these elements, we simplify the input representation and promote more consistent spatial modeling and better generalization.

At the same time, this simplification comes with an important modeling trade-off, since UI layouts are inherently hierarchical in nature. However, removing non-leaf elements also eliminates explicit parent–child relationships that encode containment and compositional structure within the UI hierarchy. As a consequence, the model no longer directly observes grouping patterns, nesting depth, or structural dependencies that are central to UI semantics. The DS2 configuration should therefore be interpreted as a structural ablation that isolates spatial reasoning from hierarchical reasoning, rather than as a universally superior representation.

B. Preprocessing

In the preprocessing stage, we normalize the *Bounds* to fit within a 0 to 640 scale, corresponding to the maximum height of the *semantic layout images*. Subsequently, transformations are applied to convert these Bounds into *centerX*, *centerY*, *width*, and *height*, with the former two indicating the center of the element, and the latter two indicating the width and height of the UI element's bounding box.

By transforming the bounds coordinates and normalizing their values, we ensure that the positional information of UI elements is accurately represented within a consistent scale.

The transformation to center-based coordinates allows for more intuitive spatial representations, facilitating the model's ability to understand the layout and structure of the UI elements [27]. The vocabulary described in Section III-B is also constructed as part of this preprocessing step.

V. Evaluation Metrics

In our study, we employed the same evaluation methods used by previous researchers. While our experiments focus on layout completion and some prior studies focus on layout generation, we use the same evaluation metrics in both cases, as they are equally applicable to assessing spatial and structural layout quality. The metrics used in this study are explained as follows:

Coverage Metric is calculated as the proportion of the UI layout area that is occupied by layout elements. In other words, it is the area covered by elements divided by the total area of the canvas (UI layout). Higher scores indicate better coverage, reflecting layouts that are more fully and effectively populated with UI components.

IoU measures the degree to which layout elements overlap within a completed layout. Specifically, it is the average Intersection over Union between each pair of elements (bounding boxes) within the same layout. Mathematically, IoU is defined as the area of Intersection divided by the area of the Union between two bounding boxes, providing a normalized measure that accounts for both the intersecting and non-overlapping regions. The computation of IoU between two elements *A* and *B* is shown in the following equation:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

where, $|A \cap B|$ denotes the area of their intersection, and $|A \cup B|$ denotes the area of their union. Lower scores indicate better IoU, ensuring that no overlap is occurring.

Max IoU provides insight into the maximum overlap any element in a layout has with other elements. It is obtained by first computing the pairwise IoU between each element within a UI, then, for each element, the maximum IoU value is retrieved, and finally, those values are summed and averaged. Similar to IoU, lower Max IoU values are preferable.

Alignment is a metric originally introduced in the Attribute-conditioned Layout GAN paper [26] and later adopted in LayoutFormer++ [24] to evaluate the spatial alignment of UI elements. For each layout element, we compute six reference points: the left, center, and right positions along the horizontal axis, and the top, center, and bottom positions along the vertical axis. Pairwise distances are then calculated between each of these points and the corresponding points of all other elements in the layout. For each element, the minimum of these distances is taken as its alignment error with respect to the rest of the layout.

Finally, the alignment scores are summed and averaged, producing a single scalar value per layout. Lower scores indicate better alignment, reflecting a more structured and visually coherent layout.

These metrics provide a comprehensive evaluation framework by measuring element distribution, spatial overlap, and spatial coherence within each layout. Specifically, Coverage reflects the overall distribution and presence of elements across the layout, IoU and Max IoU quantify the degree of element overlap, and Alignment evaluates spatial coherence.

At the same time, these metrics primarily capture geometric and structural properties of the completed layouts. They do not directly measure semantic correctness of element identities. A completion may achieve strong spatial alignment while predicting an incorrect component label. Therefore, the current evaluation emphasizes spatial fidelity rather than full element-wise reconstruction accuracy.

VI. Experiments

In this section, we trained our models on two curated subsets of the RICO dataset, *DS1* and *DS2*, introduced in Section IV. Our models were trained to infer and complete UI layouts with up to 40% of elements randomly masked. More specifically, we systematically varied the masking rate of UI elements from 5% to 40% during training and inference, in order to evaluate the models with different levels of masked information.

Furthermore, model performance is assessed using the evaluation metrics described in the previous Section: Coverage, IoU, Max IoU, and Alignment. The original, unmasked UI layouts serve as ground-truth references, and model performance is evaluated using the defined metrics by comparing the predicted layouts with these references.

A. Model Variants

We trained two variants of our proposed model using the aforementioned datasets:

- LayCoder — a model trained on *DS1*, containing up to 20 elements per UI layout, and including both leaf and non-leaf nodes. The model was trained using the same preprocessing steps as [24].
- LayCoder-NonOver-Leaves — a model trained on *DS2*, a refined version of DS1 containing only leaf nodes and no overlapping elements. This model leverages the simplified structure of the data (leaf-only nodes and non-overlapping elements) to evaluate the impact of dataset noise reduction on model performance.

We aim with the first variant to demonstrate that our proposed model, *LayCoder*, achieves superior performance compared to the baseline, *LayoutFormer++*, across standard layout evaluation metrics.

With *LayCoder-NonOver-Leaves*, we specifically test the hypothesis that reducing layout noise—through the exclusion of overlapping elements and non-leaf nodes—can enhance model performance. Though we evaluate both models on the same test set (the testing subset of *DS2*). *LayCoder* was trained on noisier and more diverse layouts (*DS1*), while *LayCoder-NonOver-Leaves* was trained on a less noisy dataset (*DS2*). We ensure as well that the test set of *DS2* does not overlap with the training data of *DS1*. This setup allows us to examine the importance of curating the data to contain only elements relevant to our task and measure its impact on the model's ability to generalize and accurately infer UI structures.

B. Experimental Setup

All experiments were conducted on an NVIDIA RTX 3090 GPU. During training, random masking between 5% and 40% was applied to the token sequence, as described in the Method section. We adopted cross-entropy loss, computed only over the masked tokens; unmasked tokens, as well as special tokens and layout category tokens, were excluded from the loss calculation. The model predicts each masked token from a predefined vocabulary of 666 discrete tokens, which includes 641 tokens representing discretized bounding box attributes and 25 tokens corresponding to element labels. The model is trained using the Adam optimizer with an initial learning rate of 1e-4. A batch size of 64 is used, and training proceeds for 1000 epochs with early stopping based on validation loss, which is monitored every 100 epochs.

VII. Results and Discussion

A. Quantitative Results

Experimental results are detailed in Table I and Table II. In these tables, the row labeled as *Real Data* indicates the mean values for each evaluation metric computed over the unmasked test subsets of DS1 and DS2, serving as the reference baseline. These values represent the expected metric scores for complete, unmasked layouts. The preceding rows, labeled as *10% masking*, *25% masking*, and *40% masking*, indicate the performance of each model when completing UI designs with 10%, 25%, and 40% of masked elements, respectively.

In Table I, we compare the performance of our first model, the *LayCoder*, with the *LayoutFormer++* [24] on the testing subset of *DS1*, which includes a mix of leaf and non-leaf elements.

As introduced in Section V, the performance of the models is evaluated using the following metrics: Coverage, IoU, Max IoU, and Alignment. Furthermore, the Δ value for each metric denotes the absolute difference between the layouts produced by each model and the reference baseline:

$$\Delta = |\text{Predicted} - \text{Baseline}| \tag{2}$$

Therefore, a smaller Δ indicates that the predicted layouts closely approximate the original data.

In the first row of Table I, when masking 10% of the data, we observe a substantial performance difference, particularly in Coverage. *LayCoder* achieves a score very close to the real data, with $\Delta = 1.76$, whereas *LayoutFormer++* shows a much higher $\Delta = 32.38$, indicating severe under-coverage. For IoU and Max IoU, *LayCoder* also achieves smaller Δ values (0.002 and 0.013, respectively) compared to *LayoutFormer++* (0.022 and 0.065). These results indicate closer alignment with the reference layouts in terms of overlap. In terms of Alignment, *LayCoder* achieves a perfect $\Delta = 0$, meaning that its completions under 10% masking are equivalent to the baseline layouts without masking. This outcome is consistent with the low masking ratio, where fewer elements are removed and reconstruction is less challenging.

As the masking ratio increases to 25%, *LayCoder* continues to outperform *LayoutFormer++* across all metrics. For Coverage, LayCoder's $\Delta = 3.20$ is still much closer to the real data than LayoutFormer++ ($\Delta = 32.17$). Similarly, for IoU ($\Delta = 0.007$ vs. 0.022), Max IoU ($\Delta = 0.037$ vs. 0.064), and Alignment ($\Delta = 0.001$ vs. 0.004), LayCoder remains stronger. Although its scores show a small degradation compared to the 10% case, this is expected as the masking ratio increases. By contrast, LayoutFormer++ shows little change compared to its 10% performance. This relative stability may be attributed to its additive, sequential training strategy, where elements are generated incrementally. As a result, increasing the masking percentage does not substantially alter its completion behavior.

At 40% masking, both models face more challenging conditions, but *LayCoder* still demonstrates stronger generalization. Its Coverage $\Delta = 3.47$ remains far superior to LayoutFormer++ ($\Delta = 32.37$), suggesting that LayCoder better preserves the overall density of elements under high masking. For Alignment, both models are close ($\Delta = 0.003$ vs. 0.002). On IoU and Max IoU, LayCoder records $\Delta = 0.017$ and 0.072, while LayoutFormer++ remains at 0.024 and 0.066. Although LayoutFormer++ appears slightly better in Max IoU at this level, its Coverage remains very poor ($\Delta \approx 32$ across masking levels).

Since Coverage measures the proportion of the layout canvas occupied by elements, such low values suggest that many UI components are missing, leading to incomplete layouts that deviate substantially from the ground truth. This behavior is closely related to the differing completion paradigms of the two models. LayoutFormer++ generates elements autoregressively in a fixed sequential manner without explicit conditioning on the final number of elements to be produced. As a result, it often yields conservative predictions under high-masking settings, leading to systematic under-coverage and relatively stable scores across masking ratios.

In contrast, LayCoder performs masked reconstruction within a fixed-length sequence, where the number of missing elements is implicitly determined by the masked slots. This formulation encourages preservation of element count and overall layout density. Both models were evaluated under identical masking ratios and metric implementations to ensure consistency.

In summary, across all three masking levels (10%, 25%, 40%), *LayCoder* consistently produces results closer to the real data. Its Coverage is significantly better than *LayoutFormer++* in every experiment, with Δ values ranging from 1.76 to 3.47 compared to around 32 for *LayoutFormer++*. In IoU, Max IoU, and Alignment, *LayCoder* also maintains lower or comparable Δ values, showing stronger robustness under increasing masking ratios.

While the performance of *LayoutFormer++* appears stable across masking levels, this stability reflects the limitations of its autoregressive decoder-based architecture. Because it generates layouts sequentially, the model appears less sensitive to the amount of masked content and shows limited adaptability as masking increases. By contrast, *LayCoder*'s encoder-only masked modeling leverages global layout context, enabling it to adapt better to different masking conditions, preserve spatial structure, and maintain performance much closer to the real-data.

The observed results in Table II indicate that in terms of Coverage, *LayCoder* remains consistently closer to the real

TABLE I. COMPARATIVE RESULTS ON DS1 AT DIFFERENT MASKING LEVELS

Masking	Model	Coverage ↑ (Mean, Δ)	Overlap (IoU) ↓ (Mean, Δ)	Max IoU ↓ (Mean, Δ)	Alignment ↓ (Mean, Δ)
10%	LayCoder	**74.88 ($\Delta = 1.76$)**	0.055 (**$\Delta = 0.002$**)	0.190 (**$\Delta = 0.013$**)	**0.014 ($\Delta = 0.000$)**
	LayoutFormer++	**44.26** ($\Delta = 32.38$)	**0.031** ($\Delta = 0.022$)	**0.112** ($\Delta = 0.065$)	**0.019** ($\Delta = 0.005$)
25%	LayCoder	**73.44 ($\Delta = 3.20$)**	0.060 (**$\Delta = 0.007$**)	0.214 (**$\Delta = 0.037$**)	**0.013 ($\Delta = 0.001$)**
	LayoutFormer++	**44.47** ($\Delta = 32.17$)	**0.031** ($\Delta = 0.022$)	**0.113** ($\Delta = 0.064$)	**0.018** ($\Delta = 0.004$)
40%	LayCoder	**73.17 ($\Delta = 3.47$)**	0.070 (**$\Delta = 0.017$**)	**0.249** ($\Delta = 0.072$)	**0.011** ($\Delta = 0.003$)
	LayoutFormer++	**44.27** ($\Delta = 32.37$)	**0.029** ($\Delta = 0.024$)	**0.111 ($\Delta = 0.066$)**	0.016 (**$\Delta = 0.002$**)
Real Data		**76.64**	**0.053**	**0.177**	**0.014**

A comparative evaluation between LayCoder and LayoutFormer++ at different masking levels using the test set from DS1, which includes both leaf and non-leaf UI elements. Real Data at the bottom of the table represents metrics computed on unmasked UI layouts as a ground-truth reference.

TABLE II. COMPARATIVE RESULTS ON DS2 AT DIFFERENT MASKING LEVELS

Masking	Model	Coverage ↑ (Mean, Δ)	IoU ↓ (Mean = Δ)	Max IoU ↓ (Mean = Δ)	Alignment ↓ (Mean, Δ)
10%	LayCoder-NonOver-Leaves	41.21 (**$\Delta = 0.49$**)	**0.002**	**0.024**	0.0183 (**$\Delta = 0.0004$**)
	LayCoder	**42.30** ($\Delta = 0.60$)	0.003	0.033	**0.0179** ($\Delta = 0.0008$)
25%	LayCoder-NonOver-Leaves	38.80 (**$\Delta = 2.90$**)	**0.008**	**0.058**	0.0204 (**$\Delta = 0.0017$**)
	LayCoder	**41.43 ($\Delta = 0.27$)**	0.015	0.089	**0.0162** ($\Delta = 0.0025$)
40%	LayCoder-NonOver-Leaves	37.00 (**$\Delta = 4.70$**)	**0.014**	**0.092**	0.0198 (**$\Delta = 0.0011$**)
	LayCoder	**41.89 ($\Delta = 0.19$)**	0.032	0.144	**0.0137** ($\Delta = 0.0050$)
Real Data		**41.70**	**0.000**	**0.000**	**0.0187**

Evaluation of LayCoder and LayCoder-NonOver-Leaves under different masking levels, using test samples from DS2, which contains only non-overlapping leaf elements. Note that the ground-truth reference scores (Real Data) for Intersection over Union (IoU) and maximum IoU are zero; thus, the reported mean values are equal to their corresponding Δ values.

data across all masking levels (Δ = 0.60, 0.27, and 0.19 at 10%, 25%, and 40%, respectively), whereas *LayCoder-NonOver-Leaves* shows larger deviations (Δ = 0.49, 2.90, and 4.70). This confirms that the leaves-only, non-overlapping training setup sacrifices Coverage, leading to layouts that omit some elements compared to the ground truth.

In contrast, for the spatial consistency metrics (IoU, Max IoU, and Alignment), *LayCoder-NonOver-Leaves* generally outperforms *LayCoder*. For instance, at 25% masking, it achieves lower Δ IoU (0.008 vs. 0.015) and Δ Max IoU (0.058 vs. 0.089), indicating reduced element overlap and stronger alignment with ground-truth structures. At 40% masking, *LayCoder* achieves a lower mean Alignment score (0.0137 vs. 0.0198). However, *LayCoder-NonOver-Leaves* reports lower Δ values for IoU, Max IoU, and Alignment, indicating stronger spatial consistency with the ground-truth.

Overall, these findings suggest that training on clean, non-overlapping leaves improves spatial consistency in terms of overlaps and relative positioning, but at the cost of reduced Coverage. Thus, the results highlight a clear trade-off: while curated datasets can strengthen spatial precision, they may also lead to incomplete reconstructions that fail to capture all layout elements present in the original designs.

B. Qualitative Analysis

To complement the quantitative results, we also conducted a qualitative analysis to examine the behavior of our proposed models under different masking conditions. For readability, the layouts are displayed using the same color-coded scheme introduced in Section IV, with component labels distinguished by the *UI color palette* (see Fig. 5).

Fig. 6 illustrates the predictions of *LayCoder* on one sample from the DS1 test set under three masking ratios: 10%, 25%, and 40%. For each ratio, the masked input and the completed layout are shown, while the corresponding ground-truth layout (labeled as "Real UI layout" in the figure) is displayed at the bottom. In many cases, the model successfully reconstructs missing elements and maintains a coherent spatial structure. The completions are not always identical to the ground truth, and in some cases the model generates alternative components or placements. While these differences can still yield plausible layouts from a design perspective, we emphasize that this observation is based on illustrative examples rather than systematic evaluation.

For further assessment, Fig. 7 presents results on three different UI layouts from DS1, each with a different masking ratio (10%, 25%, and 40%). Alongside reasonable completions, the figure highlights typical failure modes such as overlapping components, duplicated elements in the same region, or missing items. These examples demonstrate that although *LayCoder* can often predict plausible completions, it may also produce undesirable layouts under challenging conditions, especially at higher masking levels.

In Fig. 8, we evaluate both *LayCoder* and *LayCoder-NonOver-Leaves* on samples from DS2, where only leaf elements are retained and overlapping layouts are excluded. Using a 40% masking ratio, we compare the completed layouts of both models against the ground truth. The figure shows that *LayCoder-NonOver-Leaves* often avoids generating overlapping elements and produces cleaner structures, which is consistent with the hypothesis that reducing layout noise during training improves spatial consistency. However, limitations remain: in some cases, the outputs omit required elements or contain misaligned components, reflecting the cost

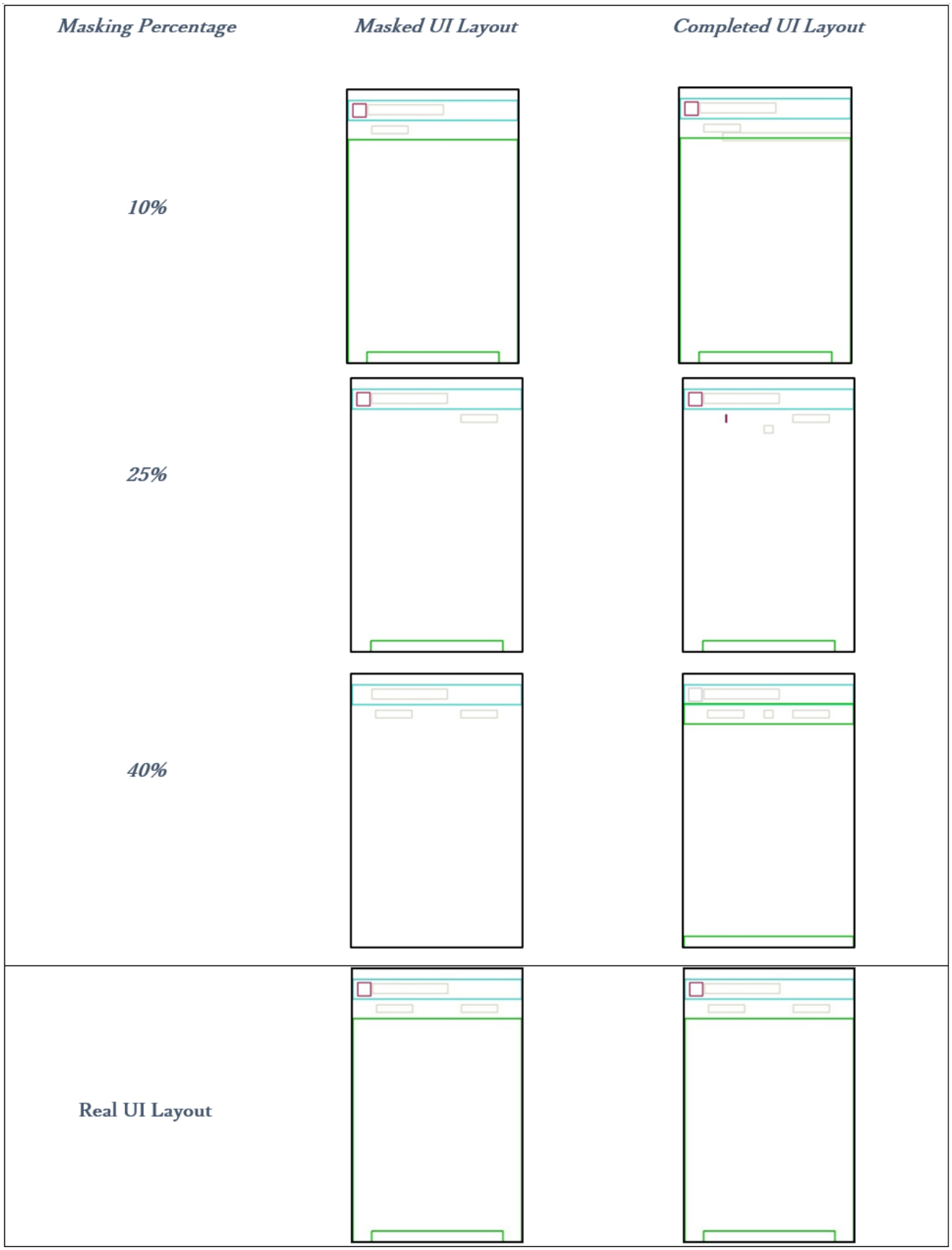


Fig. 6. Qualitative results of LayCoder on a DS1 sample under 10%, 25%, and 40% masking. Each row shows a masked input with the corresponding masking ratio and the resulting completion by LayCoder. The last row shows the ground-truth (real UI) layout.

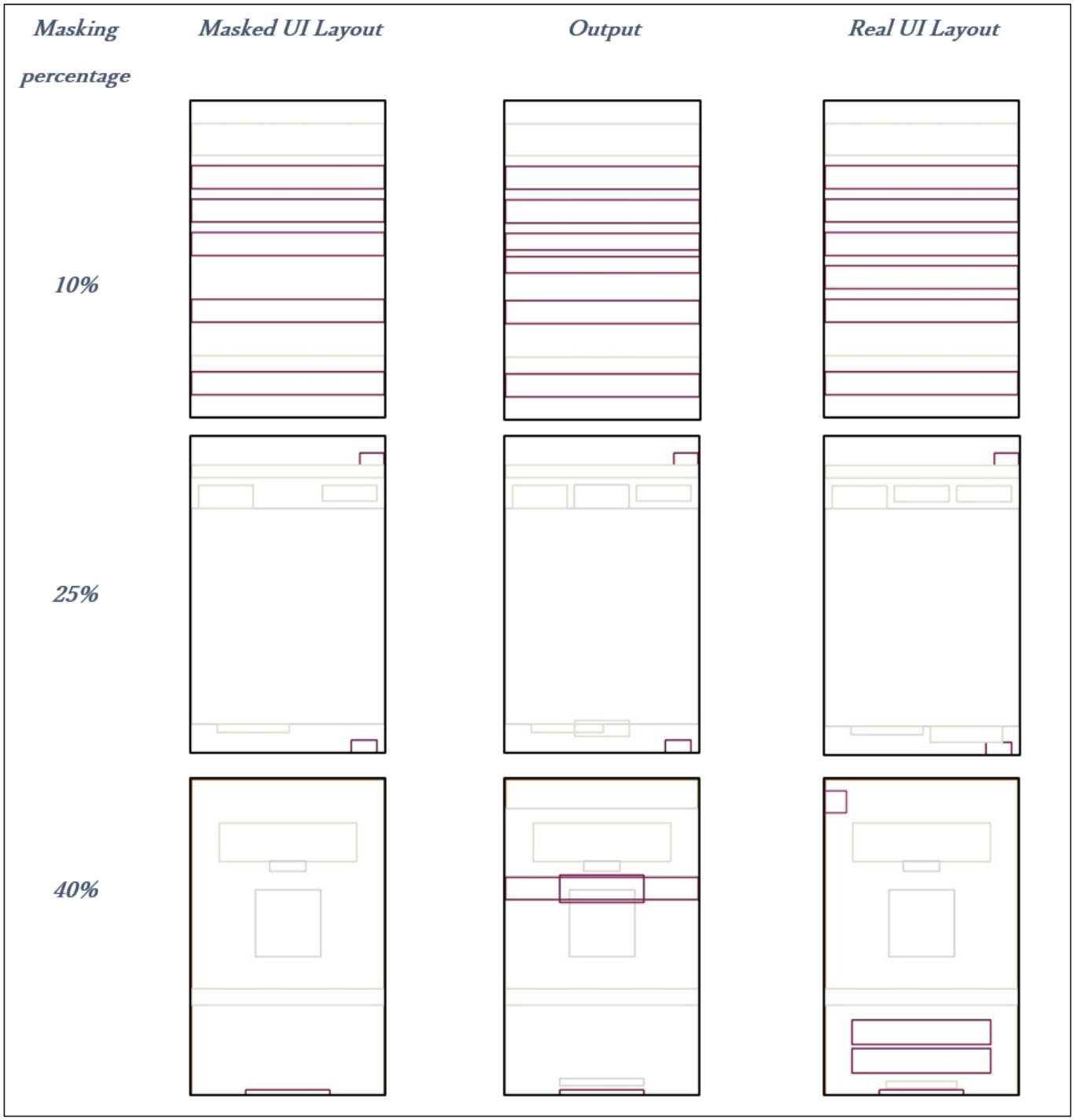


Fig. 7. Some failure cases from DS1 under different masking ratios. Examples highlight overlaps, duplicated elements, and missing components in *LayCoder* outputs.

of training on a more constrained dataset. Thus, while dataset curation enhances certain aspects of spatial accuracy, it does not guarantee uniformly superior results.

Overall, the qualitative evaluation highlights both the strengths and weaknesses of the proposed models. *LayCoder* demonstrates the ability to infer missing elements and produce plausible layouts across different masking levels, while *LayCoder-NonOver-Leaves* benefits from noise reduction by yielding cleaner structures. At the same time, both models exhibit failure cases that reveal important directions for further improvement.



Fig. 8. Comparison between *LayCoder* and *LayCoder-NonOver-Leaves* on DS2 with 40% masking. For each layout example, the masked input is shown alongside the completed outputs of both models and the corresponding ground-truth (real) UI layout.

VIII. Conclusion

In this study, we introduced LayCoder, an encoder-only Transformer formulated as a masked completion model for UI layout reconstruction. Our experiments show that LayCoder achieves competitive performance compared to LayoutFormer++, delivering improved Coverage and comparable or improved IoU and Alignment across the evaluated masking levels.

The results indicate that encoder-only transformer architectures combined with masked modeling are effective for layout completion under the evaluated settings. This formulation enables the model to leverage global context and learn structural dependencies without requiring explicit element-level annotations.

To further investigate the role of training data, we evaluated a noise-reduced subset of the dataset that excluded overlapping elements and non-leaf nodes. The results show that this curation can improve spatial consistency in terms of overlap and relative positioning, but at the cost of reduced Coverage. These findings suggest that dataset filtering shifts the balance of performance: strengthening certain structural aspects while limiting completeness, rather than uniformly enhancing predictions.

Despite these results, the current formulation relies on flat token representations and relatively shallow architectures, and does not explicitly model hierarchical relationships or complex structural constraints. Therefore, the findings should be interpreted as demonstrating the viability of masked encoder-only completion within the evaluated regime, rather than establishing scalability to deeper hierarchical modeling.

Failure cases reveal issues such as overlaps, duplications, or missing elements under challenging masking conditions, reflecting a structural trade-off between element completeness and precise spatial alignment. This trade-off suggests that incorporating additional structural constraints or relational representations may further improve the balance between layout completeness and spatial coherence in future work.

Acknowledgment

The authors would like to thank the creators of the *Rico dataset* and the authors of *LayoutFormer++* for making their data and code publicly available, which facilitated this research.

Data Availability

The datasets used in this study were derived from the publicly available *LayoutFormer++* dataset, which is based on the Rico dataset and available at https://github.com/microsoft/LayoutGeneration/tree/main/LayoutFormer++. We further processed this dataset to create two subsets, **DS1** and **DS2**, provided as `.pkl` files. All processed data and related preprocessing scripts are publicly available in our GitHub repository at https://github.com/salsik/LayCoder-Data under the MIT license.

Code Availability

The source code implementing the LayCoder model and training procedures is available at the project repository: https://github.com/salsik/cssl.

References

[1] Majumder, A. S. 2025. The Influence of UX Design on User Retention and Conversion Rates in Mobile Apps. ArXiv.

[2] Troiano, Luigi, and Birtolo, Cosimo. 2014. Genetic algorithms supporting generative design of user interfaces: Examples. Information Sciences, 259 (February 2014), 433-451. https://doi.org/10.1016/j.ins.2012.01.006

[3] Bleichner, Andreas, and Nils Hermansson. "Investigating the usefulness of a generative AI when designing user interfaces." (2023).

[4] Bandi, Ajay, Pydi Venkata Satya Ramesh Adapa, and Yudu Eswar Vinay Pratap Kumar Kuchi. "The power of generative AI: A review of requirements, models, input–output formats, evaluation metrics, and challenges." Future Internet 15.8 (2023): 260.

[5] Gupta, Kamal, et al. "LayoutTransformer: Layout generation and completion with self-attention." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

[6] Li, Yang, et al. "Auto completion of user interface layout design using transformer-based tree decoders." arXiv preprint arXiv:2001.05308 (2020).

[7] Vaswani, A., et al. (2017). "Attention is All You Need." In Advances in Neural Information Processing Systems (NeurIPS).

[8] Devlin, Jacob. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[9] Salama I, Mormille LH, Atsumi M. User Interface Design using Masked Language Modeling in a Transformer Encoder-based Model. 2024. doi:10.11517/pjsai.JSAI2024.0_3Q5IS2b05.

[10] Goodfellow, Ian, et al. "Generative adversarial networks." Communications of the ACM 63.11 (2020): 139-144.

[11] Kingma, D. P., & Welling, M. (2014). "Auto-Encoding Variational Bayes." arXiv:1312.6114.

[12] Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., & Yang, M.-H. (2022). "Diffusion Models: A Comprehensive Survey of Methods and Applications." arXiv:2209.00796.

[13] Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2022). "A Comprehensive Survey of Recent Transformers in Image, Video and Other Domains." ACM Computing Surveys (CSUR).

[14] Compare Encoder-Decoder, Encoder-Only, and Decoder-Only Architectures for Text Generation on Low-Resource Datasets. (2021). In *Advances on Broad-Band Wireless Computing, Communication and Applications* (BWCCA).

[15] Gajjar, Nishit, et al. "Akin: Generating ui wireframes from ui design patterns using deep learning." 26th International Conference on Intelligent User Interfaces-Companion. 2021.

[16] Li, Jianan, et al. "Layoutgan: Generating graphic layouts with wireframe discriminators." arXiv preprint arXiv:1901.06767 (2019).

[17] Jyothi, Akash Abdu, et al. "Layoutvae: Stochastic scene layout generation from a label set." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[18] Jing, Qianzhi, et al. "Layout generation for various scenarios in mobile shopping applications." Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. 2023.

[19] Arroyo, Diego Martin, Janis Postels, and Federico Tombari. "Variational transformer networks for layout generation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021.

[20] Inoue, Naoto, et al. "Layoutdm: Discrete diffusion model for controllable layout generation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023

[21] Zhang, Junyi, et al. "Layoutdiffusion: Improving graphic layout generation by discrete diffusion probabilistic models." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023.

[22] Chai, Shang, Liansheng Zhuang, and Fengying Yan. "Layoutdm: Transformer-based diffusion model for layout generation." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.

[23] Hui, Mude, et al. "Unifying layout generation with a decoupled diffusion model." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023.

[24] Jiang, Zhaoyun, et al. "LayoutFormer++: Conditional Graphic Layout Generation via Constraint Serialization and Decoding Space Restriction." arXiv preprint arXiv:2208.08037 (2022).

[25] Deka, Biplab, et al. "Rico: A mobile app dataset for building data-driven design applications." Proceedings of the 30th annual ACM symposium on user interface software and technology. 2017.

[26] Lee, Hsin-Ying, et al. "Neural design network: Graphic layout generation with constraints." Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. Springer International Publishing, 2020.

[27] Bai, Zihan, et al. "Layout representation learning with spatial and structural hierarchies." Proceedings of the AAAI Conference on Artificial Intelligence. 2023.

Reducing Interference in Human–Robot Collaboration: A Distance-Based Policy for Disassembly Collaborative Task

Wendy Cahya Kurniawan, Wen Liang Yeoh, Osamu Fukuda
Department of Computer Science and Intelligent Systems, Saga University, Japan

***Abstract*—This study investigates human-robot collaborative (HRC) disassembly tasks, focusing on the effects of constrained and unconstrained safety distances on human behavior and movement interference. Separate and shared target configurations were compared using a screw-picking task as a prototype for electronic waste (e-waste) disassembly. The system employ a vision-based approach to track screws, the human hand, and the robot end effector. Performance was evaluated using objective metrics, including completion time, warning distance, and movement heatmaps, as well as subjective workload assessed via NASA-TLX. Results show that the safety algorithm significantly improved movement efficiency by minimizing the spatial distribution compared to the unconstrained trials. Furthermore, significant results from the factorial design showed that separate-target task enhanced human awareness, enabling participants to anticipate warning distances more effectively than the shared-target task. Synchronizing task assignments with safety algorithms reduces the need for human intervention.**

***Keywords*—Human-robot collaboration; disassembly collaborative task; UI feedback; distance policy; vision-based sensor**

I. Introduction

Although the Industrial Revolution enabled significant technological advancements, it also introduced substantial challenges. Industry 4.0, integrating automation, big data, the Internet of Things (IoT), and artificial intelligence (AI), enhances productivity and efficiency [1], [2]. However, these advancements have caused a rapid increase in electronic waste (e-waste), posing a significant threat to environmental sustainability [3], [4]. Effective global waste management is essential for sustainability, with disassembly playing a critical role in material recovery [5]–[7]. However, disassembly is significantly more challenging to automate than assembly owing to the variability in product types. The disassembly process requires a high degree of flexibility to manage technical and economic uncertainties, which is crucial for enabling widespread industrial adoption.

The Industry 5.0 paradigm addresses these productivity and flexibility challenges [8]. Leveraging advancements in lightweight robots, human-robot collaboration (HRC) workspace can enhance efficiency and productivity, while prioritizing human safety in shared environments. Several studies have aimed to accelerate disassembly tasks, including disassembly sequence planning (DSP). Task assignment algorithms, such as the Discrete Bees Algorithm [9] and Genetic Algorithm [10], have been developed to optimize human–robot labor division under specific safety constraints [11], [12].

From a human-centered system perspective, experimental studies examining real-time human movement during collaborative disassembly task allocation are lacking. Additionally, the impact of task type—whether spatially separated or shared—on human safety, efficiency, and workload has not yet been systematically investigated. Although objective safety indicators are frequently reported, subjective aspects, such as workload and perceived stress—critical for human-centered system design—are often underexplored.

This study investigated the reduction of human interference in HRC disassembly tasks. We developed a system that uses object detection to track the locations of humans, robots, and tasks. By evaluating different task types and algorithms, we provide empirical evidence of safety and efficiency trade-offs. The primary contribution of this study is an analysis of how task-algorithm pairings influence HRC workspace design to minimize human-robot conflict.

This study is organized as follows: Section II details the proposed HRC disassembly methodology and the technical architecture of the system design. Section III presents the experimental findings, followed by a discussion of the findings and limitations. Finally, Section IV concludes the study by summarizing the main contributions and future research directions.

II. Human-Robot Collaborative Disassembly

This section outlines the research methodology, including workspace setup and requirements, an overview of the system, applied algorithms, experimental design, and the evaluation using NASA-TLX. Fig. 1 presents an overview of the system, which tracks the real-time positions of the human, robot, and task locations. The architecture, based on an object detection algorithm, comprises two primary modules: a data processing unit for robot motion stimuli and a feedback determination component.

A. Workspace Setup

The experimental environment employed a DOBOT MG400, a lightweight industrial robot with a 30 cm $\times$ 20 cm workspace, selected for its high precision in intricate tasks. The AI-integrated camera served as a single sensor for real-time environmental mapping and object detection. In this collaborative disassembly scenario, human and robot worked in a shared workspace to perform sequential screw disassembly.

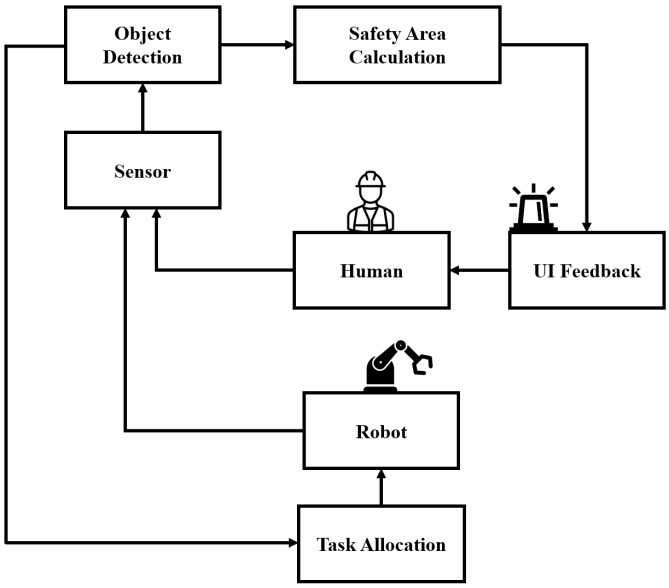


Fig. 1. Overview of proposed research.

To prioritize human safety, a task allocation algorithm dynamically adjusts the robot's target screw locations based on human movements. The experimental setup is shown in Fig. 2.

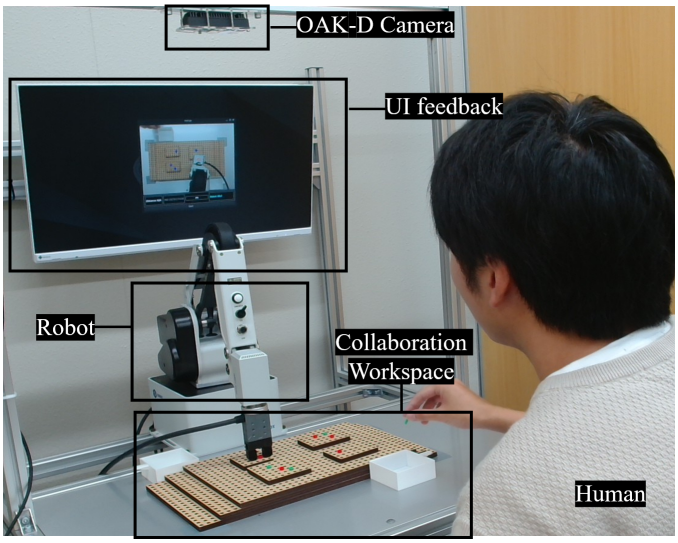


Fig. 2. Collaborative workspace setup.

The collaborative workspace was designed as a prototype for e-waste disassembly, using an MDF board for screw placement. Monitors integrated into the setup to provide real-time UI feedback, thereby facilitating communication between humans and robots. This UI displayed a safety zone, that was dynamically calculated via the human-robot object detection system to guide the user during the task.

B. Human-Robot Collaborative Algorithm

The proposed system uses object detection based on the YOLOv8 framework, selected for its proven real-time performance among state-of-the-art detection models. Detection accuracy and precision were evaluated through data collection, model training, and performance assessment.

The information obtained from the detected objects was in the form of x- and y-coordinates, with three labels. Assume that the coordinates of the humans, robots, and several screws are denoted as $H(x_h, y_h)$, $R(x_r, y_r)$, and $S_i(x_{si}, y_{si})$, respectively. This research focuses on the collaboration between a human hand and a robotic arm tasked with simultaneously picking up screws. In this study, distance calculations were required to measure the safety distance for collaboration. Eq. (1) was employed to obtain the real line distance from the x- and y-coordinates. This form is the absolute value between the human hand H and the screw S_i. where d_{HSi} is the distance between the hand and screw S_i. This distance was used for safety algorithm calculations. The calculation results serve as a stimulus for the robot to move toward the screw farthest from the human hand. Eq. (2) describes the calculation used to obtain the farthest distance. The detailed algorithm is described using the pseudocode Algorithm 1.

$$d_{HSi} = \|\mathbf{H} - \mathbf{S}_i\| \tag{1}$$

$$S_{far} = \max\{d_{HS1}(H, S_1), d_{HS2}(H, S_2), \ldots, d_{HSm}(H, S_m)\} \tag{2}$$

Algorithm 1 Safety Task Detection Algorithm

```
img ← cap_img()
H ← det_hand(img)
S ← det_screws(img)

for each S_i in S do
    dHS_i ← dist(H, S_i)
    if dHS_i > d_max then
        d_max ← dHS_i
        S_far ← S_i
    end if
end for
```

The human-robot distance, d_{HR}, is calculated using Eq. (3) and serves as a trigger for visual feedback.

$$d_{HR} = \|\mathbf{H} - \mathbf{R}\| \tag{3}$$

By observing the midpoint of both objects, safety considerations were divided into four areas calculated based on the distance between d_{min} 100 and d_{max} 400 pixels. These four areas indicate the level of danger, ranging from danger to caution, warning, and safety. Through distance normalization, as in Eq. (4):

$$\hat{d}_{HR} = min(\frac{d_{HR}}{d_{max}}, 1) \tag{4}$$

Denotes the normalized distance. This normalization maps the distance to a range of 0–1. If the human detection distance exceeded the d_{max} threshold, the person was automatically categorized as being in the safe zone. To visualize this on the

UI, the system calculates the inverse of the normalized distance to represent the danger level, as shown in the following Eq. (5):

$$D = (1 - \hat{d}_{HR}). \tag{5}$$

Eq. (6) defines four zones: danger, caution, warning, and safe. This zonal classification is fundamental to the HRC workspace, as it prioritizes collaborative safety.

$$f(D) = \begin{cases} \text{Danger}, & D > 0.5 \\ \text{Caution}, & 0.4 < D \leq 0.5 \\ \text{Warning}, & 0.3 < D \leq 0.4 \\ \text{Safe}, & D \leq 0.3 \end{cases} \tag{6}$$

This study investigated human responses to safety-based and random algorithms to assess their relative effectiveness. A combined approach integrating algorithm selection and UI feedback was implemented. Fig. 3 shows the UI feedback information used in this study, which includes the camera view serving as the sensing input, human-robot distance indicators for collision risk, and information on the screw selected for the robot pickup.

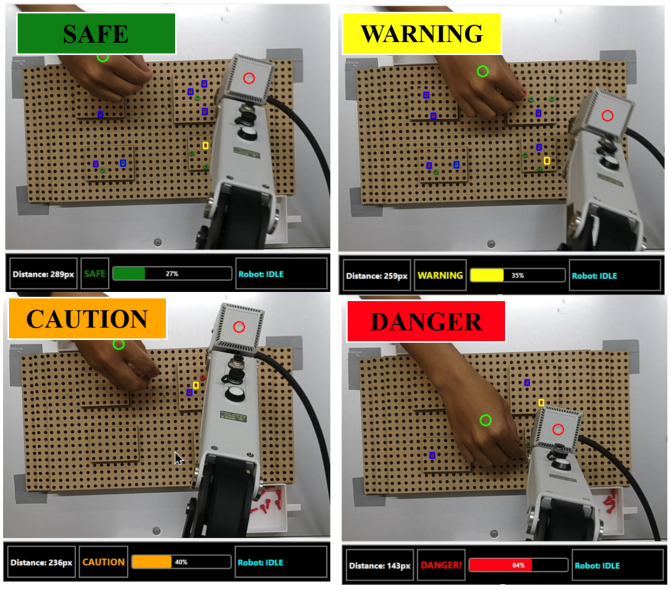


Fig. 3. Safety area UI feedback.

C. Experiment Design

The experiment involved five participants (aged 32 ± 8.3 years), each completing 10 trials per algorithm. Human responses to robot movements were evaluated using three objective parameters: task completion time, hand movement trajectories, and human-robot warning distance. Following the experimental trials, the participants completed a NASA-TLX questionnaire to assess perceived workload and system performance.

The algorithm was tested using two different task scenarios to obtain human behavior data. The separate target task (Task 1) involved 20 randomly positioned screws (10 red and 10 green screws). The robot retrieved the red screws until they were collected. The shared target task (Task 2) utilized 30 randomly positioned screws of the same color. The human-robot retrieved the screws until no screws remained. Fig. 4 illustrates the setup of Tasks 1 and 2.

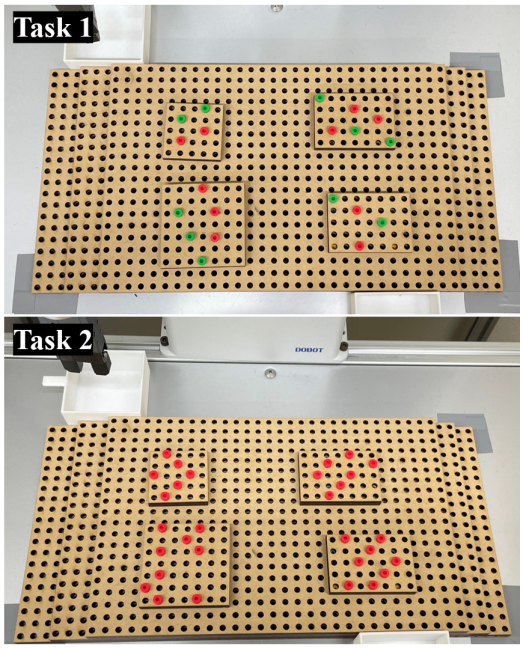


Fig. 4. Experiment task setup.

D. NASA-TLX Evaluation System

Subjective evaluation was conducted using the NASA-TLX questionnaire [13], which assess workload across six domains using a Likert scale: mental demand (MD), physical demand (PD), temporal demand (TD), performance (P), effort (E), and frustration level (FL). Participants completed the questionnaire after each experiment across the four distinct scenarios (comprising two tasks and two algorithms). The resulting data were analyzed using descriptive statistics to determine the impact of each scenario and identify significant differences in workload.

The use of NASA-TLX as a subjective evaluation identifies 'hidden costs' through direct participant feedback. Implementing such a human-centric system ensures that the operator's well-being is prioritized along with the robot's technical efficiency, providing a holistic justification for the system's deployment in industrial settings.

III. Experimental Results and Discussion

This section presents the experimental results and examines human behavior during collaborative disassembly tasks. The analysis focused on three key metrics: the completion time, frequency of proximity-based safety warnings (warning rate), and hand movements. The results of the NASA-TLX also captured the participants' perceptions.

A. Conditions

The participants were informed of the research objectives and experiments prior to providing informed consent, in accordance with the ethical requirements of the Saga University Ethics Committee (R4-38). During the experiment, the human and robot operated in a shared, barrier-free workspace, where collisions were possible. Fig. 5 shows the experimental setup in which the human and robot work together to retrieve screws until the workspace was cleared.

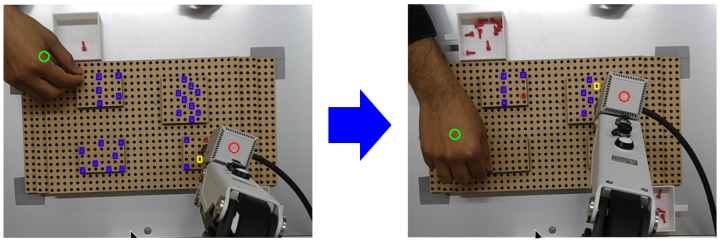


Fig. 5. Illustration of the retrieval process performed by the human-robot until task completion.

The experimental system was designed to facilitate data collection during human-robot collaboration. Following improvements in the object detection accuracy and real-time validation, the system was evaluated using both algorithms. Fig. 6 and Fig. 7 show the warning distance signals for the unconstrained and safe algorithms, respectively. The data show that the safety algorithm consistently maintains the required safety distance, confirming that the robot operates in accordance with the predefined safety thresholds.

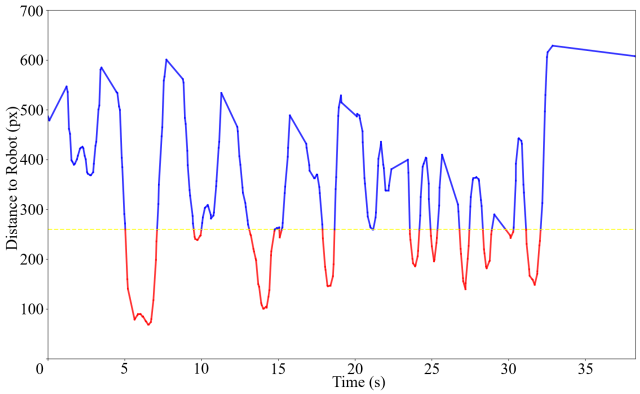


Fig. 6. Sample warning distance signals generated by unconstrained algorithm.

B. Results

The effects of algorithms and task types were examined using a factorial design, with statistical significance assessed via two-way ANOVA [14] at $\alpha < 0.05$. This analysis focused on the influence of these factors on completion time and total warning distance. Before the analysis, outliers were screened using the interquartile range (IQR) to ensure data integrity. Corresponding visualizations are presented in the bar charts in Fig. 8 and Fig. 9.

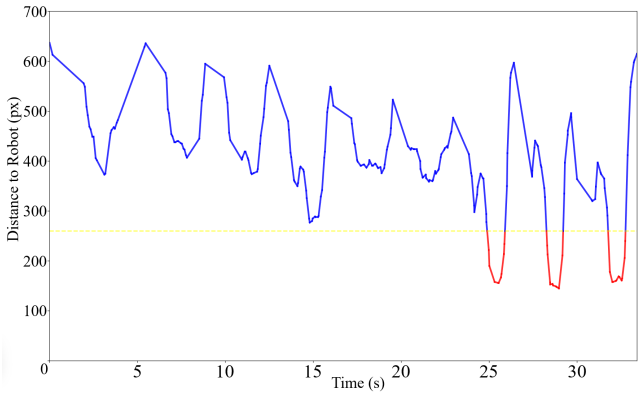


Fig. 7. Sample warning distance signals generated by safety algorithm.

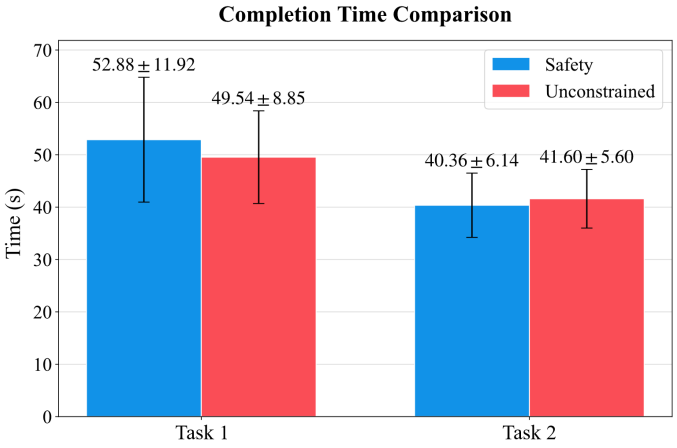


Fig. 8. Comparison of completion time performance for the safety and unconstrained algorithms during Task 1 and Task 2.

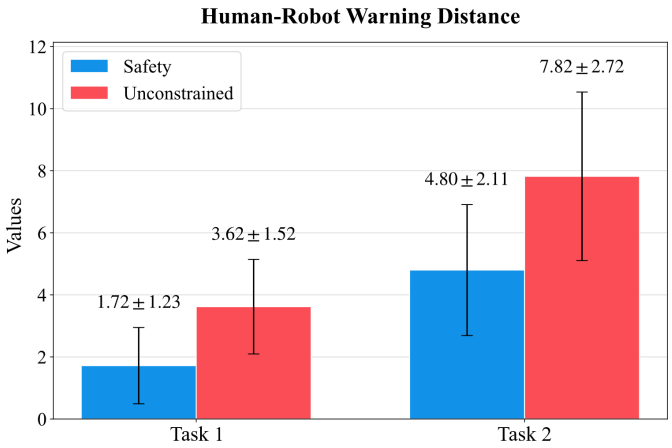


Fig. 9. Comparison of human-robot warning distance for the safety and unconstrained algorithms during Task 1 and Task 2.

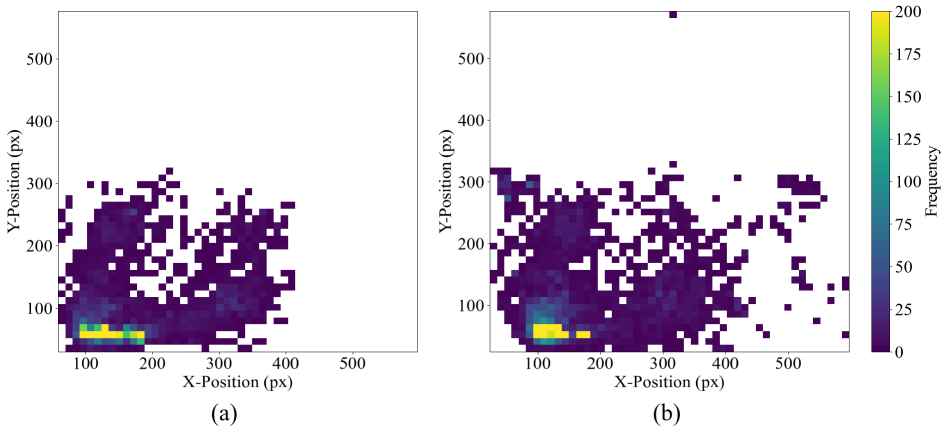


Fig. 10. Heatmaps of human hand movement during Task 1 for the (a) safety algorithm and (b) unconstrained algorithm.

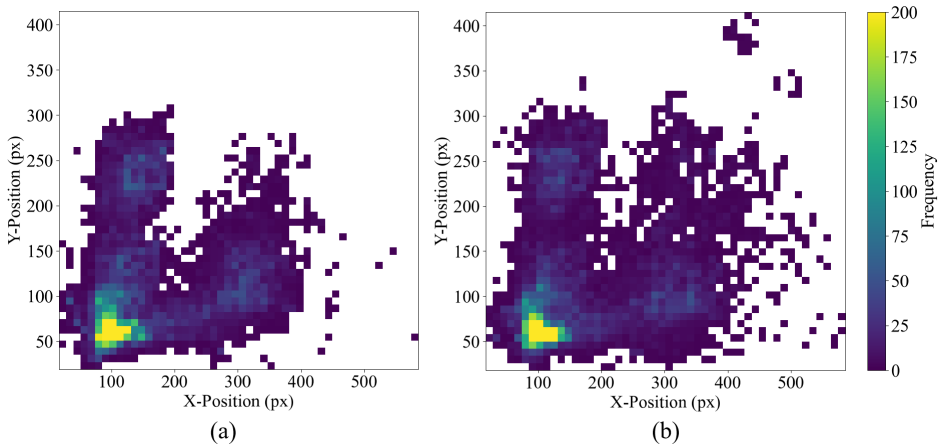


Fig. 11. Heatmaps of human hand movement during Task 2 for the (a) safety algorithm and (b) unconstrained algorithm.

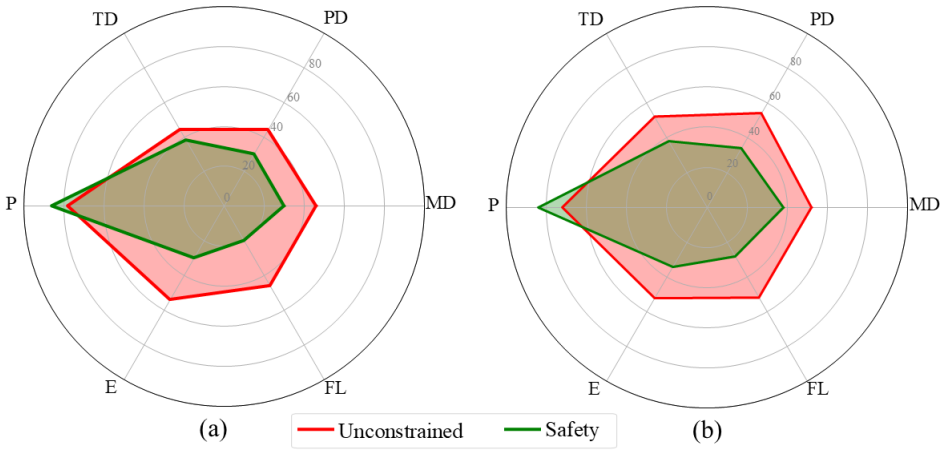


Fig. 12. NASA-TLX workload scores for (a) Task 1 and (b) Task 2 across both algorithms.

A two-way ANOVA conducted on completion time revealed a significant main effect for the task used ($F = 72.307$, $p < .001$, $\eta^2 = .269$). However, no significant main effect was found for the algorithm ($F = 0.76$, $p = .38$), and the interaction between algorithm and task did not reach statistical significance ($F = 3.62$, $p = .058$, $\eta^2 = .018$). These findings suggest that the differences in completion times were typically driven by the task itself, regardless of the algorithm being performed.

The two-way ANOVA for human-robot warning distance revealed significant main effects for both the algorithm ($F = 77.22$, $p < .001$, $\eta^2 = .283$) and the task ($F = 169.07$, $p < .001$, $\eta^2 = .463$). The interaction between algorithm and task was statistically significant ($F = 4.002$, $p = .047$, $\eta^2 = .02$). These results indicate that, while both the choice of algorithm and the specific task performed independently influenced the warning distance, the effect of the algorithm remained consistent across both tasks.

The spatial distribution of human movements was analyzed using heatmaps, as illustrated in Fig. 10 and Fig. 11 for Tasks 1 and 2, respectively. Comparative visual analysis revealed that the safety algorithm produced a significantly narrower distribution of human movement positions than the unconstrained algorithm. This concentration indicates that the safety algorithm facilitated a more predictable and structured human-robot interaction. These findings suggest that humans respond more effectively and consistently to robot movements when a safety algorithm is implemented.

Subjective assessments were performed using the NASA-TLX dimensions. A paired t-test confirmed a significant overall difference in Task 1 ($t = 4.402$, $p = .012$). In contrast to the previous measurements, the results for Task 2 were not statistically significant although they demonstrated a marginal trend ($t = 2.383$, $p = .076$). Fig. 12 shows the NASA-TLX differences between the two tasks.

C. Discussion

Experimental results indicate that the proposed algorithm significantly modulates human behavior during robot collaboration. Although real-time human movement remains unpredictable, spatial mapping using heatmaps reveals distinct behavioral patterns. Specifically, human responses were more efficient and localized when collaborating with a safety-constrained robot, indicating predictable robot movements and an enhanced sense of human safety.

Despite identical workspace dimensions, task type significantly influenced movement distribution. In Task 1 (a Separate Task), the range of human movement was much narrower than that in Task 2 (a Shared Task). This difference in spatial distribution is corroborated by the human-robot warning distance data, suggesting a lower collision probability for Task 1. This finding suggests that the separate task facilitates a safer environment by allowing humans to anticipate robot trajectories more effectively, whereas the shared task requires a wider range of avoidance maneuvers, and thus, is more complex. The integration of task assignment with safety algorithms yields results that minimize human interference.

D. Limitations

A collaborative disassembly system was developed to address the primary research questions. However, further investigation is needed to assess the system's long-term sustainability and robustness. Considering the limited sample size, the current findings may have restricted generalizability. Broader validation with a larger and more diverse population is required to support public and industrial adoption.

In addition, current hardware configurations would benefit from sensor redundancy. Although the vision-based approach effectively preserves human flexibility during collaboration, the camera's limited field of view necessitates additional sensors to mitigate occlusion issues.

IV. Conclusion and Future Work

This study investigated human interference during collaborative disassembly. The proposed safety algorithm significantly reduced the frequency of hazard warnings. Furthermore, NASA-TLX evaluations indicated that distinct task allocation, combined with feedback information, facilitated better maintenance of human-robot separation distances compared to shared-task execution. These results suggest that humans anticipate robot trajectories when task objectives are clearly defined in advance.

Note that the current disassembly process is a prototype involving simplified tasks. Future work will focus on optimizing the safety algorithm to prevent collisions during the final few tasks, thereby enhancing both efficiency and ergonomic outcomes. Additional studies with larger and more diverse participant groups are needed to improve the scalability of industrial applications.

Acknowledgment

This study was funded by JSPS KAKENHI (grant number 23H03440).

References

[1] M. Dhanda, B. A. Rogers, S. Hall, E. Dekoninck, and V. Dhokia, "Reviewing human-robot collaboration in manufacturing: Opportunities and challenges in the context of industry 5.0," *Robotics and Computer-Integrated Manufacturing*, vol. 93, p. 102937, Jun. 2025. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584524002242

[2] S. Proia, R. Carli, G. Cavone, and M. Dotoli, "Control Techniques for Safe, Ergonomic, and Efficient Human-Robot Collaboration in the Digital Industry: A Survey," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 3, pp. 1798–1819, Jul. 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9638383/

[3] K. Liu, Q. Tan, J. Yu, and M. Wang, "A global perspective on e-waste recycling," *Circular Economy*, vol. 2, no. 1, p. 100028, 2023.

[4] R. Ahirwar and A. K. Tripathi, "E-waste management: A review of recycling process, environmental and occupational health hazards, and potential solutions," *Environmental Nanotechnology, Monitoring & Management*, vol. 15, p. 100409, 2021.

[5] A. Morachioli, V. Sivtsov, N. Rojas, and F. Bonsignorio, "A Functional and Practical Taxonomy for the Industrial Implementation of Highly Automated Reverse Manufacturing Cells," *IEEE Open Journal of the Industrial Electronics Society*, vol. 5, pp. 1115–1139, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10666886/

[6] M. Caterino, M. Rinaldi, V. Di Pasquale, M. Fera, and D. T. Pham, "Addressing the impact of robots during disassembly operations: results from an empirical study," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, Aug. 2025. [Online]. Available: https://link.springer.com/10.1007/s12008-025-02383-7

[7] J. Huang, D. T. Pham, R. Li, M. Qu, Y. Wang, M. Kerin, S. Su, C. Ji, O. Mahomed, R. Khalil, D. Stockton, W. Xu, Q. Liu, and Z. Zhou, "An experimental human-robot collaborative disassembly cell," *Computers & Industrial Engineering*, vol. 155, p. 107189, May 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0360835221000930

[8] F. Barravecchia, M. Bartolomei, L. Mastrogiacomo, and F. Franceschini, "Designing Symbiotic Human–Robot Collaboration in assembly tasks," *Production Engineering*, vol. 19, no. 3-4, pp. 629–661, Jun. 2025. [Online]. Available: https://link.springer.com/10.1007/s11740-025-01333-2

[9] W. Xu, Q. Tang, J. Liu, Z. Liu, Z. Zhou, and D. T. Pham, "Disassembly sequence planning using discrete Bees algorithm for human-robot collaboration in remanufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 62, p. 101860, Apr. 2020. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584519301887

[10] S. Parsa and M. Saadat, "Human-robot collaboration disassembly planning for end-of-life product disassembly process," *Robotics and Computer-Integrated Manufacturing*, vol. 71, p. 102170, Oct. 2021. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584521000545

[11] M.-L. Lee, S. Behdad, X. Liang, and M. Zheng, "Task allocation and planning for product disassembly with human–robot collaboration," *Robotics and Computer-Integrated Manufacturing*, vol. 76, p. 102306, Aug. 2022. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0736584521001861

[12] M.-L. Lee, W. Liu, S. Behdad, X. Liang, and M. Zheng, "Robot-Assisted Disassembly Sequence Planning With Real-Time Human Motion Prediction," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 1, pp. 438–450, Jan. 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9817380/

[13] A. K. Eesee, V. Varga, G. Eigner, and T. Ruppert, "Impact of work instruction difficulty on cognitive load and operational efficiency," *Scientific Reports*, vol. 15, no. 1, p. 11028, Apr. 2025. [Online]. Available: https://www.nature.com/articles/s41598-025-95942-7

[14] C. Andrade, "Understanding Factorial Designs, Main Effects, and Interaction Effects: Simply Explained with a Worked Example," *Indian Journal of Psychological Medicine*, vol. 46, no. 2, pp. 175–177, Mar. 2024. [Online]. Available: https://journals.sagepub.com/doi/10.1177/02537176241237066

Understanding Authentication and Authorization: A Comparative Analysis of Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC), and Relationship-Based Access Control (ReBAC) Authorization Models

Madhuri Margam
Director, Software Engineering, Capital One, Virginia, USA

***Abstract*—This study elucidates the distinctions between authentication and authorization within information security, two fundamental yet frequently conflated concepts. While authentication serves to confirm an entity's identity, authorization determines the permissible actions that entity may execute. A thorough understanding of these mechanisms is critical for architecting secure, scalable systems and reducing vulnerabilities. The study further explores three widely adopted authorization paradigms using a gymnasium analogy: Role-Based Access Control (RBAC), which assigns privileges based on predefined roles; Attribute-Based Access Control (ABAC), which leverages a dynamic evaluation of user and contextual attributes; and Relationship-Based Access Control (ReBAC), which determines access based on defined relationships among entities. The concluding discussion emphasizes that optimal security is realized when authentication and authorization function cohesively.**

***Keywords*—Authentication; authorization; access control; Role-Based Access Control (RBAC); Attribute-Based Access Control (ABAC); Relationship-Based Access Control (ReBAC); information security; least privilege; Zero Trust**

I. Introduction

In modern API security and system design, authentication and authorization are foundational, yet distinct processes often misconstrued as interchangeable. Authentication verifies who an entity is, while authorization governs what that entity can do once authenticated. Misinterpretation and improper implementation of these principles can lead to critical security flaws, enabling privilege escalation or unauthorized access. Organizations implement strong authentication but assume it provides authorization, or have minimal investments in authorization. Common implementation mistakes within authorizations include poor role design, over-privileged access, inadequate access reviews, and policy management.

To illustrate, consider the analogy of a gymnasium. Authentication occurs when a member scans their access card to validate identity and membership status. Authorization follows, defining which facilities, such as the pool or private training areas, the member may access based on their membership type. Both mechanisms are indispensable for a secure and user-oriented access management framework.

II. Authentication and Authorization: Conceptual Overview

A. Authentication

Authentication is the process of verifying the identity of a user or a service attempting to access a system. It answers the question, "Who are you?" Common authentication methods include password-based login, multi-factor authentication (MFA), and cryptographic verification.

In the gymnasium analogy, scanning a valid membership card authenticates the member, verifying their legitimacy. Only authenticated members proceed to the authorization stage.

B. Authorization

Authorization defines the access level and actions an authenticated user may perform. It answers, "What are you allowed to do?" For instance, in the gym, premium members may access group classes, while basic members are restricted to general equipment. The Principle of Least Privilege is supported by this distinction. The Principle of Least Privilege (PoLP) is a fundamental information security concept that requires users, processes, and systems to be granted only the minimum level of access necessary to perform their authorized functions. The goal is to prevent unauthorized access to data and services, coupled with making the access control enforcement as granular as possible [1].

Authentication and authorization together create a secure, layered defense mechanism that validates identity before granting specific privileges, as shown in Fig. 1.

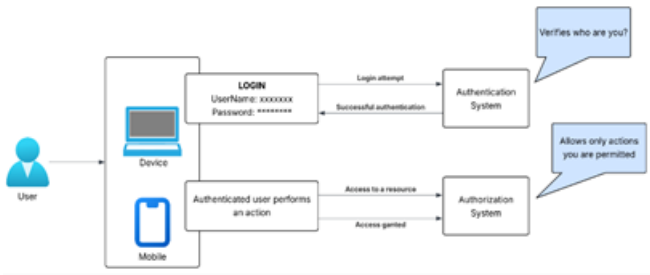


Fig. 1. Authentication and authorization mechanisms.

III. Common Authorization Models

Different domains implement authorization based on operational complexity and compliance needs. The three most prevalent models are RBAC, ABAC, and ReBAC.

A. Role-Based Access Control (RBAC)

RBAC associates permissions with predefined roles, simplifying access management for organizations with structured hierarchies. Access is granted based on the user's assigned role rather than individual identity.

In a gymnasium example, membership levels define roles:

- Basic members: access to cardio and weight areas
- Premium members: additional access to group classes
- VIP members: full access, including pool and lounge facilities

Advantages of RBAC:

- Enhanced security through role-based restriction
- Simplified policy management
- Regulatory compliance support
- Scalable and transparent administration

RBAC's simplicity and predictability make it ideal for environments with stable, well-defined roles.

B. Attribute-Based Access Control (ABAC)

ABAC extends RBAC by incorporating dynamic attributes, such as user characteristics, environmental context, and resource metadata, into policy decisions. Instead of static role assignments, ABAC evaluates combinations of attributes to determine access in real-time.

In a gymnasium example:

- Discount eligibility may depend on attributes like age < 25 or membership duration > 1 year.
- Facility access may be restricted to staffed hours, introducing contextual rules.

Advantages of ABAC:

- Fine-grained and adaptable access control.
- Context-aware and scalable.
- Dynamic response to environmental or user changes.

ABAC is a cornerstone for modern, policy-driven security architectures that demand adaptability and compliance. ABAC serves as the foundation for contemporary, policy-driven security architectures requiring both compliance and adaptability [2].

C. Relationship-Based Access Control (ReBAC)

ReBAC builds upon ABAC by evaluating relationships between entities rather than static roles or attributes. It is particularly effective in systems where access rights depend on social, organizational, or contextual links.

In a gymnasium example:

- Trainer-client relationship: Trainers access workout data only for assigned clients.
- Parental access: Parents manage reservations for children on family plans.
- Guest access: Guests enter only when sponsored and accompanied by a primary member.

Advantages of ReBAC:

- Granular and intuitive policy expression.
- Dynamic adaptation as relationships evolve.
- Reduced risk of privilege overextension.
- Scalable management in graph-like systems [3].

Systems such as Google Zanzibar exemplify ReBAC in practice, offering a globally consistent, low-latency authorization framework based on relational mappings between users and resources [4].

IV. Evaluation of Authorization Models

In the evolving landscape of digital security, access control mechanisms must balance security rigor, operational scalability, and user experience. While RBAC, ABAC, and ReBAC are often presented as distinct models, their practical implementation in large-scale systems tends to be complementary rather than mutually exclusive. Each model exhibits unique strengths, limitations, and contextual suitability.

A. Limitations in the Current Models

The Role-Based Access Control (RBAC) faces critical scalability and flexibility challenges in modern enterprise environments. Organizations frequently experience role explosion, creating hundreds of granular roles that become administratively complex and error-prone to manage. RBAC's static nature cannot adapt to dynamic business requirements such as time-sensitive access, location-based restrictions, or contextual factors without extensive manual reconfiguration. Complex role hierarchies often result in unintended privilege accumulation through inheritance, violating the principle of least privilege, while the model struggles to accommodate temporary access requests or exceptional permissions outside predefined role boundaries.

Attribute-Based Access Control (ABAC) and Relationship-Based Access Control (ReBAC) address some RBAC limitations but introduce distinct operational challenges. ABAC's policy authoring complexity requires specialized expertise, and real-time policy evaluation can introduce performance overhead in high-transaction environments [5]. Managing attribute accuracy across distributed systems remains operationally challenging, as stale attributes lead to inappropriate access decisions [6]. ReBAC faces scalability concerns with large, deeply nested relationship graphs, where traversing complex chains becomes computationally expensive [4]. Maintaining accurate relationships as organizational structures requires continuous synchronization, and transitive relationships can create unintended access paths. Additionally,

ReBAC's relative immaturity compared to RBAC means fewer standardized tools and best practices exist for implementation [4].

B. Complementary Rather than Replacement

RBAC remains one of the most widely adopted authorization mechanisms due to its simplicity, transparency, and alignment with organizational hierarchies. It excels in environments with stable structures, such as enterprises where roles (e.g., admin, developer) and privileges are well-defined and rarely change. However, RBAC's rigidity becomes a limitation in dynamic systems where access requirements evolve rapidly or depend on context-specific data.

ABAC addresses this limitation by introducing policy-based flexibility. It allows access decisions to be based on attributes that change over time, such as time of day, age [3]. This dynamic nature supports continuous access evaluation. Yet, ABAC's granularity also introduces complexity in policy definition and enforcement, which can be challenging to manage at scale.

ReBAC, as demonstrated by Google's Zanzibar [4], introduces a graph-based perspective focusing on the relationships among entities such as users, groups, and resources. This model captures nuanced access patterns prevalent in social, collaborative, and multi-tenant systems. For instance, "user A can edit document B if they are a collaborator or share a group with the document owner". ReBAC's strength lies in its ability to model these relational dynamics efficiently, often with real-time consistency guarantees.

However, it is important to note that ReBAC does not inherently subsume ABAC. While relationships define who can access what, attributes often define under what conditions access should be granted. For example, a trainer-client relationship (ReBAC) may grant access, but ABAC attributes like training session time or active membership status further refine when access is valid. Thus, a hybrid ABAC–ReBAC integration is necessary for robust and context-aware authorization [7].

C. Towards Hybrid Access Control Systems

Emerging industry and research trends increasingly favor hybrid access control architectures that integrate role-based, attribute-based, and relationship-based principles to address the complexity of authorization in dynamic and distributed environments [8],[9]. Hybrid models that combine RBAC and ABAC overcome limitations of individual paradigms by retaining RBAC's structural simplicity while introducing ABAC's contextual flexibility for fine-grained, dynamic policy evaluation [8], [10]. Additionally, hybrid frameworks incorporating relationship-based semantics (ReBAC) capture evolving trust relationships among entities, supporting access decisions in collaborative and federated systems [11]. Together, these approaches align with Zero Trust principles, where policies must adapt to both static organizational roles and dynamic operational context, as shown in Fig. 2.

In practical deployments, policy-based access control frameworks centralize policy definition while enabling distributed enforcement near the workload, such as in API gateways, service mesh proxies, and edge runtimes [10]. RBAC provides the foundational governance structure for role assignments and baseline privileges, simplifying administration and compliance oversight [8]. ABAC enhances this foundation with dynamic, context-aware decisions based on user, resource, and environmental attributes, supporting adaptive and fine-grained access in cloud-native and IoT systems [9]. ReBAC extends policy expressiveness by modeling relationships among entities, such as collaborators, delegated roles, or federated trust links, which traditional models cannot easily represent [11]. Composing these models in a unified framework enables defense-in-depth authorization, ensuring no single model becomes a point of rigidity or failure while maintaining scalable, context-aware access control across modern distributed systems [8], [10].

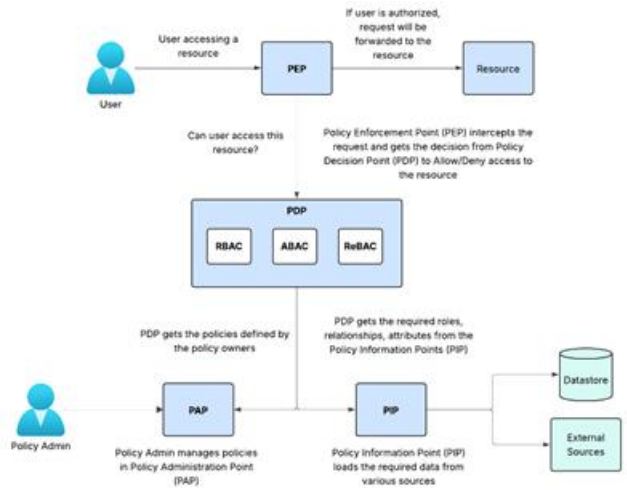


Fig. 2. Hybrid policy-based authorization framework integrating RBAC for role evaluation, ABAC for contextual policy evaluation, and ReBAC for relationship-aware access control.

D. Scalability, Performance, and Policy Governance

Scalability is another critical consideration. While RBAC scales efficiently through hierarchical role inheritance, ABAC's real-time policy evaluation and ReBAC's graph traversal can introduce computational overhead. Zanzibar suggests mitigating this through tuple-based data modeling and consistency guarantees, achieving millisecond-level authorization checks across millions of users [4]. Similarly, policy decision points (PDPs) and policy enforcement points (PEPs) in ABAC systems are often optimized through caching and pre-computation to maintain performance.

Policy governance also plays a central role in large-scale deployments. As the number of attributes and relationships increases, ensuring policy correctness, avoiding conflicts, and maintaining auditability become significant challenges. Recent research suggests using policy engineering frameworks and formal verification techniques to validate complex access rules before deployment.

E. Security and Compliance Implications

Regulatory compliance further influences model selection. RBAC aligns well with auditability and separation-of-duty requirements. ABAC enhances compliance by enabling contextual and attribute-based access controls, while ReBAC provides lineage and traceability by modeling user to resource relationships explicitly.

In highly secure systems, combining these models ensures not only that the access is correctly controlled but also that compliance requirements are demonstrably met. This multifaceted alignment of technical and regulatory priorities exemplifies the maturing discipline of robust authorization design. Table I shows the comparison of all three models.

TABLE I. COMPARISON OF AUTHORIZATION MODELS ACROSS KEY DIMENSIONS

Aspects	RBAC	ABAC	ReBAC
Complexity	Low	High	Medium
Flexibility	Low	High	High
Performance	High	Medium	High
Scalability	Medium	High	High
Context Awareness	Low	High	Medium
Relationship Modeling	Low	Medium	High
Policy Management Complexity	Low	High	Medium

V. AUTHORIZATION MODELS IN MODERN DISTRIBUTED ARCHITECTURES

Modern cloud-native architectures characterized by microservices, service meshes, serverless execution, event-driven communication, and hybrid multi-cloud deployment models dissolve traditional network perimeters and undermine implicit trust assumptions embedded in perimeter-based security models [12]. As services and workloads are distributed across heterogeneous environments and dynamically instantiated, static access control and coarse-grained network controls become insufficient to prevent lateral movement and unauthorized access [12]. Consequently, authorization must be enforced continuously and at fine granularity across users, services, and workload identities, incorporating contextual signals such as identity attributes, runtime state, and least-privilege policies [12], [13]. These requirements motivate the adoption of Zero Trust-aligned authorization frameworks that treat every access request as untrusted by default and require explicit, policy-driven authorization decisions across all access boundaries [12].

To operationalize these principles at scale, modern distributed systems increasingly adopt hybrid authorization models that decouple centralized policy governance from distributed, low-latency enforcement closer to the workload [12]. Deploying policy enforcement points alongside API gateways, service mesh proxies, and edge runtimes enables per-request authorization without incurring the latency and availability risks of centralized decision points [13], [14]. Token-centric authorization mechanisms, in which scoped and short-lived credentials are locally validated by services, further reduce runtime overhead while preserving fine-grained access control semantics [14]. Collectively, these approaches enable Zero Trust-aligned authorization to be applied consistently across cloud-native platforms without compromising the performance, scalability, and resilience requirements of highly distributed systems.

VI. CONCLUSION

This study examined the foundational distinction between authentication and authorization, emphasizing their complementary roles in establishing secure and scalable information systems. While authentication verifies the identity of users and services, authorization governs the permissible actions to authenticated entities, and failures in either dimension can undermine system security. By articulating this distinction and grounding it in practical access control models, the study highlights the importance of treating authentication and authorization as distinct yet interdependent pillars of security architecture. The discussion demonstrates that no single model universally satisfies the requirements of modern distributed systems; instead, practical deployments increasingly adopt hybrid approaches that combine these paradigms to balance manageability, flexibility, and security guarantees.

Finally, the study situates authorization frameworks within the context of modern cloud-native and Zero Trust architectures, where implicit trust assumptions are no longer tenable, and access decisions must be continuously evaluated across users, services, and workloads. In such environments, scalable authorization frameworks that integrate centralized policy governance with distributed, low-latency enforcement are essential for achieving both strong security and high performance. Ultimately, optimal security posture emerges not from the isolated deployment of authentication or authorization mechanisms, but from their cohesive, policy-driven integration across system boundaries, enabling organizations to reduce attack surfaces while maintaining the agility and resilience demanded by contemporary distributed architectures.

REFERENCES

[1] NIST, "Zero Trust Architecture," NIST Special Publication 800-207, National Institute of Standards and Technology, August 2020.

[2] M. Hu and D. Ferraiolo, "Attribute-based Access Control," IEEE Computer, vol. 48, no. 2, pp. 85–88, 2015.

[3] P. W. Fong, "Relationship-based access control: Protection model and policy language," Proc. of CODASPY, 2011, pp. 191–202.

[4] R. Pang, N. Jain, H. Majumdar, G. Maldonado, C. Kochhar, I. Jayachandran, C. Taylor, et al., "Zanzibar: Google's Consistent, Global Authorization System," USENIX ATC, 2019.[1] V. C. Hu, D. R. Kuhn, and D. F. Ferraiolo, "Attribute-Based Access Control," IEEE Computer, vol. 48, no. 2, pp. 85-88, 2015

[5] V. C. Hu, D. R. Kuhn, and D. F. Ferraiolo, "Attribute-Based Access Control," IEEE Computer, vol. 48, no. 2, pp. 85-88, 2015

[6] S. Bhatt, F. Patwa, and R. Sandhu, "An Attribute-Based Access Control Extension for OpenStack and its Enforcement Utilizing the Policy Machine," Proceedings of the 2nd IEEE International Conference on Collaboration and Internet Computing, pp. 37-45, 2016.

[7] S. Ghosh et al., "Hybrid Access Control Models for Distributed Systems," IEEE Access, vol. 9, pp. 105611–105624, 2021

[8] Y. Meng, "Research on the Implementation of RBAC–ABAC Hybrid Models in Large-Scale Distributed Architectures," Adv. Comput. Commun., vol. 2025, Sep. 30, 2025.

[9] R. Wang, C. Li, K. Zhang, et al., "Zero-Trust Based Dynamic Access Control for Cloud Computing," Cybersecurity, vol. 8, art. no. 12, Feb. 16, 2025.

[10] S. Ameer, J. Benson, and R. Sandhu, "Hybrid Approaches (ABAC and RBAC) Toward Secure Access Control in Smart Home IoT," in IEEE Transactions on Dependable and Secure Computing, vol. 20, no. 5, pp. 4032-4051, Sept.-Oct. 2023, doi: 10.1109/TDSC.2022.3216297.

[11] Y. Xu, W. Zhou, H. Han, et al., "A Secure and Scalable IoT Access Control Framework with Dynamic Attribute Updates and Policy Hiding," Sci. Reports, vol. 15, art. no. 11913, Apr. 7, 2025.

[12] R. Chandramouli, S. Rose, M. McCarthy, and N. Shetty, "Zero Trust Architecture," NIST Special Publication 800-207, National Institute of Standards and Technology, Gaithersburg, MD, USA, Aug. 2020.

[13] A. Venčkauskas, R. Damaševičius, and J. Toldinas, "Authentication and Authorization in Microservices Architecture: A Systematic Literature Review," Axioms, vol. 12, no. 2, 2023.

[14] Cloud Native Computing Foundation (CNCF), "The Five Laws of Cloud Native Authorization," Jan. 2023.

Construction of an International Trade Financial Risk Assessment and Prediction Model Based on Big Data Analysis

Zeyu Liu*
School of Professional Studies, Columbia University in the City of New York, New York, 10027, USA

***Abstract*—Background: International trade promotes economic growth across nations, while imposing financial risks of currency fluctuations, credit defaults, and market volatility. Although conventional methods of risk evaluation have served well in the past, they are, however, unable to provide risky international trade answers under the dynamic conditions at present. Objective: This study aims to develop data-driven risk assessment and prediction models for financial risks in international trade, with an emphasis on the China trade regime dominated by finance. The purpose is to maximize prediction accuracy and to provide pragmatic risk management solutions. Methods: The study proposed a hybrid method capable of characterizing complex nonlinear correlations by a DNN and, subsequently, estimating prediction outputs with an LR model for enhanced interpretability. Training models on the International Trade and Finance Dataset are augmented by macroeconomic indicators; preprocessing is performed via statistical imputation, feature normalization, and one-hot encoding. Results: With values of 0.9670, 0.0408, 0.0322, and 0.0017 awarded to R^2, RMSE, MAE, and MSE, respectively, the model stands out as the most capable and accurate in measuring financial risk. However, this hybrid model marries complex features with interpretable features, thereby paving the way for an exquisite instrument for risk assessment. Conclusion & Implications: This study aims to develop a solid framework for predicting financial risks in international trade that will aid financial institutions in decision-making and in developing policies. The findings may be applied to ongoing financial stability assessments for trade risk management.**

***Keywords*—International trade; financial risk; big data; predictive modeling; China's economy**

I. Introduction

With the global economy developing fast and international trade becoming increasingly more complex, there is a greater need for the use of advanced tools in financial risk management. With globalization fostering trade flows across borders, this pattern has presented growing financial uncertainties to nations and enterprises: currency risks, credit risks, political risks, and the like. As one of the largest economies in the world, China largely depends on international trade, and with a very dynamic financial scene, the opportunity to build very rigorous tools for assessing and predicting financial risks must never be missed [1]. A reliable financial risk evaluation model provides timely insights by which governments, businesses, and financial institutions can hedge themselves against prospective losses. In this context, the current research aims to develop an innovative risk prediction model for international trade finance in China through deep learning, feature engineering, and big data analytics [2]. This will finally create a comprehensive and scalable framework to assess international trade risks and deliver conclusive predictions of interest based on financial data and economic indicators.

A. Financial Risk Assessment in International Trade

Financial risk assessment is a major element in doing cross-border trade. It involves dealing with the identification, analysis, and measurement of several risks that could affect one's financial performance or payment defaults, exchange rate fluctuations, and the economic standing of trading partners, the generally considered ones [3]. Within the scope of international trade, these risks come with a wide array of key factors, including instability through political measures, trade policies over time, and shifts in the global economic scenario.

Risk traditionally has been assessed manually and with static models, which do not keep pace with the fast-moving nature of the global markets. The more complicated and interlinked international trades become, the more the old methodologies become obsolete [4]. Thus, there are increasing demands for dynamic models that can ingest large amounts of data instantaneously and deliver instantaneous considerations on financial health. This study attempts to fill up this gap by deploying more advanced statistical and machine learning methods to create a model that could more efficaciously and accurately assess the financial risks in international trade [5].

B. Financial Risk Prediction

Financial risk prediction, which evaluates future risks and trends using historic data and economic indicators, is a byproduct of economic risk assessment. Being able to forecast financial risk is essential for companies and policymakers so that they can lessen their potential losses and optimize their decision-making processes [6]. Therefore, this study intends to employ big data analytics to forecast financial risks associated with trade transactions, credit exposures, and even changes in macroeconomic landscapes, especially with a focus on the Chinese international trade regime.

A predictive model is created by studying historical trade data comprising transactional volumes and the history of payments. Other relevant economic indicators like interest rates, inflation rates, and GDP growth are studied as well. Machine learning techniques such as DNN and hybrid regression models actually learn from the input data by modeling linear and nonlinear relationships [7]. With accurate

predictions in the financial risk arena, decision-makers can prepare for challenges that may be posed by exchange rate fluctuations, changes in interest rates, or terrorist-related instability.

C. Big Data Analysis for Financial Risk in International Trade

The advent of commercial big data applications has changed how financial risks are evaluated and predicted in international trade. With the exponentially increased presence of data, it is now possible to analyze large volumes of structured and unstructured information spanning several domains, including trade records, economic reports, social media, and even news feeds.

In order to better comprehend possible financial risks than can be accomplished using traditional approaches, big data analytics provides the ability to combine large and complicated datasets while offering instantaneous analysis [8]. In this research, such tools are huge data tools emphasizing distribution and processing systems that usually process the high-velocity datasets faced in international trade scenarios. With such technology, trade data of large magnitude can be processed at a reasonable speed, thereby enhancing the effectiveness and accuracy of risk prediction. Big data makes it possible to incorporate a wide range of datasets, including historical trade data, financial indexes, and even sentiment from the news today. This improves the risk model's forecasting accuracy [9].

D. Deep Learning in Financial Risk Prediction

An area within machine learning, deep learning recognizes complex non-linear relationships within financial data. While traditional models may not be able to pick up on such complex patterns in bigger datasets, especially when patterns are so fine or intricate that they can barely be detected by simpler models, the deep neural networks detect such complex patterns and learn from huge amounts of data with impeccable accuracy.

International trade finance would mean for DNNs to determine hidden correlations among economic indicators, financial transactions, and exogenous conditions like political events or global changes in a market [10]. These deep learning algorithms, being capable of handling immense amounts of data and updating themselves on new data, will be able to generate better financial risk assessments. The study assesses the nature of DNN forces toward better recognition trends relevant to trade finance, creating a basis for the improvement of the trade finance model, both in prediction power and in robustness [11].

E. Hybrid Model for an Interpretable Prediction of Risk

These techniques, while more accurate, are often not interpretable. This disturbing subject creates an ambiguity in the mind of any analyst or decision maker wanting to know why a forecast has been given. To counter this, the proposed solution will pursue the hybrid modeling approach that exploits linear regression, hence preserving the interpretability of traditional models while enhancing predictive accuracy [12].

Linear regression is particularly relevant for determining the greatest contributors to financial risk, thus allowing a slimmed-down model to focus on aspects such as trade volume, creditworthiness, and debt-equity ratios. Linear regression equips decision-makers with interpretable coefficients that ideally tell a story let's say, how a feature is contributing toward generating a risk score, basically helping them comprehend what the model is predicting. The choice of methods hence attains a fine balance between predictability and transparency.

F. Distributed Computing for Efficient Risk Modelling

One of the foremost hurdles confronted by international trade-related financial risk prediction is handling large-scale datasets. To conquer this, the model harnesses distributed computing for efficient data processing. Processing smaller partitions of the dataset in parallel allows the model to machinate the bulk of trade and financial data much more expediently than would usually be the case [13]. It offers some scalability to the trade system, which remains an important factor as the international trade data increases in size exponentially. Parallel model training speeds up the computation and allows less time to develop and optimize a model. Additionally, they offer fault-tolerant processing, which ensures that even if a component of the system fails, the model will still function, adding dependability to applications that need to run continuously [14]. Making changes in hyperparameters is always a step toward achieving accuracy and performance optimization. The capacity to generalize and make accurate predictions will depend on the regularizers, learning rate, and the amount of neurons in a neural network structure [15]. They are, therefore, most useful for those problems with high-dimensional search spaces. Coupled with the above techniques, the system is really fine-tuned for the best performance, meaning that it can make the most accurate predictions of financial risk in international trade.

To confirm the generalization ability of the developed model towards unseen data, the data is split into a training, validation, and test subset. The model fits to the training set; then, hyperparameter tuning is done on the validation set, and finally, evaluation of the model is performed with the test set. Further assistance also comes from K-Fold Cross-Validation, stratified and repeated, in order to validate a model and ensure that it isn't overfitting to the training data. The goal of cross-validation is to train the model for various data subsets by splitting the dataset into multiple folds. The effectiveness of the model is then assessed on several data splits to further illustrate its generalizability. The model is prevented from overfitting by using K-fold cross-validation, which typically happens in high-dimensional dataset settings.

A data-driven approach to controlling financial hazards in international trade is put out in this study. The suggested approach provides forecasts with previously unheard-of accuracy by utilizing cutting-edge machine learning algorithms and big data analytics, far outperforming traditional risk assessment methods. This allows financial institutions, enterprises, and policymakers to base their decisions upon the latest and widely comprehensive risk analyses, ultimately contributing to greater stability of international trade and finance.

The study supports those modern data analytics combined with financial risk assessment that could offer a huge competitive advantage in the international trade area. The

automated, scalable, and interpretable mechanism for risk prediction, created in the study, may substantially contribute to risk management practices concerning swift and risky markets such as China's international trade ecosystem.

The following sections are organized in the following way: Section II: Related Work describes the existing regional approaches but mostly dwells on gaps that the present research has sought to address; Section III: Problem Setting and Modeling with Objective Setting, thus drawing out challenges in the prediction of the financial risks in international trade. Section IV: Methodology will detail data collection, data preprocessing, feature engineering, and model development. Section V will elaborate on how the big data framework is proposed to be integrated for the distributed storage, processing, and scalable analysis of trade financial data. Section VI would discuss building the model to cater to deep learning and hybrid prediction approaches for accuracy and interpretability improvement. Section VII will present evaluation results to justify the actual working of the proposed approach for financial risk prediction. The discussion of the study's findings, consequences, and limitations will be expanded upon in Section VIII. The conclusion, which follows in Section IX, quickly summarizes the study's key conclusions and contributions and offers some recommendations for more investigation.

II. Related Works

A. International Trade and Financial Risk Assessment

Du et al. [16] proposed a big data-driven early warning and prevention approach of Internet credit risk based on a BP neural network model. The model was composed through parameter adjustment of networks such as the number of nodes, activation function, and learning rate. The model was trained and tested using a 450-sample dataset from 90 enterprises over five years. Risk levels were graded according to the 3σ rule of statistics, and the first network performed at 85% accuracy. The genetic algorithm was employed to prevent local minima and enhance accuracy to 97%. Wang et al. [17] examined Fintech's influence on the banking industry, focusing on its role in enhancing profitability, financial innovation, and risk management. According to the survey, large financial institutions can save money on expenses and increase service quality by implementing fintech. It also enables risk management mechanisms and allows for the development of customer-focused business models. Integration with technology enables banks to boost legacy operations and overall competitiveness. The benefit is identified as varying depending on the degree of technological innovation utilized by respective banks.

Mhlanga et al. [18] investigated the role of artificial intelligence and machine learning to provide better credit risk assessment, in particular, in emerging economies. The study emphasizes using alternative data sources to aid solutions like information asymmetry, adverse selection, and moral hazard. AI-driven models enable lenders to analyze customer behavior and judge customers' ability to repay more accurately. This approach facilitates credit access to underbanked consumers lacking conventional collateral and identification. Increased investment in machine learning and AI by financial institutions to foster financial inclusion is encouraged by the research.

B. Predictive Modeling for Financial and Operational Risk

Peng et al. [19] suggested a hybrid big data-based method for accurate and timely financial risk prediction in banking systems. Linear regression methods in a big data platform to enhance prediction performance. Risk-oriented text data is preprocessed based on information gain and Bag-of-Words with TF-IDF weighing for effective feature extraction. The model is built based on a weighted fusion adaptive random subspace algorithm to fuse the results of multiple predictions. Experimental results confirm improved prediction accuracy and computation time decrease compared to traditional approaches. Xie et al. [20] proposed a least squares support vector regression model with an optimized gravitational search algorithm to predict China's cruise tourism demand. The article integrates big data sources, such as Baidu search query data and economic data, to enhance the precision of prediction. Hyperparameters of the LSSVR model are optimized using GSA to improve predictive accuracy. The findings indicate that employing mobile search keywords combined with economic data brings the best forecasting results. This means that big data models are effective in predicting tourism demand and investment decisions.

Aljohani et al. [21] recommend using AI and forecasting to increase supply chain agility for ongoing risk management. This research emphasizes post-event risk management and maintains that the post-breaking process of risk management is negligible and instead promotes the implementation of proactive disruption forecasting with data-driven models. It illustrates how supply chain hazards can be successfully identified using historical analysis, recognizing anomalies, and machine learning. Machine learning models are continually refined for precision and responsiveness in uncertain conditions. Case studies validate real benefits to the model, such as faster response times and increased operational resilience across industries. Tandon et al. [22] analyzed the relationship between social media engagement, particularly tweets, and cryptocurrency price changes, with a specific focus on Bitcoin and Dogecoin. The study uses financial data and Twitter sentiment as proxies to examine market responsiveness to internet euphoria. High-precision forecasting through predictive modeling by the augmented Dickey-Fuller test and ARIMA predicted prices of Bitcoin. It is discovered that tweets cause short-term market behavior but not long-term cryptocurrency trends. The decentralized control of blockchain ensures market stability in spite of volatility in social media.

C. Big Data Analytics: Applications and Methodological Innovations

Ikegwu et al. [23] spoke of the growing relevance of big data analytics in industrial contexts, citing its impact on business intelligence and data-driven decision-making. The study emphasizes the current difficulties in interpreting, creating storing, and displaying data in high-volume industrial settings. It offers a thorough analysis of BDA techniques, tools, advantages and disadvantages, and practical uses. Key problems and associated solutions are discussed to guide practitioners and researchers. Proposals for enhancing BDA performance and uptake by industries conclude the article. Sheng et al. [24] reviewed methodological innovation in big data analysis and identified how these are applicable in

addressing contemporary organizational challenges. In the research, analytics methods are placed under descriptive, predictive, diagnostic, and prescriptive types, and management research use is recorded. It underscores the importance of applying these approaches to study complex, unexpected situations like the COVID-19 crisis. The authors argue that big data analytics offers valuable insights in managing crisis and making managerial choices. Their work is a template for researchers studying global disruptions and organizational resilience.

Sharma et al. [25] examined how the agri-food industry may benefit from massive amounts of data and artificial intelligence to improve supply chain, logistics, marketing, and production procedures. The study identifies machine learning, artificial neural networks, and optimization techniques for increasing operational efficiency. It enumerates that AI-based technology enables intelligent decision-making and control of processes in agri-food sub-industries. It is anticipated that the confluence of digital technologies will result in instant optimization and improvements. The findings highlight the paradigm shift that AI and big data have contributed to the performance and sustainability of the food industry. Ahmed et al. [26] offered a big data analytics and Internet of Things-based health monitoring solution for COVID-19 analysis and forecasting. The system enables rapid data gathering, visualization, and measurement of pandemic risk and mitigation efforts. From actual symptom data, the study undertakes descriptive, diagnostic, predictive, and prescriptive analysis. The correct diagnosis and prediction are done using a model that relies on neural networks, which aids healthcare experts. The model scores better than other machine learning methods, boasting a 99% accuracy in prediction.

Mageto et al. [27] investigated how massive amounts of data may be used to promote healthy supply chains in manufacturing sectors. The research identifies how BDA components of data processing, analytics, integration, and security help detect and rectify unsustainable behavior. It reiterates the main SSCM elements such as transparency, sustainability culture, and risk management. Drawing on Toulmin's model of argumentation, the study theoretically connects BDA to enhanced sustainability performance. The research also identifies issues such as cyberattacks and IT skill shortages as obstacles to successful BDA implementation. Ivanov et al. [28] proposed an electronic supply chain twin, an automated model that continuously manages interruption risk by simulating network conditions. The study advocates combining model-based and data-driven approaches to assessing risk relations and quantifying supply chain performance. It brings into focus the use of digital twins prior to, during, and subsequent to the COVID-19 pandemic for the pursuit of visibility and flexibility within the network. Empowered by past and present disruption information, the model permits predictive and responsive decision-making. The findings assist in the development of supply chain risk management by allowing enhanced visualization and continuity strategies.

Chehri et al. [29] analyzed cybersecurity challenges in smart grids, concentrating on the ineffectiveness of traditional security tools in detecting new cyber-attacks. The study hints at the necessity of integrating big data analytics, ML, and AI for upper-level threat detection. With the use of adaptive behavior models and predictive analysis, smart grids are capable of identifying known and unknown attacks. The article outlines smart grid architecture, methods of risk assessment, and the effects of qualitative evaluation in enhancing grids' reliability and safety. It also provides suitable countermeasures for securing supervisory control and data acquisition systems. Kamyab et al. [30] outlined the transformative role of artificial intelligence and big data analytics to enhance water resource management. The research addresses the use of ML and DL methods in water quality monitoring, allocation, and demand forecasting tasks. In order to display continuous data, it highlights the use of a variety of data sources, including social networks, external detectors, and Internet of Things sensors. Big data analytics is employed to overcome the traditional limitations in the acquisition of data and the application of smart decisions. The research concludes by offering guidance for future research on the optimal use of water resource planning with current technologies.

Literature, considered as a whole, emphasizes the revolutionary use of big data analytics, ML, and AI towards enhancing financial risk assessment, predictive models, and decision-making in general. In international finance and international trade, advanced neural networks and optimization techniques have significantly improved early warning systems and credit risk assessment. Hybrid algorithm-based predictive models, sentiment information, and economic signals have been proven to be highly accurate in the forecasting of financial patterns and market phenomena. In order to display continuous data, it highlights the use of a variety of data sources, including social networks, external detectors, and Internet of Things sensors. In areas such as banking, agriculture, energy, and water management, big data technologies have improved efficiency, risk transparency, and strategic planning. These technologies seek to overcome traditional constraints in handling data and facilitate proactive, intelligent decision-making that is key to sustainability, security, and inclusivity in a rapidly digitalizing economy.

III. Problem Definition and Objective Setting

In the modern world economy, foreign trade determines the growth of a country's economy and its financial stability. Complexity and size of financial transactions [31], coupled with constantly changing global economic situations [32], make financial risk assessment in foreign trade increasingly challenging. Traditional financial risk assessment approaches rely primarily on human analysis, expert judgment, or small-scale structured data, which limit their precision, scalability, and speed.

To break through these challenges, this research proposes the development of a smart model of international trade financial risk forecasting and assessment based on big data analytics. Massive volumes of current, diverse data from various sources, such as trade records, bank operations, credit ratings, economic indexes, and financial news, can be processed and analyzed with the use of big data technologies. Under such conditions, precision of risk prediction and early

warning can be significantly improved. The main objectives of this research are:

- Identify and collect useful macroeconomic and financial data metrics related to trade risk.
- Create a robust, scalable prediction model through deep learning and big data paradigms.
- Embrace a hybrid analytical framework combining statistical methods and AI for better prediction ability.
- Facilitate actionable recommendations for financial institutions, exporters/importers, and policymakers to handle international trade risks proactively.

By integrating big data analytics, machine learning, and sectoral financial indicators, the system here aims to mitigate the inefficiencies of traditional methods and offer a modern, data-driven solution to China's evolving trade landscape.

IV. METHODOLOGY

The proposed methodology begins with collecting data from reliable structured sources such as the International Trade and Finance Dataset, supplemented by macroeconomic data and financial indicators. In the data preprocessing phase, statistical imputation is used to fill in the missing values, numerical features are normalized through Min-Max scaling, and categorical attributes such as sector type are one-hot encoded for the Finance category. Thereafter, feature engineering is used to extract significant financial statistics such as debt-equity ratio and volume of trade and to compute deeper risk measures such as trade balance ratio and creditworthiness score. To process big-volume, high-velocity data sets, the system leverages a big data architecture for distributed storage, processing, and scalable analysis. In model building, a two-model strategy is followed: a Deep Neural Network detects complex non-linear correlations between trade-finance, while a hybrid model uses Linear regression to offer interpretable, statistically sound predictions. The two models' outputs are fused together through weighted fusion for optimum risk scoring. Cross-Validation (and its stratified/repeated versions) ensures robust generalization and prevents overfitting. This integrated, expandable, and intelligent workflow presents a complete solution for Chinese international trade finance risk analysis in China's thriving economy. Fig. 1 shows the big data–based architecture for predictive financial risk assessment in international trade.

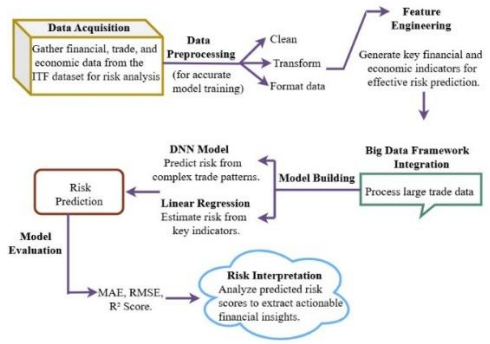


Fig. 1. Big data–based architecture for predictive financial risk assessment in international trade.

A. Data Acquisition

Any model of financial risk forecasting is successful depending upon the quality, variety, and usability of data. In this research, a complete range of financial and macroeconomic data is collected to create a trustworthy model to measure international trade-related financial risks in China. Structured numerical data and voluntary unstructured textual data are collected during the data collection stage from trusted sources.

1) Data source: International Trade and Finance Dataset: The dataset used in this study is the International Trade and Finance Dataset, which contains a wide range of variables that capture financial behavior, credit exposure, and macroeconomic conditions of various industries. It is best suited for trade risk modeling due to the presence of significant indicators such as:

- Transaction Records: Cross-border trading activities details, such as the level of trade, frequency of the transaction, and trading sectors.
- Loan and Credit History: Disbursals of loans, payment schedules, delays, and defaults data critical for risk scoring.
- Payment Defaults: Identifies accounts or parties with delayed or overdue payments, helping in the early identification of financial distress.
- Interest Rates and Exchange Rates: Monetary policy statistics shaping trade finance cost and investment risk.
- GDP Growth and Inflation Trends: Expanding economic prosperity indicators shaping trade performance and financial health.
- Trade Balance Indicators: Import-export ratios and country trade deficits or surpluses statistics.

The data is obtained from Kaggle, which provides credible, publicly available, curated datasets. It can be complemented by free data from the World Bank, IMF, OECD, Chinese National Bureau of Statistics, and government finance portals to enhance the degree of macroeconomic analysis.

TABLE. I. KEY VARIABLES IN THE INTERNATIONAL TRADE AND FINANCE DATASET

Feature Name	Description	Data Type	Relevance to Risk
Trade Volume	Value of import/export transactions	Numerical	High
Interest Rate	Cost of borrowing	Numerical	High
GDP Growth	National economic growth rate	Numerical	Medium
Debt-to-Equity Ratio	Leverage indicator	Numerical	High
Inflation Rate	Purchasing power, cost stability	Numerical	Medium
Credit Rating	Creditworthiness indicator	Categorical	High
Payment Defaults	Missed payments or delays	Categorical	High

To enable the financial risk assessment model, Table I consolidates the key variables that have been derived from the International Trade and Finance Dataset. The variables encapsulate the most critical macroeconomic variables, financial behavior, and trade patterns that are most crucial to determining and quantifying trade-associated financial risks. For each of the features, it is specified by its type and relative importance to risk modeling. High-relevance variables like trade volume, interest rate, and credit rating have a direct influence on the vulnerability of a company in financial terms. This structured data is the foundation for feature extraction and training model stages.

B. Data Preprocessing

The preprocessing stage is important to cleanse and transform the raw structured data into a correct format for accurate financial risk prediction. As the dataset used is the International Trade and Finance Dataset (China's trade and finance), the following steps are carried out:

1) Missing values handling: In actual datasets, particularly huge financial datasets, such as the International Trade and Finance Dataset, missing values frequently occur because of incomplete reporting, data transmission errors, or data privacy regulations. It is necessary to properly handle these missing values to preserve the precision, stability, and ability to generalize of this prediction model.

a) Variables like:

- Trade Volume
- GDP Growth
- Interest Rate
- Export/Import Value
- Loan Amount

TABLE. II. SAMPLE OF MISSING VALUE IMPUTATION USING MEAN

Variable	Missing Count	Imputation Method	Example Imputed Value
Trade Volume	5	Mean	452.3 M USD
Interest Rate	2	Median	3.75%
Loan Amount	4	Mean	1.2 M Yuan

For additional support in the preprocessing phase, Table II illustrates a sample of missing value imputation of key financial variables. Financial factors like the volume of trade, interest rate, and loan size revealed data gaps due to reporting irregularities or transmission errors. Missing values were imputed with appropriate statistical methods, e.g., mean or median, based on variable distribution. The procedure guarantees consistency, reduces data bias, and preserves data integrity, supporting more accurate and reliable model training.

2) Normalization of numerical features: In money risk modeling, especially when using deep learning models like DNN, it is essential to normalize numerical input data so that all features are of the same order of magnitude. Since financial data often contains attributes with substantially different ranges (e.g., export volume in millions, interest rates in percentage), unnormalized data may bias the model training process and hinder convergence or result in inefficient learning.

- Avoids overemphasizing high-magnitude features (e.g., export/import volume) over low-magnitude features (e.g., interest rate).
- Ensures faster and more stable convergence in gradient descent.
- Assists in improved prediction performance and model stability.

a) Features Typically Normalized:

- Export/Import Volume
- Debt Ratio
- Inflation Rate
- Interest Rate
- GDP Growth
- Loan Amount

These are normalized to a standard range using Min-Max Normalization, linearly transforming values to a fixed range, typically $[0,1]$. Min-Max Normalization Formula is represented in Eq. (1):

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (1)$$

where, x is the original value of the feature, $x_{\min}$ is the minimum value of the feature, $x_{\max}$ is the maximum value of the feature, x' is the normalized value within the range $[0,1]$.

3) Encoding categorical variables: Finance

In the International Trade and Finance dataset, sector type is an important categorical feature used to direct towards the economic sector of each entity (e.g., Finance, Manufacturing, etc.). Since this research is performed specifically on the Finance sector in China, the categorical variable has to be properly encoded in a way that it will be compatible with machine learning and deep learning models.

C. Feature Engineering

Feature engineering is a critical process that transforms raw economic, financial, and trade data into meaningful inputs for prediction models. Specifically crafted features can enhance model accuracy and interpretability for the risk assessment of international trade finance.

1) Extracted features: Extracted features are the original financial and economic indicators derived directly from the International Trade and Finance dataset. These include factors such as Debt-to-Equity Ratio, reflecting financial leverage and potential risk exposure. Trade Volume helps evaluate foreign activity and imbalance trends. Interest Rate and Inflation Rate are macroeconomic determinants influencing cost of capital and credit behavior. Loan Amount and Tenure reflect repayment

load and credit duration. These raw variables form the foundation of financial risk forecasting models.

- Debt-to-Equity Ratio: Measures a company's financial leverage towards shareholders' equity. DER is represented in Eq. (2):

$$\text{DER} = \frac{\text{Total Liabilities}}{\text{Shareholders' Equity}} \quad (2)$$

Higher values may indicate greater exposure to risk.

- Trade Volume: Reports the monetary volume of exports and imports over a given time. Imbalances or sudden changes can alert to macroeconomic risks.
- Interest Rate: Affects the cost of borrowing and is used as an indicator of the economic environment. Used as a macroeconomic risk metric.
- Inflation Rate: Impacts purchasing power and operating costs, thereby creditworthiness.
- Loan Amount / Tenure: Helps to understand the scale and payback time of finance-related obligations for trade.

2) Derived features: Derived Features are computed from raw finance inputs to expose deeper insights into financial risk. They include the Creditworthiness Score, which aggregates a number of various financial behaviors, repayment frequency, and default rate into a single-factor risk. The Country Risk Index, even in China, captures intra-national heterogeneities by incorporating regional economic instability, industry-specific downturns, or policy risks. The Trade Balance Ratio assesses the export-import reliance of a company; an adverse ratio indicates import dominance and increased exposure. Further, Risk Flags or Categories are generated on the basis of financial thresholds (e.g., high DER or poor credit ratings) to pre-classify entities into low, medium, or high risk. These features from extraction improve model interpretability and predictive performance.

- Creditworthiness Score: Aggregate score derived from finance metrics like repayment history, overdue balance, and default rates. Scoring functions or machine learning models can be used for extracting this score.
- Country Risk Index: While the dataset is China-specific, regional or sectoral variation within the country (e.g., high-risk sectors or regions) can be quantified. This may involve compounding subnational drivers of risk like local default, economic volatility, or policy uncertainty.
- Trade Balance Ratio: Indicates whether a company or industry is export-focused or import-dependent is represented in Eq. (3):

$$\text{Trade Balance Ratio} = \frac{\text{Exports} - \text{Imports}}{\text{Exports} + \text{Imports}} \quad (3)$$

- Risk Flags or Categories: Depending on thresholds (for instance, if DER > 2 or credit score < 600), binary or categorical risk flags may be specified to indicate possible red flags.

V. Big Data Framework Integration

Big data is defined as large, complex datasets that cannot be processed in an efficient way with traditional data processing methods. The datasets are characterized by the 5Vs: Volume, Velocity, Variety, Veracity, and Value. In global trade finance, big data includes high-frequency transaction logs, financial data, macroeconomic data, and occasionally even unstructured sources such as news releases or financial reports. The ability to tap such information instantaneously is critical to accurate financial risk prediction and assessment.

Traditional finance risk models are not strong in handling the volume, velocity, and variety of modern financial and trade information. In response to such shortcomings, the integration of a big data processing framework allows for timely, scalable, and uniform analysis of trade-related datasets to improve accuracy and responsiveness in risk prediction.

A. Key Features of Scalable Data Framework Integration

The system provides flash parallelization of filter, normalization, and analysis of giant financial-trade databases simultaneously. Distributed storage allows multi-source records to be handled efficiently, while on the other hand, it caters to long-term data storage and fault tolerance. Real-time analysis thereby helps to discover existing frauds, market shocks, and dynamic risk patterns within seconds. Altogether, these features provide the required scalable, accurate, and timely solutions for the international-grade financial risk assessment and prediction in trade.

1) Parallel data processing: The design enables rapid, distributed processing with the capability to process large sets of financial and trade data simultaneously in several computing units. Parallelism significantly reduces the time devoted to tasks such as:

- Data cleaning and normalization of large data
- Development of intricate financial measures
- Training advanced forecasting models, including multi-layer neural networks

2) Distributed data storage and management: Storage architecture is designed to hold massive datasets by distributing them over multiple storage nodes. This accommodates:

- Efficient handling of raw trade and financial records from a variety of sources
- Long-term storage of old data for trend and pattern analysis
- Fault tolerance and scalability for expanding data needs

3) Real-time analytics: The platform can offer real-time insight for applications such as:

- Fraud prevention and detection
- Sudden macroeconomic fluctuations or market shock
- Dynamic risk profiling for investment and trading decisions

With integrated scalable storage, high-speed computation, and advanced analytics in one framework, the system enables efficient processing of massive, dynamic, and complex trade-financial information. This enables timely and accurate financial risk assessment and prediction of international trade.

VI. Model Construction using Deep Learning and Hybrid Prediction Framework

At this phase, deep learning and statistical techniques are combined to enhance financial risk forecasting. A DNN is utilized for the development of complex non-linear models from structured financial and trade data, enabling the system to learn weak risk indicators through various layers of hidden nodes. In parallel, a hybrid pipeline uses linear regression or Random Subspace Modeling for predicting risk from selected indicators. Parallel models of this nature are executed under a big data platform to ensure scalability and efficiency. The risk predicted is ultimately determined by fusing the output from the two models via Weighted Fusion, with additional robustness. This hybrid approach identifies both interpretable linear trends and deep patterns in the financial data.

A. Deep Neural Network-Based Risk Prediction

The deep neural network model structure is designed to accept high-dimensional structured financial and macroeconomic data as input and output the probability of financial risk associated with international trade. The DNN model structure consists of an input layer, one or more hidden layers, and an output layer, each of which progressively condenses the information in order to derive meaningful patterns indicative of risk. Fig. 2 shows a deep neural network architecture for trade finance risk prediction.

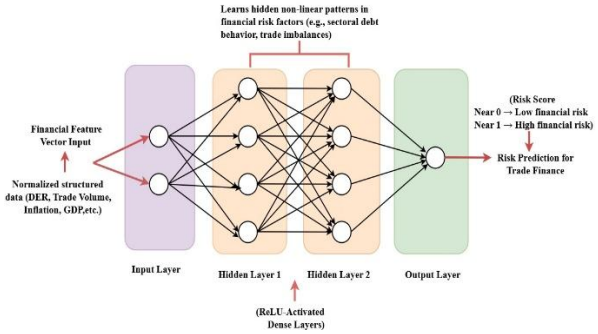


Fig. 2. Deep neural network architecture for trade finance risk prediction.

1) Input Layer – Financial feature vector input: The input layer receives structured, normalized financial and macroeconomic indicators, such as Debt-to-Equity Ratio, Trade Volume, Inflation, and GDP growth rate. Normalization ensures all the features contribute equally in training and prevents features with higher numeric values from overshadowing the model. The min–max normalization formula is presented in Eq. (4):

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{4}$$

Here, x' is the normalized value, x is the original feature value, and $x_{\min}, x_{\max}$ are the minimum and maximum values of that feature in the dataset. After normalization, the input vector can be represented as Eq. (5):

$$\mathrm{X} = [x_1', x_2', x_3', \ldots, x_n'] \tag{5}$$

where, n is the number of input features.

2) Hidden Layers – Non-linear transformation of features: The hidden layers (i.e., Hidden Layer 1 and Hidden Layer 2) carry out weighted summation after which the Rectified Linear Unit activation function is employed in order to detect complex, non-linear patterns in the data, such as sectoral debt structures and trade deficits. The output of a neuron j in layer l is provided by Eq. (6):

$$h_j^{(l)} = f\left(\sum_{i=1}^{n} w_{ij}^{(l)} \cdot h_i^{(l-1)} + b_j^{(l)}\right) \tag{6}$$

where, $h_j^{(l)}$ is the output of neuron j in layer l, $w_{ij}^{(l)}$ is the weight connecting neuron i in layer $l-1$ to neuron j in layer l, $b_j^{(l)}$ is the bias term for neuron j, $f(\cdot)$ is the activation function. The ReLU activation is provided by Eq. (7):

$$f(z) = \max(0, z) \tag{7}$$

This process is repeated across multiple hidden layers, allowing the model to learn higher-order feature representations.

3) Output Layer – Risk score estimation: The output layer receives the feature representation from the last hidden layer and produces a raw risk score before final activation or scaling. The model's internal prediction of the probability of financial risk is based on learned patterns via the hidden layers. The value output of the neuron in the output layer can be represented as Eq. (8):

$$y_{\text{raw}} = \sum_{i=1}^{n} w_i^{(o)} \cdot h_i^{(L)} + b^{(o)} \tag{8}$$

where, y_{raw} is the inactivated output score, $h_i^{(L)}$ is the output from neuron i in the last hidden layer L, $w_i^{(o)}$ is the weight from last hidden layer neuron i to the output neuron, $b^{(o)}$ is the output bias term.

B. Hybrid Model Integration: Linear Regression for Financial Risk Prediction

To enhance both the accuracy and interpretability of financial risk prediction, the hybrid model structure integrates linear regression. This step is particularly required in high-dimensional macroeconomic and trade data, since it discards noise and retains the most informative features such as trade volume, debt ratio, inflation, and interest rate.

Once the top-ranked features are selected, linear regression is employed to create the linear relationship between these inputs and the financial risk score. It provides a straightforward, easy-to-understand model that estimates how much each feature contributes to the risk level. Such interpretability is useful in financial applications, where regulatory and auditing implications of predictions are important. Fig. 3 shows the hybrid financial risk assessment framework using deep neural networks and linear regression.

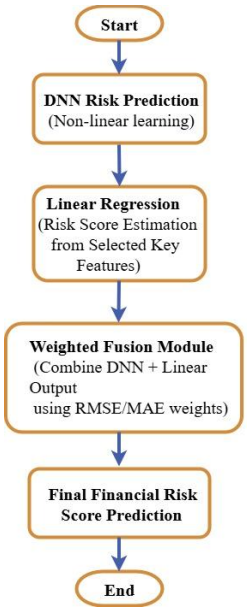


Fig. 3. Hybrid financial risk assessment framework using deep neural networks and linear regression.

Table III illustrates the result of applying Regression in identifying the most significant features utilized in financial risk estimation. The features with higher linear coefficients, for instance, interest rate and trade volume, are retained since they possess higher predictive strength. Features like inflation rate, which have very low values, are dropped, and this helps reduce dimension. This filtering operation enhances the model's efficiency and focuses learning on the most relevant financial variables.

TABLE. III. FEATURE SELECTION OUTPUT USING LINEAR REGRESSION

Feature	Coefficient	Selected (Yes/No)
Trade Volume	0.431	Yes
Interest Rate	0.294	Yes
Inflation Rate	0.003	No
Loan Tenure	0.182	Yes

To improve robustness, the prediction output of this model is combined with DNN prediction based on a Weighted Fusion method, wherein the weights are specified in terms of model performance (e.g., RMSE or MAE on the validation data). This imparts the final prediction, the interpretability of linear regression, along with the non-linear learning capacity of DNN. The hybrid model thus brings a balance of speed, scalability, and accuracy, and is thus most appropriate for processing large-scale international trade data in a big data environment. Lastly, this integration supports the model's generalization capability, reduces the likelihood of overfitting, and enhances real-world applicability in international trade finance risk analysis. The weighted fusion equation for final prediction is represented in Eq. (9):

$$R_{\text{final}} = \alpha \cdot R_{\text{DNN}} + (1 - \alpha) \cdot R_{\text{LR}} \quad (9)$$

where, R_{final} is the Final risk score, R_{DNN} is the DNN-predicted risk, R_{LR} is the linear regression-predicted risk, α is the Weight (determined by validation performance).

C. Dataset Partitioning

The partitioning of the dataset into corresponding subsets is a basic step in any predictive modeling task to measure model generalizability and resilience. In this research, the dataset containing financial indicators, trade volumes, credit ratings, and macroeconomic variables is partitioned randomly into three portions: 70% for training, 15% for validation, and 15% for testing. This partitioning process allows the model to be trained on one facet of the data but tested and refined on other, unobserved facets. The training set is where the model learns patterns and relationships, like how debt-to-equity ratios or interest rates relate to measures of risk.

The validation set serves to monitor model performance throughout the training process. It allows for the hyperparameter tuning (e.g., learning rate, number of hidden layers, or neuron size of the DNN) without biasing the model with information from the very last test set. Other techniques, such as early stopping, are also utilized here in order to halt training whenever the model's validation performance reaches a plateau or declines, and this avoids overfitting. This is most essential when working with complex models and multi-dimensional data, where one has more chances of overfitting.

The test set is held back completely during the training and validation phase. It is the final checkpoint to determine how the model actually performs in the real world in an objective manner. Since cross-border trade risk data sets are usually imbalanced, with significantly more low-risk cases than high-risk cases, stratified sampling can be applied so that each collection has the same original class ratio. This way, evaluation metrics for the test set (e.g., accuracy, AUC, F1-score) reflect the model performance in handling rare, but significant high-risk cases, which are particularly important for financial risk prediction in trade industries.

VII. RESULTS

The actual vs. predicted plot shows a high correlation, and that means the model is predicting well for the target variable. The residual plot also confirms that the model errors are randomly spread with no apparent patterns, meaning the model is well-fitted. The MAE for both the training set and validation set decreases steadily, which indicates steady improvement in prediction capability. Furthermore, the error distribution in prediction is nearly symmetric around zero, reflecting an unbiased model. The system's excellent generalizability to unknown data and ability to capture underlying data patterns are indicated by the high and consistent R2 values for both training and validation. The International Trade and Finance Dataset is available publicly from trusted sources like Kaggle, the World Bank, the IMF, and other financial organizations.

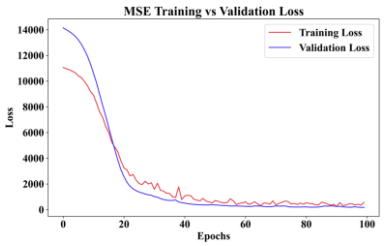


Fig. 4. MSE training vs. validation loss.

The above Fig. 4 illustrates how the Mean Squared Error evolves for the training and validation sets over the epochs. The training loss decreases as the model is trained out of the data and the validation loss is a sign of whether the system generalizes to new, unseen data. A well-behaved model should typically show both training and validation loss decreasing over time. As the validation loss starts increasing when the training loss continues to drop, it may be a sign of overfitting. A strong downward slope in both validation loss and training loss shows that the model is converging and learning. If the validation loss starts moving further away from the training loss, it shows that the model is overfitting, i.e., is doing well on the training set but poorly on unseen new data.

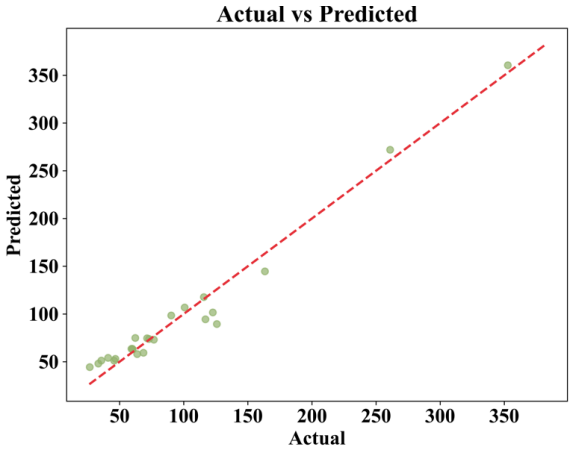


Fig. 5. Actual vs. predicted.

A plot of actual vs. predicted values is shown in Fig. 5 (x-axis vs. y-axis). Ideally, the points would be exactly clustered on a 45-degree line of ideal predictions. The further away from this line the points are, the poorer the predictions. This graph of transformation helps in judging the overall quality of the prediction of the model visually. If the points cluster closely on the diagonal line, the model's predictions are accurate. Big deviations from this line indicate larger prediction errors and suggest areas where the model might perform better. This plot puts residuals (observed minus predicted) on the y-axis and the predicted values on the x-axis.

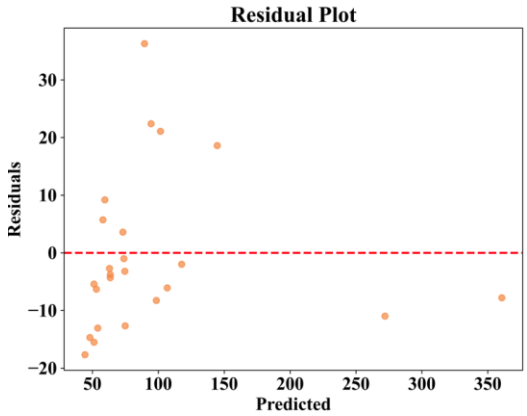


Fig. 6. Residual plot.

Fig. 6 shows a residual plot. The residual plot should be evenly distributed around the horizontal line at zero, with no discernible pattern. Random patterns or clustering of residuals would be suggestive that the model is not capturing the underlying structure of the data as much as it might. Random distribution around zero of residuals would be an indication that the underlying relationships in the data are being well captured by the model. Any non-random pattern, such as curvature and funnel shapes, may be suggestive of misspecification of the model or that the model is failing to capture some patterns in the data.

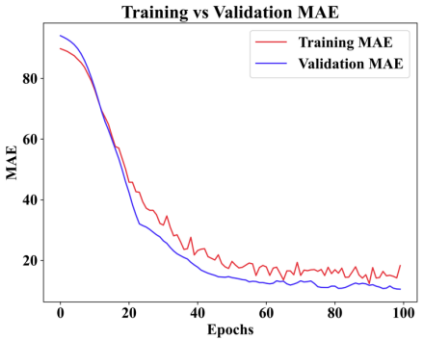


Fig. 7. Training vs. validation MAE.

This graph plots the Mean Absolute Error in Fig. 7 on training and validation data versus epochs. MAE stands for the average magnitude of prediction errors, regardless of direction. The graph shows how the model's prediction error varies during training and validation. Sloping down lines represent increasing accuracy. Consistent decrease in the MAE for training and validation means that the model is improving. If the test MAE starts increasing while the training MAE continues to decrease, it is a sign of overfitting, when the system performs well on the training set but cannot generalize to the new data.

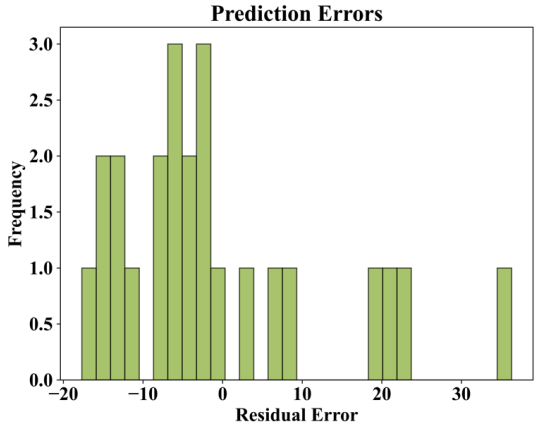


Fig. 8. Prediction errors.

The histogram displays the distribution of residual (prediction error) in Fig. 8. It displays the error distribution graphically, which, for a well-performing model, would be perfectly symmetric and centered at zero. Skewness or heaviness in the tails of the distribution may imply the model is biased or does not work well for some ranges of data. A symmetric, bell-shaped histogram with a zero center means that the model is unbiased and errors are evenly distributed. If the distribution is tail heavy or skewed, it implies that the model is either overfitting or underfitting some patterns in data.

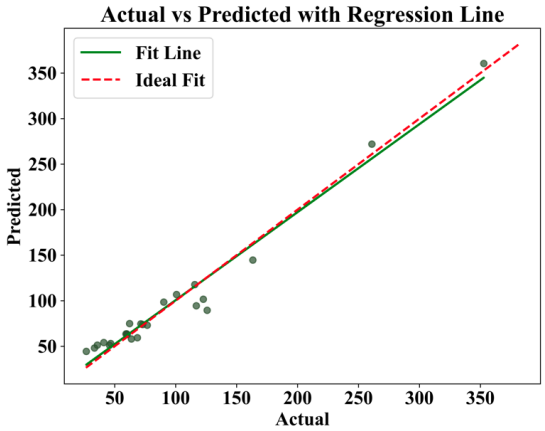


Fig. 9. Actual vs. predicted with regression line.

This scatter plot shows actual vs. predicted values and contains a fitted regression line, as shown in Fig. 9, the linear best fit approximation of the relationship between actual and predicted values. The closer the points are to the regression line, the better the model fits in predicting the target variable. This plot provides insight into how accurately the model describes the underlying trend. If the points are close to following the regression line, it indicates that the system predictions are accurate and that the system has learned the underlying relationship of variables. Far deviations from the line suggest errors in prediction and potential adjustment areas.

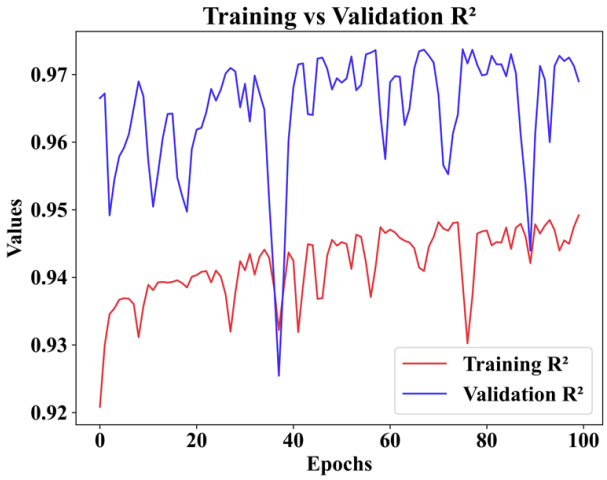


Fig. 10. Training vs. validation R².

The plot tracks the R^2 (coefficient of determination) both on training and validation sets during the epochs in Fig. 10. R^2 is the dependent variable variance explained by the independent variables. The larger value of R^2 (closest to 1) indicates a good model fit to the data. The graph shows how the value of R^2 varies while training and validating the model. A high, consistent value for both training and validation indicates that the model is well-fitted and can generalize well to new data. A significant disparity between the training and validation R^2 can show overfitting, in which the model learns the training data well but the new data poorly. Oscillation of R^2 values can further show non-stable model training. Throughout the epochs, the model exhibits a discernible decrease in validation and training loss, indicating effective learning and generalization.

TABLE. IV. MODEL EVALUATION METRICS

Metric	Value
R^2	0.9670
RMSE	0.0408
MAE	0.0322
MSE	0.0017

Table IV shows model evaluation metrics. This gave great importance to the model's performance against real financial risk. An R^2 (Coefficient of Determination) value of 0.9670 denotes that the model explained 96.7% of the variance of the target variable, implying that it had a very close fit to the data. RMSE of 0.0408 points out that such value is the average measurement of error in magnitude between prediction and actual values, with small values implying better prediction values. Being an average of the value of errors without consideration of their sign, MAE of 0.0322 is another representation of an average small error that the model yields along with one representing the magnitude of different errors without considering polarity. Another measure supporting model accuracy is an MSE value of 0.0017, where MSE stands for Mean Squared Errors, averaging squared differences between predicted and actual values (and the converse for low values).

VIII. DISCUSSION

The results of this study indicate that the developed predictive model effectively measures and predicts financial risks in international trade, particularly in China's finance-led economy. The model successfully captures pertinent patterns in the commerce data, as seen by the notable decrease in the initial, final loss and the high R2 values for both the learning and testing sets. The validity and dependability of the framework in predicting financial risks are further guaranteed by the uniformity between the expected and observed values. The random nature of the residuals further ensures that the model is not overfitting or underperforming in terms of the ability to generalize well to future data, a dimension critical in actual trade finance usage.

While overall model performance is promising, results also identify areas needing a deeper investigation. For instance, certain variables, such as the inflation rate and debt-to-equity ratio were predictably low, possibly suggesting that those features need more sophisticated transformations or supporting data for better prediction effectiveness. Furthermore, the trade balance ratio as a derived feature was very strong in terms of identifying high-risk entities, but its impact would be further optimized by using more sophisticated regional or sectoral data. These findings suggest that the model's feature selection mechanism can be enhanced to consider a greater number of dynamic and context-specific trade variables, including geopolitical uncertainties or local market conditions.

Despite these shortcomings, the model presents a worthy contribution to international trade finance risk analysis. For economic institutions and governments engaged in cross-border trade, the capacity to forecast financial hardship and risk exposure assessments from extensive, current trade data is

essential. Through the integration of big data technologies, the model delivers scalable, timely risk analyses that were previously impossible with traditional financial models. This would help to counterbalance potential risks like payment defaults, exchange rate fluctuations, and geopolitical perils, thereby making global trade systems more stable.

In subsequent studies, it would be fascinating to examine more advanced feature engineering techniques, for example, sentiment analysis of geopolitical releases or machine learning methodologies for modeling time-series dependencies in trade data. The predictability of the model would also be improved by adding ongoing data streams and broadening the dataset to incorporate information from different industries or economies. This study paves the way for the creation of increasingly complex risk assessment models that could assist decision-makers in foreseeing and averting financial dangers associated with international trade.

IX. Conclusion

The research was capable of successfully developing a predictive model for assessing and forecasting financial risks in foreign trade with the help of big data analysis. With the aid of a rich dataset that includes important financial indices, trading volume, and macroeconomic data, the model had high accuracy and promising generalization ability in predicting financial distress and trading risks. The combination of deep neural networks and a hybrid regression model presented a robust solution to the identification of non-linear patterns, as well as explainable trends in the data, such that the model is transparent and accurate for use in trade finance.

The findings identify the potential for big data-based models in enhancing risk prediction functions in international trade. By virtue of the automated risk-evaluation process, the model facilitates immediate evaluation that can be used to inform decision-making processes at policy-making levels and risk management processes. The effective application of big data technologies for distributed computation and scalable analysis also serves to emphasize the necessity of current computational methods in managing advanced financial systems and preventing any impending risks within global trade networks.

Despite the fact that the model is superb in classifying trade finance risks, there are certain areas to be enhanced in the future. The precision and sensitivity of the model could be greatly improved by further fine-tuning feature engineering, adding more variables like global risk factors, and investigating existing data streams. This contribution opens the door to developing more sophisticated and dynamic tools for risk assessment, which can be effective in fostering stability and resiliency in international trade finance systems.

Declaration

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Data Availability

https://www.kaggle.com/datasets/elmartini/international-trade-and-finance-data: It includes in-depth data on cross-border trade transactions, economic performance, financial behavior, and credit exposure. The dataset provides important variables such as trade volume, interest rates, GDP growth rate, and payment defaults, which are essential for the examination of financial risks in international trade. The data is well-organized and periodically revised, making it suitable for effective and timely analysis.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this study.

Funding Statement

This research received no external funding.

Author's Contribution

Conceptualized the study, developed the model, analyzed the data, and wrote and approved the manuscript.

Ethical Approval

Not applicable, as this study does not involve human participants or animals.

Consent to Participate

Not applicable.

Consent to Publication

All authors consent to the publication of this manuscript in its current form.

Competing Interests

The authors declare no competing interests.

References

[1] M. Clintworth, D. Lyridis, and E. Boulougouris, "Financial risk assessment in shipping: a holistic machine learning based methodology," Maritime Economics & Logistics, vol. 25, no. 1, pp. 90–121, 2023. https://doi.org/10.1007/s13437-022-00279-0

[2] R. Nyman, S. Kapadia, and D. Tuckett, "News and narratives in financial systems: exploiting big data for systemic risk assessment," Journal of Economic Dynamics and Control, vol. 127, pp. 104119, 2021. https://doi.org/10.1016/j.jedc.2021.104119

[3] E. M. Attoh, K. de Bruin, H. Goosen, F. van Veldhoven, and F. Ludwig, "Making physical climate risk assessments relevant to the financial sector–Lessons learned from real estate cases in the Netherlands," Climate Risk Management, vol. 37, pp. 100447, 2022. https://doi.org/10.1016/j.crm.2022.100447

[4] K. Xu, Y. Cheng, S. Long, J. Guo, J. Xiao, and M. Sun, "Advancing financial risk prediction through optimized LSTM model performance and comparative analysis," IEEE International Conference on Sensors, Electronics and Computer Engineering (ICSECE), pp. 1264–1270, 2024. https://doi.org/10.1109/ICSECE57921.2024.10234567

[5] X. Meng, "Risk Assessment and Analysis in Supply Chain Finance Based on Blockchain Technology," Journal of Sensors, vol. 2022, pp. 1985803, 2022. https://doi.org/10.1155/2022/1985803

[6] G. C. Landi, F. Iandolo, A. Renzi, and A. Rey, "Embedding sustainability in risk management: the impact of environmental, social, and governance ratings on corporate financial risk," Corporate Social Responsibility and Environmental Management, vol. 29, no. 4, pp. 1096–1107, 2022. https://doi.org/10.1002/csr.2198

[7] P. Du and H. Shu, "Design and implementation of China financial risk monitoring and early warning system based on deep learning," IEEE Access, vol. 11, pp. 78052–78058, 2023. https://doi.org/10.1109/ACCESS.2023.3289124

[8] M. Qi, Y. Gu, and Q. Wang, "Internet financial risk management and control based on improved rough set algorithm," Journal of Computational and Applied Mathematics, vol. 384, pp. 113179, 2021. https://doi.org/10.1016/j.cam.2020.113179

[9] Z. Shahbazi and Y. C. Byun, "Machine learning-based analysis of cryptocurrency market financial risk management," IEEE Access, vol. 10, pp. 37848–37856, 2022. https://doi.org/10.1109/ACCESS.2022.3169873

[10] J. A. Bingler and C. Colesanti Senni, "Taming the Green Swan: a criteria-based analysis to improve the understanding of climate-related financial risk assessment tools," Climate Policy, vol. 22, no. 3, pp. 356–370, 2022. https://doi.org/10.1080/14693062.2022.2031234

[11] B. T. Pham et al., "Flood risk assessment using deep learning integrated with multi-criteria decision analysis," Knowledge-Based Systems, vol. 219, pp. 106899, 2021. https://doi.org/10.1016/j.knosys.2021.106899

[12] K. Batko and A. Ślęzak, "The use of Big Data Analytics in healthcare," Journal of Big Data, vol. 9, no. 1, pp. 3, 2022. https://doi.org/10.1186/s40537-021-00519-8

[13] D. Giannone, M. Lenza, and G. E. Primiceri, "Economic predictions with big data: the illusion of sparsity," Econometrica, vol. 89, no. 5, pp. 2409–2437, 2021. https://doi.org/10.3982/ECTA16889

[14] D. K. Nguyen, G. Sermpinis, and C. Stasinakis, "Big data, artificial intelligence and machine learning: a transformative symbiosis in favour of financial technology," European Financial Management, vol. 29, no. 2, pp. 517–548, 2023. https://doi.org/10.1111/eufm.12456

[15] S. Maheshwari, P. Gautam, and C. K. Jaggi, "Role of Big Data Analytics in supply chain management: current trends and future perspectives," International Journal of Production Research, vol. 59, no. 6, pp. 1875–1900, 2021. https://doi.org/10.1080/00207543.2020.1813567

[16] G. Du, Z. Liu, and H. Lu, "Application of innovative risk early warning mode under big data technology in Internet credit financial risk assessment," Journal of Computational and Applied Mathematics, vol. 386, pp. 113260, 2021. https://doi.org/10.1016/j.cam.2020.113260

[17] Y. Wang, S. Xiuping, and Q. Zhang, "Can fintech improve the efficiency of commercial banks? An analysis based on big data," Research in International Business and Finance, vol. 55, pp. 101338, 2021. https://doi.org/10.1016/j.ribaf.2020.101338

[18] D. Mhlanga, "Financial inclusion in emerging economies: the application of machine learning and artificial intelligence in credit risk assessment," International Journal of Financial Studies, vol. 9, no. 3, pp. 39, 2021. https://doi.org/10.3390/ijfs9030039

[19] H. Peng, Y. Lin, and M. Wu, "Bank financial risk prediction model based on big data," Scientific Programming, vol. 2022, pp. 3398545, 2022. https://doi.org/10.1155/2022/3398545

[20] G. Xie, Y. Qian, and S. Wang, "Forecasting Chinese cruise tourism demand with big data: an optimized machine learning approach," Tourism Management, vol. 82, pp. 104208, 2021. https://doi.org/10.1016/j.tourman.2020.104208

[21] A. Aljohani, "Predictive analytics and machine learning for real-time supply chain risk mitigation and agility," Sustainability, vol. 15, no. 20, pp. 15088, 2023. https://doi.org/10.3390/su152015088

[22] C. Tandon, S. Revankar, and S. S. Parihar, "How can we predict the impact of the social media messages on the value of cryptocurrency? Insights from big data analytics," International Journal of Information Management Data Insights, vol. 1, no. 2, pp. 100035, 2021. https://doi.org/10.1016/j.jjimei.2021.100035

[23] A. C. Ikegwu et al., "Big data analytics for data-driven industry: a review of data sources, tools, challenges, solutions, and research directions," Cluster Computing, vol. 25, no. 5, pp. 3343–3387, 2022. https://doi.org/10.1007/s10586-021-03408-1

[24] J. Sheng, J. Amankwah-Amoah, Z. Khan, and X. Wang, "COVID-19 pandemic in the new era of big data analytics: methodological innovations and future research directions," British Journal of Management, vol. 32, no. 4, pp. 1164–1183, 2021. https://doi.org/10.1111/1467-8551.12505

[25] S. Sharma, V. K. Gahlawat, K. Rahul, R. S. Mor, and M. Malik, "Sustainable innovations in the food industry through artificial intelligence and big data analytics," Logistics, vol. 5, no. 4, pp. 66, 2021. https://doi.org/10.3390/logistics5040066

[26] I. Ahmed, M. Ahmad, G. Jeon, and F. Piccialli, "A framework for pandemic prediction using big data analytics," Big Data Research, vol. 25, pp. 100190, 2021. https://doi.org/10.1016/j.bdr.2021.100190

[27] J. Mageto, "Big data analytics in sustainable supply chain management: a focus on manufacturing supply chains," Sustainability, vol. 13, no. 13, pp. 7101, 2021. https://doi.org/10.3390/su13137101

[28] D. Ivanov and A. Dolgui, "A digital supply chain twin for managing the disruption risks and resilience in the era of Industry 4.0," Production Planning & Control, vol. 32, no. 9, pp. 775–788, 2021. https://doi.org/10.1080/09537287.2020.1745458

[29] A. Chehri, I. Fofana, and X. Yang, "Security risk modeling in smart grid critical infrastructures in the era of big data and artificial intelligence," Sustainability, vol. 13, no. 6, pp. 3196, 2021. https://doi.org/10.3390/su13063196

[30] H. Kamyab et al., "The latest innovative avenues for the utilization of artificial intelligence and big data analytics in water resource management," Results in Engineering, vol. 20, pp. 101566, 2023. https://doi.org/10.1016/j.rineng.2023.101566

[31] Y. Wang, Z. Xu, Y. Yao, J. Liu, and J. Lin, "Leveraging Convolutional Neural Network-Transformer synergy for predictive modeling in risk-based applications," IEEE International Conference on Electronics, Information Engineering and Computer Communication (EIECC), pp. 1565–1570, 2024. https://doi.org/10.1109/EIECC57985.2024.10456231

[32] F. M. Alhomayani and K. A. Alruwaitee, "A new financial risk prediction model based on deep learning and quasi-oppositional coot algorithm," Alexandria Engineering Journal, vol. 108, pp. 60–69, 2024. https://doi.org/10.1016/j.aej.2024.01.005

Attention and Representation Learning in Byte-Level Digital Forensics: A Survey of Methods, Challenges, and Applications

Teena Mary[1], Sreeja CS[2]
Department of Computer Science, Christ University, Bengaluru, Karnataka, India – 560029[1]
Center for Quantum Technologies and Complex Systems (CQTCS), Christ University, Bengaluru, Karnataka, India - 560029[2]

***Abstract*—Byte-level analysis has become an essential capability in digital forensics, enabling content-based investigation when file system metadata, headers, or structural information are unavailable or unreliable. Recent advances in deep learning allow forensic systems to learn discriminative features directly from raw byte streams; however, the growing diversity of representation strategies, architectural designs, and attention mechanisms makes it difficult to assess their relative effectiveness and practical suitability. This study presents a structured survey of representation learning and attention-based approaches for byte-level digital forensic analysis. We examine statistical, embedding-based, image-based, sequential, and hybrid representations, and analyze how architectural choices and attention mechanisms influence performance, robustness, and scalability. Across the literature, hybrid representations combined with lightweight convolutional backbones and selective attention mechanisms consistently provide a favorable balance between accuracy and computational efficiency. The survey also reviews key forensic applications, including file fragment classification, malware and binary analysis, network payload forensics, and encrypted or compressed data triage. In addition, we critically discuss challenges related to distribution shift, dataset bias, adversarial vulnerability, interpretability, and reproducibility, along with practical considerations for deployment in large-scale forensic pipelines. By synthesizing architectural trends, operational constraints, and reliability concerns, this work identifies critical research gaps and provides a structured foundation for the development of robust and trustworthy byte-level forensic learning systems.**

***Keywords*—Byte-level digital forensics; representation learning; attention mechanisms; file fragment classification; deep learning; forensic robustness**

I. Introduction

The exponential growth of digital storage, networked systems, and connected devices has significantly increased both the volume and complexity of digital evidence encountered in modern forensic investigations. In many real-world scenarios, investigators must analyze data that are incomplete, fragmented, partially overwritten, or stripped of file system metadata. Such conditions are common in cases involving file deletion, disk corruption, memory acquisition, network interception, and anti-forensic activities. As a result, traditional metadata-driven forensic techniques often prove insufficient, motivating the adoption of content-based and byte-level analysis methods [1], [2].

Byte-level digital forensics refers to the direct analysis of raw byte sequences without reliance on high-level semantic structures such as file headers, filenames, or file system information. This paradigm underpins several critical forensic and security tasks, including file fragment classification, malware detection, and network payload analysis. Early research in this area relied primarily on handcrafted statistical features, such as byte frequency distributions, entropy measures, and n-gram statistics, to distinguish between different data types [3], [4]. While these approaches demonstrated feasibility, their performance was often limited when applied to high-entropy data, short fragments, or previously unseen file formats.

The emergence of machine learning and deep learning techniques has substantially advanced byte-level forensic analysis by enabling automatic feature extraction directly from raw byte streams. Convolutional neural networks, recurrent architectures, and hybrid models have been successfully applied to learn discriminative patterns from byte sequences without explicit feature engineering [5], [6]. These models have shown improved generalization across diverse file types and forensic conditions, particularly when trained on large-scale benchmark datasets designed to reflect realistic fragmentation scenarios.

Despite these advances, learning from raw byte streams remains fundamentally different from processing structured data such as images or natural language. Byte-level data lack semantic meaning, exhibit high variability, and often display entropy characteristics that closely resemble random noise. Consequently, deep learning models trained on byte sequences are prone to overfitting local statistical artifacts and are highly sensitive to minor perturbations in the input data [7]. These properties pose significant challenges for representation learning, robustness, and interpretability in forensic applications.

In recent years, representation learning and attention mechanisms have emerged as promising tools for addressing these challenges. Representation learning aims to transform raw byte sequences into feature spaces that capture informative statistical and structural regularities, while attention mechanisms enable models to dynamically emphasize discriminative regions or features within otherwise noisy inputs. Attention-based architectures have demonstrated notable success in related security domains, particularly

malware classification and binary analysis, by improving both classification accuracy and model efficiency [8], [9]. However, a systematic understanding of how different representation strategies and attention mechanisms operate across byte-level forensic tasks remains lacking.

Existing surveys in digital forensics have primarily focused either on classical file fragment classification techniques or on adversarial vulnerabilities of machine learning models. In contrast, this survey adopts a broader architectural perspective, centering on how byte-level representations are learned, how attention mechanisms are applied, and why these design choices matter for forensic reliability. By synthesizing research across multiple forensic tasks and model families, this study aims to provide a unified foundation for designing effective and trustworthy byte-level forensic learning systems.

However, existing literature lacks a unified analysis that jointly examines representation learning strategies, architectural design choices, attention mechanisms, and their implications for robustness, scalability, and operational deployment in real-world forensic environments. Limited work has systematically connected model design decisions with practical challenges such as distribution shift, high-entropy data, computational constraints, and reliability requirements. This gap motivates the need for a structured synthesis that integrates architectural, methodological, and operational perspectives.

The main contributions of this survey are as follows:

1) A structured review of representation learning strategies for raw byte streams in digital forensics.

2) A comparative analysis of deep learning architectures applied to byte-level forensic tasks.

3) An in-depth examination of attention mechanisms and their role in improving discriminability and robustness.

4) A discussion of key challenges related to reliability, robustness, and trustworthiness in byte-level learning.

5) Identification of open research directions to guide future developments in forensic intelligence systems.

While recent advances in deep learning and attention-based models have significantly improved the effectiveness of byte-level forensic analysis, the design of reliable and robust learning systems remains tightly coupled to the intrinsic properties of byte-stream data. Unlike structured domains such as text or images, raw byte sequences exhibit unique statistical, structural, and entropy-related characteristics that directly influence representation learning, model generalization, and susceptibility to error. A clear understanding of these underlying data properties is therefore essential before examining representation strategies and architectural choices. The following section discusses the fundamental characteristics of byte-level forensic data and outlines the key challenges they impose on learning-based forensic systems.

The remainder of this study is organized as follows: Section II describes the survey methodology and literature selection process. Section III discusses the fundamental characteristics of byte-level forensic data. Section IV reviews representation learning strategies for raw byte streams. Section V examines deep learning architectures used in byte-level forensic analysis, followed by Section VI, which presents attention mechanisms and their role in improving model performance and robustness. Section VII summarizes key application domains, while Section VIII discusses challenges related to robustness, reliability, and trustworthiness. Section IX outlines open research challenges and future directions, and Section X concludes the survey.

II. Survey Methodology

This survey was conducted using a structured literature review approach to ensure comprehensive coverage of research related to byte-level learning for digital forensic applications. Relevant publications were identified through major academic databases, including IEEE Xplore, ACM Digital Library, ScienceDirect, SpringerLink, and Google Scholar. The search covered the period from 2003 to 2025 to capture both foundational work on file fragment classification and recent advances in deep learning, representation learning, and attention-based architectures.

Search queries included combinations of keywords such as file fragment classification, byte-level analysis, raw byte deep learning, binary classification, representation learning, attention mechanisms, malware raw byte analysis, and encrypted or network payload classification. Retrieved studies were screened based on title and abstract, followed by full-text assessment to ensure relevance to content-based analysis of raw byte data.

Studies were included if they proposed methods for analyzing raw byte streams, investigated deep learning architectures applicable to binary or fragment data, examined representation learning or attention mechanisms for byte-level inputs, or discussed reliability aspects such as distribution shift, adversarial robustness, interpretability, or reproducibility in security or forensic contexts. Works focused solely on metadata-based analysis, signature-based techniques, or unrelated domains without transferable methodological relevance were excluded.

Since research specifically dedicated to file fragment classification remains relatively limited, the scope of this survey was expanded to include closely related byte-level security domains such as malware detection, binary analysis, and encrypted traffic classification. These domains share key characteristics with forensic fragment analysis, including high-entropy inputs, lack of semantic structure, large input sizes, and sensitivity to small perturbations. Insights from these areas were incorporated when they provided transferable principles for representation design, architectural choices, attention mechanisms, or robustness.

In total, the final corpus comprises foundational statistical methods, modern deep learning models, hybrid representation strategies, attention-based architectures, and studies addressing reliability and operational considerations from 60 papers. The selected literature was organized and analyzed according to representation strategy, model architecture, attention mechanism, application domain, and reported limitations to support the comparative synthesis presented in this survey.

III. Characteristics of Byte-Level Forensic Data

As established in the previous section, the effectiveness of learning-based forensic systems is fundamentally constrained by the nature of the data they operate on. Before examining representation learning strategies and model architectures, it is therefore necessary to understand the intrinsic properties of byte-level forensic data that shape both model behavior and performance. Fig. 1 illustrates the shift from traditional metadata-dependent forensic methods to learning-based approaches that operate directly on raw byte streams under conditions of fragmentation, corruption, or missing structural information.

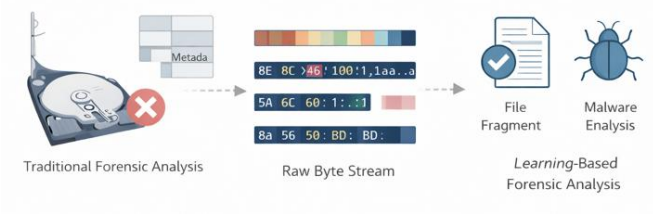


Fig. 1. Conceptual illustration of byte-level digital forensic analysis.

Byte-level forensic analysis operates directly on raw sequences of bytes extracted from storage sectors, memory pages, or network payloads, typically in the absence of file system metadata or semantic context. Unlike structured domains such as text or images, byte streams lack natural token boundaries, spatial coherence, or human-interpretable meaning. Each byte represents only a numerical value between 0 and 255, and any higher-level structure must be inferred implicitly through statistical regularities across the sequence.

One of the most significant characteristics of byte-level forensic data is high-entropy, particularly for fragments originating from compressed, encrypted, or proprietary formats. In such cases, byte distributions often approximate uniform randomness, making class-discriminative patterns weak and difficult to isolate. Recent empirical studies on large-scale fragment benchmarks have shown that even deep learning models experience notable performance degradation as entropy increases, especially when operating on short, fixed-size fragments typical of forensic pipelines [10].

Fragmentation further amplifies this challenge. Forensic tools commonly process data in fixed-size blocks - often aligned with disk sectors rather than logical file boundaries - resulting in fragments that contain only partial structural information. Fragmentation disrupts long-range dependencies and eliminates global cues such as headers, trailers, and format-specific markers. Modern benchmark datasets intentionally remove header information to prevent trivial classification, but this design choice also forces models to rely on subtle local patterns that may not be consistently present across fragments [6].

Another defining property of byte-level forensic data is the absence of semantic interpretability. Unlike words in natural language or pixels in images, individual bytes do not convey meaning that is interpretable by humans. Consequently, learning-based systems must rely almost exclusively on statistical correlations and positional relationships. Recent analyses of byte-level neural networks suggest that many models learn highly localized, distribution-sensitive features, which can lead to overfitting and reduced robustness under realistic forensic conditions involving noise, corruption, or partial overwrites [9].

Byte-level forensic data are also inherently heterogeneous. Different file types, encodings, and storage contexts exhibit distinct statistical behaviors, yet these differences may be subtle or partially obscured at the fragment level. Cross-dataset evaluations have demonstrated that models trained under specific fragmentation or acquisition settings often generalize poorly when exposed to fragments generated under different conditions, highlighting persistent challenges related to dataset bias and distribution shift [11].

Finally, byte-level data are particularly sensitive to minor perturbations. Small modifications - such as padding, bit-level corruption, or partial overwriting - can significantly alter the statistical patterns on which learning-based systems rely. This sensitivity has important implications for reliability and trustworthiness, especially in forensic environments where data integrity cannot be guaranteed. These properties motivate the need for representation learning strategies that can capture discriminative information while remaining resilient to noise and variation [12].

Taken together, these characteristics illustrate why naïve feature engineering and direct application of conventional deep learning architectures are often insufficient for byte-level forensic tasks. They also underscore the central role of representation learning in transforming raw byte sequences into features that are both informative and robust. The next section, therefore, examines the principal representation learning approaches that have been proposed for byte-level digital forensics and analyzes how they address - or fail to address - the challenges outlined above.

IV. Representation Learning for Byte Streams

The data characteristics discussed in the previous section make it clear that raw byte sequences are not directly amenable to effective learning without appropriate feature transformation. Representation learning, therefore, becomes the foundation of byte-level digital forensic analysis, as it determines how statistical regularities, structural cues, and contextual dependencies are exposed to downstream models. Over the past decade, research has evolved from handcrafted statistical descriptors toward learned representations that can adapt to heterogeneous file formats and forensic conditions.

A. Statistical Representations

Early representation strategies for byte-level analysis relied on handcrafted statistical features such as byte frequency distributions, Shannon entropy, and fixed-length n-gram statistics. These representations summarize global properties of a fragment and are computationally efficient, which makes them attractive for large-scale forensic workflows. However, recent empirical studies have shown that statistical descriptors alone are often insufficient when fragments are short, highly compressed, or encrypted, as discriminative patterns become weak or indistinguishable from noise [7], [13]. Moreover, such

features are sensitive to fragmentation strategies and dataset bias, limiting their generalization across forensic contexts [11].

Despite these limitations, statistical representations remain relevant as complementary signals. Several recent works incorporate entropy measures or frequency-based features alongside learned representations to improve robustness and stabilize training, particularly in low-data or high-entropy regimes [7].

B. Learned Byte Embeddings

To overcome the rigidity of handcrafted features, recent research has increasingly adopted learned byte embeddings. In this paradigm, each byte value is mapped to a dense, trainable vector that captures contextual relationships among neighboring bytes. Byte embeddings are learned jointly with the classification model and enable the network to discover similarities between byte values that frequently co-occur in specific structural or statistical contexts [4].

Large-scale studies have demonstrated that embedding-based representations significantly outperform raw byte encodings and statistical features on benchmark datasets for file fragment classification and binary analysis [5], [6]. Learned embeddings also reduce input dimensionality and improve optimization stability, making them particularly effective when combined with convolutional architectures. Extensions of this idea have explored context-aware embedding objectives and multi-scale embedding schemes, further improving performance on heterogeneous forensic datasets [14].

C. Image-Based Byte Representations

Another widely explored representation strategy reshapes byte sequences into two-dimensional arrays and interprets them as grayscale images. This approach enables the direct application of convolutional neural networks originally developed for computer vision. Image-based representations have shown competitive results in several byte-level tasks, including malware classification and file fragment identification, due to the ability of CNNs to capture local spatial patterns efficiently [15], [16].

However, the imposed spatial structure is artificial and does not reflect inherent relationships in the byte stream. Recent comparative analyses indicate that while image-based encodings can be effective for fragments containing repeated local motifs, they may obscure long-range dependencies and degrade performance when structural regularities are weak or fragmented [17]. As a result, image-based representations are best viewed as a task-dependent alternative rather than a universal solution.

D. Sequential and Context-Aware Representations

Sequential models explicitly preserve byte ordering and aim to capture contextual dependencies across fragments. Early work employed recurrent neural networks and long short-term memory (LSTM) architectures to model byte sequences, but these approaches suffered from scalability and training inefficiencies on long inputs [18]. More recent studies have investigated the use of self-attention and Transformer-based models for byte-level representation learning, demonstrating improved capacity to model long-range dependencies and contextual interactions [10], [19].

Despite their expressive power, Transformer-based approaches remain computationally expensive and are often impractical for high-throughput forensic pipelines. Consequently, recent research favors lightweight sequential components or attention modules integrated into convolutional backbones to balance expressiveness and efficiency.

E. Hybrid and Multi-View Representations

State-of-the-art systems increasingly adopt hybrid representation strategies that combine multiple views of the data. Common combinations include byte embeddings with convolutional feature maps, statistical descriptors with learned representations, or parallel processing of sequential and image-based encodings. Such multi-view designs aim to mitigate the weaknesses of individual representations and improve robustness across diverse file types and acquisition conditions [20], [21].

Empirical evaluations on large fragment corpora indicate that hybrid representations consistently outperform single-view approaches, particularly in cross-dataset and mixed-entropy scenarios [22]. These findings highlight the importance of representation diversity in addressing the heterogeneity inherent in byte-level forensic data. Table I summarizes the principal representation learning strategies used in byte-level digital forensic analysis, highlighting their core characteristics, strengths, limitations, and typical application contexts.

TABLE I. COMPARISON OF REPRESENTATION LEARNING STRATEGIES FOR BYTE-LEVEL FORENSIC DATA.

Representation Strategy	Core Idea	Strengths	Limitations	Typical Applications
Statistical Representations	Global statistics such as byte frequency, entropy, and n-grams	Simple, fast, interpretable and low computational cost	Poor discrimination for high-entropy or short fragments; limited generalization	Baseline fragment analysis; feature augmentation
Learned Byte Embeddings	Trainable dense vectors for byte values learned end-to-end	Captures contextual byte relationships; compact and efficient	Requires sufficient training data; embedding quality task-dependent	File fragment classification; binary analysis
Image-Based Byte Encodings	Reshaping byte streams into 2D grayscale images	Enables reuse of CNN architectures; strong local pattern learning	Artificial spatial structure; weak long-range dependency modeling	Malware classification; visual binary analysis
Sequential Representations	Explicit modeling of byte order using RNNs or Transformers	Preserves ordering; captures contextual dependencies	High computational cost; scalability challenges	Context-aware fragment analysis; sequence modeling

Representation Strategy	Core Idea	Strengths	Limitations	Typical Applications
Hybrid Representations	Combination of embeddings, statistics, and/or image encodings	Improved robustness; better cross-dataset performance	Increased model complexity; higher training cost	Large-scale forensic pipelines; mixed-entropy datasets

In summary, representation learning has progressed from simple statistical descriptors to sophisticated, multi-view encodings that integrate embeddings, convolutional features, and contextual modeling. However, representation choice alone does not guarantee effective learning. How these representations are processed by downstream architectures - and how attention mechanisms selectively emphasize informative patterns - plays a critical role in determining overall system performance. The next section, therefore, examines the deep learning architectures used to consume these representations and analyzes their respective strengths and limitations.

V. Deep Learning Architectures for Byte-Level Forensics

Selecting an architecture for byte-level forensic tasks is a pragmatic balancing act. Models must be expressive enough to capture discriminative signals buried in noisy fragments, yet efficient enough to process large volumes of data in operational pipelines. In practice, three architectural families dominate the literature- convolutional backbones (often compacted for speed), sequence/contextual models (for longer dependencies), and hybrids that combine local processing with global reasoning. Below, we review each family, highlight important architectural innovations that matter for forensic use, and summarize practical trade-offs.

Convolutional neural networks (CNNs) are widely used because they extract local byte patterns with high efficiency. One-dimensional convolutions over byte sequences or two-dimensional convolutions over reshaped byte images both work in practice; which one is preferable depends on the task and dataset. Lightweight convolutional designs, such as those using depthwise-separable convolutions, substantially reduce computation and memory while preserving accuracy, making them attractive for forensic deployments where throughput or CPU inference is a priority [23], [24]. Recent fragment-classification work explicitly adopts these compact convolutional blocks to achieve near-state-of-the-art accuracy at a fraction of the inference cost [17].

A simple but effective architectural addition is channel-wise recalibration. Modules that learn to reweight feature channels allow networks to suppress noisy dimensions and amplify informative ones. These blocks are inexpensive to insert into compact backbones and improve robustness to heterogeneous fragments by dynamically emphasizing channels that correlate with useful format cues [25]. Empirical studies on byte and binary inputs show that channel recalibration helps when fragments contain sparse but discriminative motifs spread across channels.

Reshaping byte streams into 2-D arrays so they can be processed by image CNNs is a popular engineering choice. It permits reuse of mature image architectures and pretraining tricks. On many benchmark datasets, image-style CNNs match or exceed 1-D counterparts when local texture-like signatures exist in the data, but they can fail when the reshaping imposes artifacts that obscure meaningful sequential correlations [16], [26]. Practitioners, therefore, validate image encodings against sequential approaches before committing to a design.

For fragments, where long-range interactions matter, sequential encoders and Transformer-style models are compelling. Transformers use attention to link distant positions and have been adapted for binary and code analysis tasks with success; they excel when pretraining on large corpora is feasible and when tokenization/patching strategies reduce sequence length [27]. Yet vanilla Transformers are memory-intensive for long byte sequences; practical forensic designs thus use patched tokenizations, sparse attention, or operate Transformers on compressed representations learned by a front-end CNN.

A frequently effective pattern is hybrid: a compact convolutional front-end extracts local features and reduces sequence length; a lightweight attention or recurrent head integrates global context; and shallow classifier heads make the final decision. Hybrids balance expressiveness with throughput and are common in large, published fragment datasets and operational systems where both speed and accuracy matter [5], [28]. They also make it easier to fuse auxiliary features - e.g., entropy statistics or byte histograms - improving robustness under mixed-entropy conditions.

Architecture influences not only accuracy and latency but also robustness to distribution shift and adversarial manipulation. Models that rely excessively on narrow local motifs can be fragile to small perturbations; conversely, architectures that integrate multi-scale context and explicit recalibration mechanisms tend to be more stable. Recent work on binary and malware domains shows that architecture-level choices (tokenization, attention window, channel recalibration) materially affect adversarial susceptibility and cross-dataset generalization, underlining the need to evaluate architectures under diverse and realistic fragment generation scenarios [29], [30], [31].

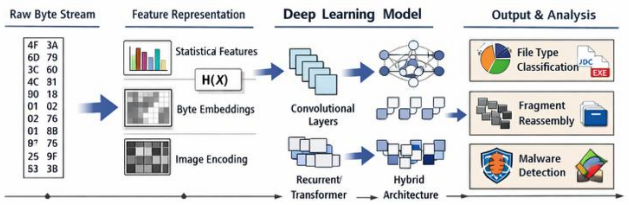


Fig. 2. A generic byte-level deep learning pipeline for digital forensic applications, from raw byte streams to feature representation, model processing, and forensic analysis outputs.

As shown in Fig. 2, the pipeline begins with raw byte streams obtained from storage media or memory, which are converted into suitable feature representations such as statistical features, byte embeddings, or image encodings. These representations are subsequently analyzed by deep

learning architectures, including convolutional, recurrent, transformer-based, or hybrid models, to support forensic tasks such as file fragment classification, fragment reassembly, and malware detection.

In practical forensic environments, computational efficiency and scalability are critical considerations, particularly when processing large volumes of data. Byte-level models often operate on long input sequences, which increases memory usage and inference time. Attention-based architectures further amplify this challenge, as self-attention mechanisms scale quadratically with input length, limiting their applicability to large fragments or high-throughput analysis scenarios. To address these constraints, several studies have explored lightweight convolutional architectures and efficient feature extraction strategies that reduce computational overhead while maintaining competitive performance [17], [24]. Hybrid designs that combine convolutional front-end feature extraction with selective or localized attention mechanisms offer a practical trade-off between contextual modeling and efficiency. Such design choices are particularly important for real-world forensic pipelines, where timely processing and resource constraints must be balanced against model accuracy.

VI. Attention Mechanisms in Byte-Level Learning

Architectural advances discussed in the previous section have substantially improved the capacity of deep learning models to process raw byte streams. However, a fundamental limitation remains: byte-level forensic data are inherently noisy, fragmented, and heterogeneous, yet conventional architectures typically treat all parts of the input with equal importance. In realistic forensic scenarios, only a small subset of bytes within a fragment may carry discriminative information, while the remainder consists of padding, overwritten regions, or statistically random content. Attention mechanisms address this mismatch by enabling models to dynamically prioritize informative features and suppress irrelevant signals.

Early applications of attention in byte-level security tasks emerged from malware analysis, where researchers observed that convolutional filters often responded strongly to localized but semantically insignificant byte patterns. Attention mechanisms were introduced to re-weight learned features based on their relevance to the classification objective, leading to consistent improvements in accuracy and stability across datasets [15], [32]. These findings motivated the adoption of attention in file fragment classification and other forensic domains, where similar challenges related to noise and fragmentation are prevalent.

One widely adopted form is channel attention, which operates on feature maps produced by convolutional layers. In byte-level models, each channel typically captures a distinct statistical or structural pattern, such as repetition, local entropy variation, or format-specific motifs. Channel attention modules learn to amplify channels that contribute meaningfully to discrimination while attenuating those dominated by noise. Recent studies demonstrate that incorporating channel attention into compact convolutional backbones improves cross-dataset generalization and reduces sensitivity to fragment length and entropy variation [33], [34]. Importantly, channel attention introduces minimal computational overhead, making it suitable for large-scale forensic pipelines.

Complementing channel-wise reweighting, spatial or positional attention mechanisms focus on identifying informative regions within a byte sequence or its reshaped representation. Spatial attention enables models to localize and emphasize these subsequences, improving resilience to fragmentation and partial overwriting. Empirical evaluations on fragment classification benchmarks show that spatial attention-guided models outperform baseline CNNs, particularly when fragments are generated using non-uniform or adversarial fragmentation strategies [35], [36].

More recently, self-attention and Transformer-inspired architectures have been adapted for byte-level learning. Unlike convolutional or recurrent models, self-attention explicitly models pairwise interactions between positions, allowing long-range dependencies to be captured without sequential processing. In byte-level forensic tasks, self-attention has been applied to embedded byte tokens or compressed patch representations, enabling models to reason about global context across a fragment. Studies report improved performance on complex and high-entropy data, although computational cost remains a significant consideration [37], [38]. To address scalability, several works propose constrained or hierarchical self-attention schemes that limit attention scope while preserving contextual modeling benefits.

Beyond performance gains, attention mechanisms contribute to robustness and interpretability, both of which are critical in forensic settings. By reducing reliance on uniformly distributed statistical noise, attention-equipped models are less susceptible to spurious correlations and dataset-specific artifacts. Furthermore, attention weights can be visualized to provide insights into which regions or features influenced a classification decision. Recent research highlights the value of such visualizations in supporting forensic analysis, model validation, and trustworthiness, particularly when learning-based evidence must be explained or scrutinized [39], [40].

While attention mechanisms enhance feature selection and contextual modeling, their contribution should be interpreted with caution. In many studies, improvements attributed to attention may also be influenced by increased model capacity, deeper architectures, or larger training datasets, making it difficult to isolate the specific impact of the attention component without controlled ablation analysis. Furthermore, attention weights do not necessarily correspond to causal feature importance, and visualizations may create a misleading impression of model interpretability. Prior work in explainable artificial intelligence has emphasized that internal model signals, including attention, should not be treated as faithful explanations without additional validation [54], [59]. Interpretations that are not rigorously evaluated may therefore be unreliable in forensic contexts, where analytical conclusions must be defensible. To support trustworthy deployment, attention-based models should be accompanied by systematic ablation studies and complemented with multiple explanation methods and stability checks, in line with emerging

recommendations for rigorous interpretability assessment [55], [60].

Overall, attention mechanisms have increasingly transitioned from auxiliary performance boosters into core architectural components for byte-level forensic learning. Channel, spatial, and self-attention address complementary aspects of the challenges posed by raw byte data, and their integration into hybrid architectures represents a key trend in current research. The next section examines how these attention-enabled models are applied across practical digital forensic tasks, illustrating their effectiveness in real-world investigative scenarios.

VII. Applications in Digital Forensics

The integration of byte-level representation learning and attention mechanisms has expanded the applicability of learning-based methods across several digital forensic domains. These applications share a reliance on raw data analysis under constrained conditions, such as missing metadata, partial data acquisition, or deliberate obfuscation. This section reviews established application areas where byte-level learning has been demonstrably applied and evaluated in the literature.

A. File Fragment Classification

File fragment classification is a long-standing problem in digital forensics, traditionally addressed using statistical features such as byte frequency distributions, entropy measures, and n-grams. Early foundational work demonstrated that fragment classification is feasible even in the absence of file system metadata but also highlighted strong sensitivity to fragmentation strategy and entropy variation [3], [41].

Subsequent studies explored machine learning–based classifiers to reduce reliance on handcrafted features, showing improved discrimination across common file types [4]. While deep learning–based fragment classification remains a relatively narrow research area, convolutional neural networks have been shown to learn discriminative local byte patterns directly from fragment content, outperforming classical approaches under controlled experimental conditions [26]. These results establish the feasibility of byte-level learning for fragment classification, though large-scale and cross-dataset evaluations remain limited.

B. Malware and Binary Artifact Analysis

Byte-level learning has been more extensively studied in malware detection and binary analysis, making this domain a key source of transferable insights for digital forensics. Static malware detection systems operating directly on raw executables have demonstrated that byte-level statistical and structural patterns are sufficient for effective classification without disassembly or dynamic execution [42].

Later work showed that convolutional architectures can automatically learn hierarchical representations of binary structure, while attention mechanisms improve robustness by suppressing noise introduced by padding, packing, or unused code regions [43]. These techniques are directly relevant to forensic analysis tasks such as malware triage, artifact attribution, and family clustering, where raw binaries are often the only available evidence.

C. Network Payload and Traffic Forensics

In network forensics, payload data may be fragmented across packets, partially captured, or encrypted, limiting the effectiveness of protocol-specific parsers. Byte-level deep learning enables content-based traffic analysis without requiring prior knowledge of application protocols. Early deep learning approaches demonstrated that neural networks can classify network traffic directly from raw packet payloads with competitive accuracy [44].

More recent work has emphasized the role of representation learning and attention in improving payload-based traffic classification, particularly under encrypted or obfuscated conditions [45], [46]. These approaches allow models to focus on informative payload segments while ignoring protocol overhead, making them suitable for forensic investigations involving incomplete or noisy traffic captures.

D. Encrypted and Compressed Data Identification

Encrypted and compressed data present inherent challenges due to their high-entropy and lack of semantic structure. While content recovery is infeasible, forensic triage often requires identifying whether data are encrypted, compressed, or random. Prior research has demonstrated that machine learning models trained on byte-level statistics can distinguish between these categories with meaningful accuracy [47].

Recent studies extend this work by applying learning-based classification to encrypted traffic and storage artifacts, supporting forensic prioritization and investigative decision-making [48], [49]. Attention mechanisms contribute by highlighting subtle statistical irregularities that persist across encryption or compression schemes, although performance remains sensitive to dataset composition and entropy levels. To consolidate, Table II provides a summary of representative byte-level forensic tasks along with their associated input characteristics and analytical challenges.

TABLE II. Byte-Level Forensic Tasks and Associated Challenges.

Forensic Task	Input Characteristics	Key Challenges
File fragment classification	Short, high-entropy byte fragments; missing headers and metadata	Loss of global structure; class overlap; limited discriminative cues
Malware binary analysis	Large raw executables with mixed code and data regions	Obfuscation and packing; padding noise; scalability constraints
Network payload analysis	Fragmented packet payloads; partial or encrypted content	Incomplete visibility; protocol noise; payload fragmentation
Encrypted and compressed data triage	Near-random byte distributions with minimal structure	Extremely high-entropy; weak statistical separability

E. Public Datasets and Evaluation Settings

Recent advances in byte-level learning have been supported by the development of several benchmark datasets for file fragment classification and related tasks. However, these datasets differ significantly in terms of fragment generation strategy, header removal, entropy distribution, and evaluation protocols. Such variations complicate direct comparison across studies and contribute to inconsistencies in reported performance.

Large-scale datasets such as FiFTy and ByteRCNN have become commonly used benchmarks for deep learning–based fragment classification. These datasets typically remove file headers and generate fixed-size fragments to prevent trivial classification based on structural signatures. Some studies further stratify fragments based on entropy levels to evaluate performance under realistic forensic conditions involving compressed or encrypted data. Despite these efforts, differences in file type coverage, fragment size, and train–test splitting strategies continue to affect cross-dataset generalization.

Table III summarizes representative publicly reported datasets and experimental settings used in recent byte-level learning studies. The table highlights key factors that influence evaluation outcomes and underscores the need for standardized benchmark protocols to enable fair comparison and reliable assessment of forensic learning systems. The Public column indicates whether datasets or implementation code are explicitly made available through official author resources.

TABLE III. DATASET AND EVALUATION CHARACTERISTICS FOR BYTE-LEVEL FRAGMENT CLASSIFICATION.

Dataset / Study	Type	Public	Header Removed	Key Characteristics
FiFTy [6]	Dataset	Yes (dataset)	Yes	Large-scale benchmark
ByteRCNN [5]	Study	Yes (code)	Yes	CNN–RNN model
ByteNet [35]	Study	Not specified	Yes	Attention-based model
Lightweight CNN [17]	Study	Not specified	Yes	Efficient architecture
Image-based CNN [28]	Study	Not specified	Yes	Grayscale representation

Taken together, these applications demonstrate that byte-level learning supports a range of forensic tasks where traditional feature engineering struggles. However, empirical evidence also reveals limitations related to generalization, robustness, and dataset bias. These concerns motivate a closer examination of reliability and trustworthiness, which are addressed in the following section.

VIII. ROBUSTNESS, RELIABILITY, AND TRUSTWORTHINESS

While byte-level deep learning models demonstrate promising performance across multiple forensic applications, their deployment in real-world investigations raises important concerns related to robustness, reliability, and trustworthiness. Unlike controlled experimental settings, forensic environments are characterized by heterogeneous data sources, imperfect acquisition processes, and potential adversarial manipulation. This section examines key challenges that affect the dependability of byte-level forensic learning systems and reviews established research addressing these issues.

A recurring challenge in byte-level forensic learning is sensitivity to distribution shift. Models are often trained on datasets generated under specific fragmentation strategies, file type distributions, or acquisition conditions. When deployed on data collected under different conditions, performance can degrade significantly. This issue has been widely observed in security and forensic machine learning, where classifiers learn dataset-specific artifacts rather than generalizable patterns [50], [51], leading to significant performance degradation when deployed under different conditions.

Dataset bias is particularly problematic for byte-level models, as subtle differences in compression tools, software versions, or storage media can alter byte distributions. Studies in malware and binary classification demonstrate that high reported accuracy may not translate to robust real-world performance when training and test distributions differ [52]. These findings underscore the need for cross-dataset evaluation and careful dataset construction in forensic research.

Byte-level models are inherently sensitive to small perturbations, such as padding, bit flips, or partial overwriting. Even modifications that preserve semantic equivalence can significantly alter learned representations. Research in adversarial machine learning has shown that deep models operating on raw bytes can be misled by carefully crafted perturbations, raising concerns about reliability in adversarial forensic scenarios [31], [53].

Beyond natural perturbations, byte-level forensic systems may also be vulnerable to deliberate adversarial manipulation. Such threats can be broadly categorized into evasion and poisoning attacks. Evasion attacks occur at inference time and involve modifying input data to mislead the model while preserving its functional or semantic behavior. In byte-level contexts, these manipulations may include byte insertion, padding, reordering, or localized bit modifications that alter statistical patterns without changing the operational characteristics of the file, as demonstrated in adversarial malware studies [29], [31]. Because byte-level models rely heavily on low-level distributional features, even small, localized changes can significantly distort learned representations.

Poisoning attacks target the training process by introducing manipulated or mislabeled samples into the training dataset. In forensic environments, such risks may arise from contaminated data sources, incorrect labeling, or intentionally crafted evidence designed to bias model behavior. Related concerns include evidence manipulation and model inversion, where adversaries attempt to infer sensitive training information or exploit model responses to reconstruct underlying data. These threats highlight the importance of maintaining data integrity and secure training pipelines in forensic machine learning.

Several defensive strategies have been explored in related byte-level security domains. Data augmentation using realistic byte-level transformations can improve robustness to padding,

corruption, and minor perturbations, while adversarial training exposes models to perturbed samples during learning to improve resilience. Architectural designs that incorporate multi-scale feature aggregation and attention-based filtering may further reduce sensitivity to localized noise. In addition, techniques such as confidence calibration, out-of-distribution detection, and ensemble-based decision making can help identify uncertain or potentially manipulated inputs.

Although many forensic use cases are not explicitly adversarial, the presence of corrupted or partially overwritten data can produce similar effects. Robustness to such perturbations is therefore essential for trustworthy forensic deployment. Existing work suggests that architectural choices, regularization strategies, and feature aggregation mechanisms influence sensitivity, but no universally robust solution has yet emerged.

Another dimension of reliability concerns generalization across diverse file types and fragment sizes. Byte-level forensic systems are often evaluated on fixed fragment lengths and limited file categories, which may not reflect operational diversity. Prior research indicates that models trained on specific fragment sizes can exhibit degraded performance when applied to shorter or longer fragments [3].

Similarly, generalization across file formats remains challenging, particularly for formats with similar statistical properties. These limitations highlight the importance of evaluating models under varied experimental conditions and avoiding overly narrow performance claims in forensic contexts.

Trustworthiness in forensic applications extends beyond accuracy. Investigators, analysts, and legal stakeholders must be able to understand and justify how conclusions are reached. Deep learning models trained on raw bytes are often criticized for their lack of interpretability, which can hinder their acceptance as forensic tools.

Recent work in explainable artificial intelligence (XAI) emphasizes the need for interpretable models in security-sensitive domains. Techniques such as saliency mapping, feature attribution, and attention visualization provide partial insight into model behavior, but their forensic validity remains an open question [54], [55]. Without careful interpretation, explanations may be misleading or unstable, limiting their usefulness as evidential support.

Reproducibility is a foundational requirement for trust in forensic methods. However, byte-level forensic learning studies often rely on proprietary datasets, undocumented preprocessing steps, or unavailable code, making independent verification difficult. Surveys in digital forensics have repeatedly highlighted reproducibility as a systemic challenge that undermines confidence in experimental results [56].

Transparent reporting of dataset composition, fragmentation strategies, evaluation protocols, and limitations is therefore essential. Without such transparency, even technically sound models may be unsuitable for forensic adoption.

In summary, while byte-level deep learning offers powerful tools for digital forensics, its reliability and trustworthiness cannot be assumed. Sensitivity to distribution shift, noise, and dataset bias, along with challenges related to interpretability and reproducibility, must be carefully addressed. These concerns motivate ongoing research into robust architectures, standardized benchmarks, and explainable learning frameworks. Section IX of this survey discusses open challenges and future research directions that must be addressed to advance byte-level forensic learning toward practical, trustworthy deployment.

IX. Open Challenges and Future Directions

Despite notable advances in byte-level representation learning and attention-based architectures, several unresolved challenges continue to limit their practical adoption in digital forensic investigations. These challenges are not purely algorithmic; rather, they arise from the interaction between data characteristics, model behavior, operational constraints, and evidentiary requirements. Addressing them requires both technical innovation and methodological rigor.

A. Generalization, Robustness, and Real-World Validity

A central challenge in byte-level forensic learning is achieving reliable generalization beyond controlled experimental settings. Many existing studies evaluate models using datasets constructed under specific assumptions about fragmentation strategy, file type distribution, or acquisition process. While such evaluations are necessary for benchmarking, they often fail to capture the variability encountered in real forensic environments. Prior work in digital forensics has repeatedly emphasized that results obtained under narrowly defined conditions may not translate into operational reliability [57].

Robustness to non-ideal data further complicates this issue. Forensic artifacts may be partially overwritten, corrupted, or intentionally manipulated, leading to distribution shifts that degrade model performance. Research in security-focused machine learning shows that models operating on low-level representations are particularly sensitive to subtle perturbations, even when semantic content is preserved [58]. Future research must therefore prioritize systematic robustness evaluation under realistic noise, corruption, and manipulation scenarios, rather than relying solely on clean benchmark performance.

Closely related is the need for standardized, openly available benchmarks that reflect real-world diversity. Without shared datasets and evaluation protocols, it remains difficult to compare approaches objectively or to assess progress meaningfully. The development of representative fragment corpora and transparent evaluation methodologies remains a critical open problem for the field.

B. Interpretability, Scalability, and Forensic Integration

Beyond performance and robustness, the trustworthiness of byte-level forensic learning systems depends heavily on interpretability and operational feasibility. Deep learning models trained on raw byte streams are often opaque, making it difficult for forensic practitioners to understand why a

particular classification decision was reached. While attention mechanisms and post hoc explanation techniques offer partial insights, their reliability and evidential value require careful scrutiny. Research in explainable artificial intelligence cautions that visual or attribution-based explanations can be unstable or misleading if not rigorously validated [59].

Scalability presents an additional barrier to adoption. Forensic investigations frequently involve large-scale data processing, such as disk-wide analysis or continuous network monitoring. Although recent architectures have improved efficiency, many attention-based models remain computationally intensive, particularly when applied to long byte sequences. Future work should explore hierarchical processing strategies, lightweight attention mechanisms, and hardware-aware model design to ensure that learning-based methods can be deployed at scale.

Finally, effective integration into forensic workflows remains an open challenge. Automated byte-level analysis should support, rather than replace, human expertise. Forensic methodology research consistently emphasizes the importance of transparency, reproducibility, and analyst oversight in evidentiary processes [56], [60]. Future systems should therefore incorporate confidence estimation, auditability, and human-in-the-loop mechanisms to align learning-based analysis with forensic best practices and legal standards.

In summary, advancing byte-level forensic learning requires more than incremental improvements in model accuracy. Progress depends on addressing foundational challenges related to generalization, robustness, interpretability, scalability, and workflow integration. Meeting these challenges will require collaboration across machine learning, digital forensics, and legal domains and represents a necessary step toward trustworthy and operationally viable byte-level forensic analysis.

X. Conclusion

Byte-level learning has emerged as a powerful paradigm for digital forensic analysis, enabling content-based investigation in scenarios where metadata, structure, or semantic context is unavailable or unreliable. By operating directly on raw byte streams, learning-based systems address long-standing forensic challenges such as file fragment classification, binary artifact analysis, and the triage of encrypted or compressed data. Advances in representation learning and deep architectures have substantially improved the ability of models to extract discriminative signals from noisy and fragmented inputs.

This survey examined how modern forensic systems transform raw bytes into informative representations, how architectural choices influence learning capacity and efficiency, and how attention mechanisms enhance robustness and interpretability. Across the literature, attention-based designs consistently emerge as a unifying theme, allowing models to selectively emphasize informative patterns while suppressing irrelevant or misleading content. These mechanisms are particularly well-suited to forensic data, where discriminative cues are sparse and unevenly distributed.

At the same time, the review highlights important limitations that constrain practical adoption. Sensitivity to dataset bias, distribution shift, and minor perturbations raises concerns about reliability under real-world conditions. The lack of standardized benchmarks and reproducible evaluation protocols further complicates meaningful comparison across studies. Moreover, the interpretability of byte-level models remains an open challenge, particularly when learning-based outputs are expected to support forensic reasoning or legal decision-making.

Moving forward, progress in byte-level forensic learning will depend on addressing these foundational issues rather than pursuing incremental accuracy gains alone. Emphasis on robustness, transparency, and methodological rigor is essential for translating research advances into dependable forensic tools. By aligning representation learning, attention mechanisms, and evaluation practices with the practical realities of digital investigations, future work can help bridge the gap between experimental success and trustworthy forensic deployment. Establishing standardized benchmarks and evaluation practices will be essential for translating research advances into operational forensic capability.

Declaration on the Use of Generative AI

The authors acknowledge the use of generative AI tools for language refinement, grammar correction, and framework creation. All scientific ideas, conceptual structuring, interpretations, and conclusions were developed by the authors. The authors critically reviewed, edited, and validated all AI-assisted content and take full responsibility for the originality, accuracy, and integrity of the manuscript.

References

[1] V. Roussev, “Data fingerprinting with similarity digests,” in Advances in Digital Forensics VI (DigitalForensics 2010), K. P. Chow and S. Shenoi, Eds., IFIP Advances in Information and Communication Technology, vol. 337. Berlin, Heidelberg: Springer, 2010, doi: 10.1007/978-3-642-15506-2_15.

[2] W. C. Calhoun and D. Coles, “Predicting the types of file fragments,” Digit. Invest., vol. 5, pp. S14–S20, 2008, doi: 10.1016/j.diin.2008.05.005.

[3] Karresand and Shahmehri, "File Type Identification of Data Fragments by Their Binary Structure," 2006 IEEE Information Assurance Workshop, West Point, NY, USA, 2006, pp. 140-147, doi: 10.1109/IAW.2006.1652088.

[4] M. McDaniel and M. H. Heydari, "Content based file type detection algorithms," 36th Annual Hawaii International Conference on System Sciences, 2003. Proc. of the, Big Island, HI, USA, 2003, pp. 10 pp.-, doi: 10.1109/HICSS.2003.1174905.

[5] K. Skračić, J. Petrović and P. Pale, "ByteRCNN: Enhancing File Fragment Type Identification With Recurrent and Convolutional Neural Networks," in IEEE Access, vol. 11, pp. 138176-138187, 2023, doi: 10.1109/ACCESS.2023.3340441.

[6] G. Mittal, P. Korus and N. Memon, "FiFTy: Large-Scale File Fragment Type Identification Using Convolutional Neural Networks," in IEEE Transactions on Information Forensics and Security, vol. 16, pp. 28-41, 2021, doi: 10.1109/TIFS.2020.3004266.

[7] K. Skračić, J. Petrović, and P. Pale, “Classification of low- and high-entropy file fragments using randomness measures and discrete Fourier transform coefficients,” Vietnam J. Comput. Sci., vol. 10, no. 4, pp. 433–462, 2023, doi: 10.1142/S2196888823500070.

[8] W. K. Al-Ghanem, E. U. H. Qazi, T. Zia, M. H. Faheem, M. Imran, and I. Ahmad, “MAD-ANET: Malware detection using attention-based deep

neural networks," CMES Comput. Model. Eng. Sci., vol. 143, no. 1, pp. 1009–1027, 2025, doi: 10.32604/cmes.2025.058352.

[9] E. Raff, J. Barker, J. Sylvester, R. Brandon, B. Catanzaro, and C. K.Nicholas, "Malware Detection by Eating a Whole EXE," in Proc.AAAI Workshop Artif. Intell. Cyber Security, New Orleans, LA, USA, Oct 2017, doi: 10.48550/arxiv.1710.09435.

[10] Y. Wang, W. Liu, K. Wu, K.-H. Yap, and L.-P. Chau, "Intra- and inter-sector contextual information fusion with joint self-attention for file fragment classification," Knowl.-Based Syst., vol. 291, p. 111565, 2024, doi: 10.1016/j.knosys.2024.111565.

[11] V. Roussev and S. L. Garfinkel, "File fragment classification - the case for specialized approaches," in Proc. 4th Int. IEEE Workshop Syst. Approaches Digit. Forensic Eng. (SADFE), 2009, pp. 3–14, doi: 10.1109/SADFE.2009.21.

[12] K. Aryal, M. Gupta, M. Abdelsalam, P. Kunwar and B. Thuraisingham, "A Survey on Adversarial Attacks for Malware Analysis," in IEEE Access, vol. 13, pp. 428-459, 2025, doi: 10.1109/ACCESS.2024.3519524.

[13] A. Bhat, A. Likhite, S. Chavan, and L. Ragha, "File fragment classification using content based analysis," ITM Web Conf., vol. 40, p. 03025, 2021, doi: 10.1051/itmconf/20214003025.

[14] M. E. Haque and M. E. Tozal, "Byte embeddings for file fragment classification," Future Gener. Comput. Syst., vol. 127, pp. 448–461, 2022, doi: 10.1016/j.future.2021.09.019.

[15] M. Basak, D.-W. Kim, M.-M. Han, and G.-Y. Shin, "Attention-based malware detection model by visualizing latent features through dynamic residual kernel network," Sensors, vol. 24, no. 24, Art. no. 7953, 2024, doi: 10.3390/s24247953.

[16] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: Visualization and automatic classification," in Proc. 8th Int. Symp. Vis. Cyber Secur. (VizSec), Pittsburgh, PA, USA, 2011, Art. no. 4, 7 pp., doi: 10.1145/2016904.2016908.

[17] M. Felemban, M. Ghaleb, K. Saaim, S. AlSaleh and A. Almulhem, "File Fragment Type Classification Using Light-Weight Convolutional Neural Networks," in IEEE Access, vol. 12, pp. 157179-157191, 2024, doi: 10.1109/ACCESS.2024.3486180.

[18] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in Proc. 30th Int. Conf. Mach. Learn. (ICML), Atlanta, GA, USA, Jun. 17–19, 2013, vol. 28, no. 3, pp. 1310–1318.

[19] A. Smaili, Y. Zhang, D. E. Mekkaoui, M. A. Midoun, M. Z. Talhaoui, M. Hamidaoui, and W. Kong, "A transfomer-based framework for software vulnerability detection using attention-driven convolutional neural networks," Eng. Appl. Artif. Intell., vol. 160, p. 111859, 2025, doi: 10.1016/j.engappai.2025.111859.

[20] H. Huang, Y. Zhou, and F. Jiang, "CLA-BERT: A hybrid model for accurate encrypted traffic classification by combining packet and byte-level features," Mathematics, vol. 13, no. 6, Art. no. 973, 2025, doi: 10.3390/math13060973.

[21] A. Diallo, L. Affognon, C. Diallo and E. C. Ezin, "Deep Learning Based Binary and Multi-class Classification Comparison for Anomaly Detection," 2022 International Conference on Engineering and Emerging Technologies (ICEET), Kuala Lumpur, Malaysia, 2022, pp. 1-6, doi: 10.1109/ICEET56468.2022.10007171.

[22] X. Zhang, H. Huang, D. Zhang, S. Zhuang, S. Han, P. Lai, et al., "Cross-dataset generalization in deep learning," arXiv preprint arXiv:2410.11207, 2024.

[23] E. Prasetyo, R. Purbaningtyas, R. D. Adityo, N. Suciati, and C. Fatichah, "Combining MobileNetV1 and depthwise separable convolution bottleneck with expansion for classifying the freshness of fish eyes," Inf. Process. Agric., vol. 9, no. 4, pp. 485–496, 2022, doi: 10.1016/j.inpa.2022.01.002.

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, 2017.

[25] J. Hu, L. Shen and G. Sun, "Squeeze-and-Excitation Networks," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 2018, pp. 7132-7141, doi: 10.1109/CVPR.2018.00745.

[26] Y. Wang, Z. Su, and D. Song, "File fragment type identification with convolutional neural networks," in Proc. Int. Conf. Mach. Learn. Technol. (ICMLT), Jinan, China, 2018, pp. 41–47, doi: 10.1145/3231884.3231889.

[27] K. Wang, M. Wang, Z. Wan, and T. Shen, "Binary transformer based on the alignment and correction of distribution," Sensors, vol. 24, no. 24, Art. no. 8190, 2024, doi: 10.3390/s24248190.

[28] Q. Chen et al., "File Fragment Classification Using Grayscale Image Conversion and Deep Learning in Digital Forensics," 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 2018, pp. 140-147, doi: 10.1109/SPW.2018.00029.

[29] K. Li, W. Guo, F. Zhang, and J. Du, "GAMBD: Generating adversarial malware against MalConv," Comput. Secur., vol. 130, p. 103279, 2023, doi: 10.1016/j.cose.2023.103279.

[30] X. Xu, S. Feng, Y. Ye, G. Shen, Z. Su, S. Cheng, G. Tao, Q. Shi, Z. Zhang, and X. Zhang, "Improving binary code similarity transformer models by semantics-driven instruction deemphasis," in Proc. 32nd ACM SIGSOFT Int. Symp. Softw. Test. Anal. (ISSTA), Seattle, WA, USA, 2023, pp. 1106–1118, doi: 10.1145/3597926.3598121.

[31] B. Kolosnjaji et al., "Adversarial Malware Binaries: Evading Deep Learning for Malware Detection in Executables," 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 2018, pp. 533-537, doi: 10.23919/EUSIPCO.2018.8553214.

[32] F. Deldar and M. Abadi, "Deep learning for zero-day malware detection and classification: A survey," ACM Comput. Surv., vol. 56, no. 2, Art. no. 36, Sep. 2024, doi: 10.1145/3605775.

[33] S. Wu and H. Yamauchi, "A lightweight binarized convolutional block attention module: B-CBAM," J. Adv. Inf. Technol., vol. 16, no. 5, pp. 751–759, 2025, doi: 10.12720/jait.16.5.751-759.

[34] Z. Hou, X. Li, L. Li, J. Yuan and K. Deng, "An End-to-End Raw Bytes Based Malware Classifier via Self-Attention Residual Convolutional Network," 2022 IEEE 8th International Conference on Computer and Communications (ICCC), Chengdu, China, 2022, pp. 1666-1670, doi: 10.1109/ICCC56324.2022.10065922.

[35] W. Liu, K. Wu, T. Liu, Y. Wang, K. -H. Yap and L. -P. Chau, "ByteNet: Rethinking Multimedia File Fragment Classification Through Visual Perspectives," in IEEE Transactions on Multimedia, vol. 27, pp. 1305-1319, 2025, doi: 10.1109/TMM.2024.3521830.

[36] M. Li, H. Chen, and Z. Cheng, "An attention-guided spatiotemporal graph convolutional network for sleep stage classification," Life, vol. 12, no. 5, Art. no. 622, 2022, doi: 10.3390/life12050622.

[37] T. Sasi, A. H. Lashkari, R. Lu, P. Xiong, and S. Iqbal, "An efficient self attention-based 1D-CNN-LSTM network for IoT attack detection and identification using network traffic," J. Inf. Intell., vol. 3, no. 5, pp. 375–400, 2025, doi: 10.1016/j.jiixd.2024.09.001.

[38] H. Lu, H. Cai, Y. Liang, A. Bianchi and Z. B. Celik, "A Progressive Transformer for Unifying Binary Code Embedding and Knowledge Transfer," 2025 IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER), Montreal, QC, Canada, 2025, pp. 383-393, doi: 10.1109/SANER64311.2025.00043.

[39] S. Aribe Jr., "A hybrid deep learning and forensic approach for robust deepfake detection," Int. J. Adv. Comput. Sci. Appl., vol. 16, no. 10, 2025, doi: 10.14569/IJACSA.2025.0161028.

[40] F. Ullah, A. Alsirhani, M. M. Alshahrani, A. Alomari, H. Naeem, and S. A. Shah, "Explainable malware detection system using transformers-based transfer learning and multi-model visual representation," Sensors, vol. 22, no. 18, Art. no. 6766, 2022, doi: 10.3390/s22186766.

[41] V. Roussev and G. G. Richard III, "Breaking the performance wall: The case for distributed digital forensics," Digit. Invest., 2004.

[42] B. Anderson, D. Quist, J. Neil, C. Storlie, and T. Lane, "Graph-based malware detection using dynamic analysis," J. Comput. Virol., vol. 7, no. 4, pp. 247–258, 2011, doi: 10.1007/s11416-011-0152-x.

[43] B. Kolosnjaji, A. Zarras, G. Webster, and C. Eckert, "Deep learning for classification of malware system call sequences," in AI 2016: Advances in Artificial Intelligence (AI 2016), B. H. Kang and Q. Bai, Eds., vol. 9992, Lecture Notes in Computer Science. Cham, Switzerland: Springer, 2016, pp. 137–149, doi: 10.1007/978-3-319-50127-7_11.

[44] M. Lotfollahi, M. Jafari Siavoshani, R. Shirali Hossein Zade, and M. Saberian, "Deep packet: A novel approach for encrypted traffic

classification using deep learning," Soft Comput., vol. 24, no. 3, pp. 1999–2012, 2020, doi: 10.1007/s00500-019-04030-2.

[45] X. Lin, G. Xiong, G. Gou, Z. Li, J. Shi, and J. Yu, "ET-BERT: A contextualized datagram representation with pre-training transformers for encrypted traffic classification," in Proc. ACM Web Conf. (WWW), Virtual Event, Lyon, France, 2022, pp. 633–642, doi: 10.1145/3485447.3512217.

[46] G. Aceto, D. Ciuonzo, A. Montieri and A. Pescapé, "Mobile Encrypted Traffic Classification Using Deep Learning," 2018 Network Traffic Measurement and Analysis Conference (TMA), Vienna, Austria, 2018, pp. 1-8, doi: 10.23919/TMA.2018.8506558.

[47] F. De Gaspari, D. Hitaj, G. Pagnotta, L. De Carli, and L. V. Mancini, "Reliable detection of compressed and encrypted data," Neural Comput. Appl., vol. 34, pp. 20379–20393, 2022, doi: 10.1007/s00521-022-07586-7.

[48] G. Srivastava, M. P. Singh, P. Kumar, and J. P. Singh, "Internet Traffic Classification: A Survey," p. 611, Jun. 2016, doi: 10.1142/9789814704830_0058.

[49] K. Harinath and G. K. Kumar, "An efficient approach for encrypted traffic classification and intrusion detection using packet transformer encoder and CNN," Int. J. Comput. Exp. Sci. Eng., vol. 11, no. 3, 2025, doi: 10.22399/ijcesen.1377.

[50] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, "Learning and classification of malware behavior," in Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2008), D. Zamboni, Ed. Berlin, Heidelberg: Springer, 2008, vol. 5137, Lecture Notes in Computer Science, pp. 108–125, doi: 10.1007/978-3-540-70542-0_6.

[51] D. Arp, M. Spreitzenbarth, M. Hübner, H. Gascon, and K. Rieck, "DREBIN: Effective and explainable detection of Android malware in your pocket," in Proc. Netw. Distrib. Syst. Secur. Symp. (NDSS), 2014.

[52] E. Gandotra, D. Bansal, and S. Sofat, "Malware analysis and classification: A survey," J. Inf. Secur., vol. 5, pp. 56–64, 2014, doi: 10.4236/jis.2014.52006.

[53] H. S. Anderson, A. Kharkar, B. Filar, D. Evans, and P. Roth, "Learning to evade static PE machine learning malware models via reinforcement learning," arXiv preprint arXiv:1801.08917, 2018.

[54] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," ACM Comput. Surv., vol. 51, no. 5, Art. no. 93, Aug. 2019, doi: 10.1145/3236009.

[55] D. Gunning and D. Aha, "DARPA's explainable artificial intelligence (XAI) program," AI Mag., vol. 40, no. 2, pp. 44–58, 2019, doi: 10.1609/aimag.v40i2.2850.

[56] G. Horsman, "Framework for Reliable Experimental Design (FRED): A research framework to ensure the dependable interpretation of digital data for digital forensics," Comput. Secur., vol. 73, pp. 294–306, 2018, doi: 10.1016/j.cose.2017.11.009.

[57] D. Lillis, B. A. Becker, T. O'Sullivan, and M. Scanlon, "Current challenges and future research areas for digital forensic investigation," arXiv preprint arXiv:1604.03850, 2016.

[58] B. Biggio and F. Roli, "Wild patterns: Ten years after the rise of adversarial machine learning," Pattern Recognit., vol. 84, pp. 317–331, 2018, doi: 10.1016/j.patcog.2018.07.023.

[59] Z. C. Lipton, "The mythos of model interpretability," Commun. ACM, vol. 61, no. 10, pp. 36–43, Oct. 2018, doi: 10.1145/3233231.

[60] F. Doshi-Velez and B. Kim, "Towards a Rigorous Science of Interpretable Machine Learning," arXiv preprint arXiv:1702.08608, 2017.

Estimation of Landslide Hazard Zones Using Deep Learning Based on Diverse Geospatial Data

Kohei Arai, Kengo Oiwane, Hiroshi Okumura
Information Science Dept., Saga University, Saga City, Japan

***Abstract*—Traditional landslide hazard mapping in Japan relies on labor-intensive field surveys, which are slow, costly, and fail to update dynamically amid rising climate-driven disasters like the 2018 Heavy Rain Event, leaving gaps in timely evacuations. This study addresses these challenges by proposing a semantic segmentation framework using ResUNet to fuse Sentinel-2 optical, Sentinel-1 SAR amplitude, DEM-derived Terrain Ruggedness Index (TRI), and JAXA land cover data, tackling class imbalance with BCE + Dice loss and providing probability/uncertainty maps via 4-TTA for robust hazard delineation under adverse weather. The principal aim is to enable operational, weather-robust hazard zone extraction with AUC upto 0.89 (best multimodal configuration), outperforming single-modality baselines (e.g., optical-only AUC 0.74; SAR-only 0.69) through synergistic feature fusion, while highlighting multimodal SAR's edge for cloud-obscured scenarios. Validated on Hiroshima Prefecture data—Japan's highest-risk region with ~32,000 hazard spots—this approach demonstrates pre/post-disaster change detection, but reveals limitations in spatial generalization due to region-specific training.**

Keywords*—SAR; Optical; ResUNet++; land cover classification; Digital Elevation Model; landslide hazard zones; The Heavy Rain Event of July 2018

I. Introduction

In recent years, the incidence of sediment-related disasters in Japan has increased significantly, and the resulting damage has become more severe. According to statistics from the Ministry of Land, Infrastructure, Transport and Tourism (MLIT), the annual average number of sediment-related disasters in Japan was 1,064 from 1995 to 2004, but this figure has increased to 1,523 over the most recent 10-year period (2015 to 2024) [1]. The main factors cited for this increase are the frequent occurrence of record-breaking heavy rainfall caused by linear precipitation zones and typhoons. According to observational data published by the Japan Meteorological Agency (JMA) from approximately 1,300 locations nationwide, the annual average number of occurrences of hourly rainfall exceeding 50 mm was 285 from 1995 to 2004. In contrast, the annual average for the most recent 10-year period (2015 to 2024) was 333, also showing an increasing trend [2]. Thus, a strong correlation is suggested between climate change and the increased risk of landslides and debris flows.

To prevent and mitigate damage from such disasters, national and prefectural governments maintain hazard maps. These hazard maps visualize the risk levels not only for landslides and debris flows but also for other natural disasters like tsunamis, storm surges, and floods. They are used to provide residents with evacuation criteria and information on evacuation sites. However, the selection of landslide hazard zones currently relies heavily on on-site surveys conducted by experts. Consequently, the selection process requires extensive time and cost, making it extremely difficult to comprehensively cover large areas and continuously reflect changes in land use, environmental changes due to climate change, and topographic conditions. Furthermore, cases have been reported where landslides occurred outside designated hazard zones, indicating the potential for hazards to occur beyond these areas [3]. Insufficient update frequency thus risks undermining the reliability of hazard map warning zones. To enhance residents' disaster preparedness awareness, it is essential to provide reliable warning zone information updated at near-real-time.

This study contributes: 1) optimal multimodal input analysis showing AUC gains (e.g., +0.15 from optical to optical+DEM+land cover); 2) probability/uncertainty visualization for operational trust; 3) pre/post-2018 Heavy Rain Event application revealing surface change sensitivity; 4) BCE+Dice loss for imbalance handling. Focused on Hiroshima's granite-prone terrain (31,987 hazard spots), it prioritizes methodological validation over broad generalization, with restrained claims pending cross-region tests.

This study aims to extract dynamically changing landslide hazard zones. To achieve this, we propose a hazard zone extraction method that integrates deep learning models with multiple geospatial datasets. Specifically, we use optical imagery or Synthetic Aperture Radar (SAR) amplitude imagery as the base input data. By combining this with a Digital Elevation Model (DEM) and land cover imagery, we quantitatively compare the improvement in extraction accuracy compared to using a single data type alone. Furthermore, we analyze the reliability of the extraction results by visualizing not only a probability map (which displays the model's confidence) as a heatmap, but also an uncertainty map calculated from the variability of the predictions. Additionally, we apply the proposed model to satellite images before and after The Heavy Rain Event of July 2018 to compare changes in the landslide hazard zones.

As mentioned earlier, the occurrence of landslides and debris flows is not only caused by prolonged heavy rain or torrential downpours but also faces increased risk from natural disasters such as earthquakes and volcanic activity. However, it is also pointed out that besides natural disasters, deforestation and changes in land use alter soil stability, increasing the risk of landslides and debris flows [4]. Illegal logging and clear-cutting often occur in remote mountain areas where humans rarely venture. Consequently, detection is delayed, potentially leading to the occurrence of unknown landslides and debris flows. Given

this background, utilizing satellite imagery suitable for wide-area observation is effective.

The remainder of this study is organized as follows: Section II reviews related work; Section III details the proposed method and the materials used; Section IV presents the results; and Section V offers concluding remarks.

II. Related Research Works and Differences from this Study

A. Related Research Works

Satellite remote sensing has become an indispensable technology for landslide detection and monitoring. Scaioni et al. categorized the applicability of remote sensing technology into three categories: landslide recognition, landslide monitoring, and landslide hazard analysis and prediction, explaining the potential of remote sensing through further advancements [5].

Research on landslide detection is advancing through various approaches. Kanai et al. performed landslide detection from aerial photographs using the deep learning model Mask R-CNN [6]. Qi et al. evaluated the landslide detection accuracy of U-Net and ResUNet models, demonstrating that ResUNet outperforms U-Net in terms of precision, recall, and F1 score [7]. Furthermore, regarding verification focused on versatility, Prakash et al. constructed a convolutional neural network (CNN) training method using composite inventories and proposed an approach applicable to new areas [8]. Liu et al. proposed the Multi-Scale Complex Background Emphasis (MSSCBE) method to address the challenge of landslides being easily confused with complex backgrounds, leading to false detections and missed detections [9].

Research on landslide hazard zone detection using various deep learning models has also been conducted. However, optical imagery alone has limitations in capturing elevation and topographic features. Consequently, research integrating geospatial data is advancing. Liu et al. proposed a semantic segmentation network (FFS-Net) that integrates high-resolution remote sensing images and DEM data [10]. Similarly, Yoshida et al. proposed a decision tree model using explanatory variables derived from optical satellite imagery and DEM [11].

Regarding landslide hazard zones, Ado et al. analyzed articles on machine learning (ML) models used to create landslide hazard maps, reporting a comprehensive review while emphasizing the importance of landslide hazard maps [12]. To mitigate future landslide risks and enhance the applicability of sustainable land use planning, Habumugisha et al. (2022) [13] constructed multiple deep learning models (DNN/LSTM/CNN/RNN) for landslide susceptibility mapping. Similarly, Chen et al. proposed an improved U-Net model addressing the limitation that conventional machine learning models focused solely on individual pixels and failed to consider the surrounding environment (spatial structural information) [14]. By utilizing diverse inputs, including satellite imagery, topographic data, and geological information, they reported improved accuracy and the generation of landslide susceptibility maps.

Several publications provide important insights regarding applications in SAR, and numerous studies have been conducted on disaster detection using SAR data. Mondini et al. published a comprehensive review on SAR imagery for landslide detection, detailing SAR's utility, operational frameworks, and future applications [15]. Furthermore, Mondini et al. used SAR amplitude imagery from the Sentinel-1 satellite to identify landslides by leveraging SAR-specific changes and photo interpretation techniques, aiming for rapid landslide detection [16].

Regarding research utilizing deep learning, Liang et al. proposed a landslide detection method using a morphological model (DLM) for single-polarization SAR images after disasters [17]. Arai et al. proposed an approach to identify non-vegetated areas indicating landslide-prone zones using the Radar Vegetation Index (RVI) derived from Sentinel-1 SAR data and the EfficientNetV2 deep learning model [18].

As a study combining terrain data, Nava et al. proposed an Attention U-Net model utilizing Sentinel-1 satellite SAR amplitude images and slope angles [19]. By utilizing phase information and interferometrically combining phase data from multiple time periods, Interferometric Synthetic Aperture Radar (InSAR) analysis technology can detect displacement. As research applying InSAR technology, Li et al. proposed an MSFD-Net model using displacement images and tilt angles as input data [20].

Numerous studies utilizing InSAR have also been conducted regarding landslide hazard zones. Yuan et al. proposed an approach to extract landslide susceptibility factors and predict landslide susceptibility using InSAR technology by constructing a hybrid model combining a convolutional neural network (CNN) and a recurrent neural network (RNN) strong in handling time-series data [21]. Additionally, Yang et al. proposed an approach to identify potential landslide sites in specific regions by combining InSAR technology with multiple topographic and meteorological datasets [22].

B. Differences from this Study

Much current research focuses primarily on landslide detection and landslide hazard zone extraction after disasters occur. This study aims to extract landslide hazard zones from satellite imagery observed during routine periods, regardless of whether they preceded or followed a disaster. Incorporating changes that increase landslide risks-such as Land Use/Land Cover changes and deforestation—is expected to enhance the reliability of hazard maps and support their revision. Furthermore, to verify its effectiveness in rapidly assessing post-disaster damage and mitigating secondary disaster risks, satellite imagery before and after The Heavy Rain Event of July 2018 is input and compared.

III. Proposed Method and Materials

A. Intensive Study Area

This study focuses on Hiroshima Prefecture in Japan. This region is characterized by the steep terrain of the Chugoku Mountains, presenting an extremely high risk of sediment-related disasters. During The Heavy Rain Event of July 2018, the prefecture suffered extensive damage due to numerous slope failures and debris flows. Table I lists the top five prefectures in Japan with the highest number of designated sediment-related

disaster hazard zones, based on statistics from the Ministry of Land, Infrastructure, Transport and Tourism [23]. Hiroshima Prefecture recorded approximately 9,000 more hazard zones than Shimane Prefecture, which ranked second, making it the region with the highest landslide risk nationwide. Against this backdrop, Hiroshima Prefecture was selected as the test site for verifying the proposed extraction method.

The reason for focusing solely on Hiroshima Prefecture rather than the whole of Japan lies in its distinct geological characteristics [24]. Granite, widely distributed in Hiroshima Prefecture, is particularly prone to weathering into sand and soil (decomposed granite, locally known as *Masa soil*) due to rain and wind, and has been reported to have a high potential for triggering landslides [25,26].

Fig. 1 shows the geographical scope of the study area, depicting Hiroshima Prefecture and its surrounding regions. The optical satellite imagery (Sentinel-2[1]) and SAR satellite imagery (Sentinel-1[2]) used in this analysis were obtained from the Copernicus Browser[3] within the Copernicus Data Space Ecosystem.

TABLE I. TOP 5 PREFECTURES IN JAPAN WITH THE HIGHEST NUMBER OF LANDSLIDE HAZARD SPOTS

Rank	Prefecture	Debris flow	Landslide	Steep Slope Failure	Total
1	Hiroshima	9,964	80	21,943	31,987
2	Shimane	8,120	264	13,912	22,296
3	Yamaguchi	7,532	285	14,431	22,248
4	Hyogo	6,912	286	13,550	20,748
5	Oita	5,125	222	14,293	19,640



Fig. 1. Map of the study area, Hiroshima Prefecture, Japan.

B. Data Used

1) Optical imagery: For optical imagery, satellite data from the European Space Agency (ESA) Sentinel-2 mission were acquired. Sentinel-2 provides multispectral data with a spatial resolution of 10 meters, enabling detailed observation of surface spectral reflectance characteristics. Its high revisit frequency (approximately every 5 days) is advantageous for rapid monitoring, although observations are constrained by cloud cover and nighttime conditions. For this study, true-color images (TCI) synthesized from RGB bands were acquired. To minimize cloud effects, images captured during daylight hours on a clear day, August 8, 2024, were obtained. The datasets were created using QGIS4 by saving images while shifting the location of the target area. Fig. 2 shows a portion of the Sentinel-2 optical image of Hiroshima Prefecture acquired on August 8, 2024, as exported from QGIS.



Fig. 2. Sentinel-2 optical image acquired on August 8, 2024, near Hiroshima City.

2) SAR imagery: For SAR imagery, amplitude (Ground Range Detected, GRD) data from the Sentinel-1 mission operated by the ESA were acquired. Sentinel-1 is an active C-band sensor that transmits microwaves, enabling observation of the Earth's surface regardless of weather conditions or the time of day. However, raw GRD data contain geometric distortions such as radar shadows and speckle noise, particularly in mountainous areas of the study region. To eliminate these distortions and ensure geometric consistency with other geospatial data, preprocessing was performed using the ESA Sentinel Application Platform (SNAP[5]).

The processing steps are as follows:

- Apply Orbit File: Correction of orbital errors using precise orbit ephemerides.
- Thermal Noise Removal: Reduction of additive thermal noise from the sensor.
- Calibration: Conversion of digital pixel values to physical backscatter coefficients.
- Range Doppler Terrain Correction: Orthorectified using Copernicus 30m global DEM to precisely align SAR images with geographic coordinates.

In this study, images were acquired on August 7 and August 19, 2024, to align with the timing of optical data acquisition. Fig. 3 shows a portion of the Sentinel-1 SAR image acquired over Hiroshima Prefecture on August 7, 2024, as exported from QGIS.

[1] https://sentinels.copernicus.eu/copernicus/sentinel-2
[2] https://sentinels.copernicus.eu/copernicus/sentinel-1
[3] https://dataspace.copernicus.eu/ecosystem/services/copernicus-browser
[4] https://qgis.org/
[5] https://step.esa.int/main/toolboxes/snap/



Fig. 3. Sentinel-1 SAR image acquired on August 7, 2024, near Hiroshima City.

3) Digital Elevation Model (DEM)

- Topographic features are key factors influencing landslide hazard potential. We utilized the 5-meter mesh DEM provided by the Geospatial Information Authority of Japan (GSI). To quantify the local terrain complexity correlated with slope instability, we calculated the Terrain Ruggedness Index (TRI). The TRI for a central pixel is defined as the square root of the sum of the squares of the elevation differences between the central pixel and its eight adjacent pixels, as shown in Eq. (1) [27]:

$$TRI = \sqrt{\sum_{i=1}^{8}(X_{center} - X_i)^2} \quad (1)$$

where, X_{Center} represents the elevation of the central pixel, and Xi represents the elevation of the adjacent pixels. TRI was calculated using Raster Terrain Analysis tool available in QGIS. This index functions as a feature that quantitatively represents surface ruggedness.

In this study, 5-meter mesh DEMs (5A) were obtained from the GSI Base Map Information[6] download service. The creation date is June 20, 2025. Under the assumption that this date is close to the acquisition dates of the optical and SAR images, the data were used accordingly. Fig. 4 shows a portion of DEM converted to TRI format for Hiroshima Prefecture, as exported from QGIS.

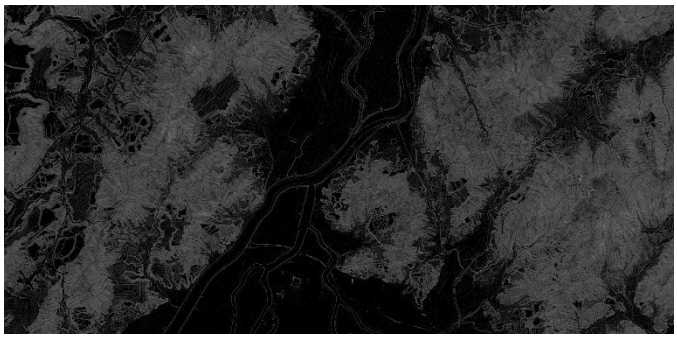


Fig. 4. TRI image derived from the DEM for the area around Hiroshima City.

[6] https://service.gsi.go.jp/kiban/app/

[7] https://www.eorc.jaxa.jp/ALOS/jp/dataset/lulc/lulc_v2504_j.htm

[8] https://disaportal.gsi.go.jp/index.html

4) Land Cover Classification Map

- This study utilized the high-resolution Land Use/Land Cover (LULC[7]) map provided by the Japan Aerospace Exploration Agency (JAXA). This dataset features a 10-meter spatial resolution and classifies the land surface into 15 categories (e.g., water bodies, grasslands, and bare land). In the proposed model, these 15 classification categories were converted into one-hot-encoded vectors and treated as independent feature channels.
- These data from 2024 were released in April 2025. As with the DEM, it was assumed that the observation dates of the optical and SAR images were close to the release date. Fig. 5 shows a portion of the LULC map for Hiroshima Prefecture, as exported from QGIS.

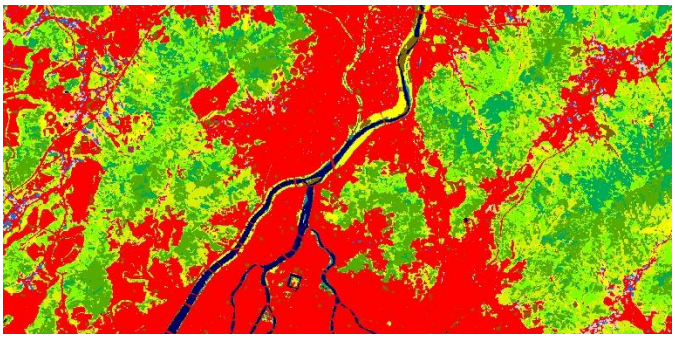


Fig. 5. Land cover classification image for the area around Hiroshima City.

5) Landslide Hazard Zone

- The ground truth data labels used for model training and evaluation were obtained from the "Landslide Hazard Areas (Steep Slope Collapse)" raster data[8] (2023 edition) provided by the Geospatial Information Authority of Japan. These areas are designated under the Sediment-Related Disaster Prevention Act as zones where steep slopes (30 degrees or more) are prone to collapse and pose a danger to residents. Fig. 6 shows a portion of the landslide hazard zones within Hiroshima Prefecture, as exported from QGIS.
- To create correct labels, we performed a binarization process. Designated areas were extracted, and the "landslide danger zone" (yellow area) and the "special landslide danger zone" (red area) were defined as "danger zone" (value = 1) and merged into a single positive class. All other areas were defined as "nondanger zone" (value = 0).

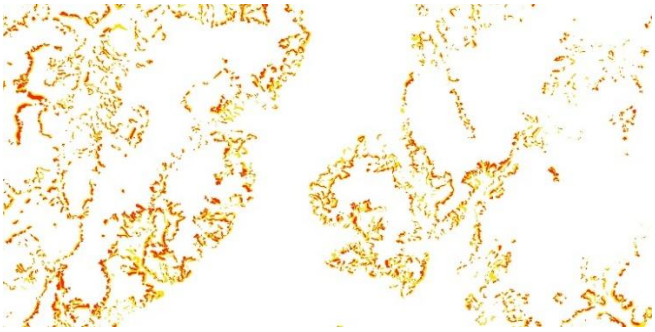


Fig. 6. Image of landslide hazard warning zones around Hiroshima City.

C. Proposed Methodology and Model Architecture

- The methodology consists of data preparation, dataset generation, model training using various input combinations, and comparative evaluation.
- For deep learning dataset generation, the aligned data were cropped into 256×256-pixel patches. These patches were subsequently divided into training, validation, and test datasets. Note that during dataset creation, to capture only the topographic structure and geological features within Hiroshima Prefecture, the model was comprehensively trained by changing the image positions in all areas each time. Therefore, areas included in training may also be present within test cases (though not as identical images). This is because the research aims to identify models that maximize accuracy when combining geospatial data and to establish specialized methods for each prefecture; the input of new areas during testing is not anticipated.
- For the model implementation phase, ResUnet++ architecture was adopted as the architecture. To evaluate the contribution of various environmental factors to landslide detection accuracy, multiple models were implemented using different combinations of input data (e.g., optical data only, SAR + DEM, etc.). Each model was trained individually, and the extraction accuracy was evaluated and compared using the test dataset.
- Finally, based on quantitative results, we conducted a comparative analysis between the highest-performing model and a baseline model that used only sensor data. This analysis involved acquiring sensor data before and after the disaster to detect and compare changes in the estimated landslide hazard zones.
- We employed ResUNet++, a semantic segmentation method, as a model for extracting landslide hazard zones [28]. This ResUNet++ is an architecture that improves upon the standard U-Net and ResUNet. In this study, extraction is performed using multiple channels. Therefore, it is necessary to evaluate the importance of channels while performing detailed extraction of complex boundary regions. For these reasons, ResUNet++ was adopted.

D. Experimental Setup

- Table II shows the model types and input channel counts. The dataset consisted of 2,390 pairs in total, which were randomly split into three subsets: 1,673 pairs (70%) for training, 478 pairs (20%) for validation, and 239 pairs (10%) for testing. Furthermore, for all models, using the same seed value, vertical/horizontal flipping and scaling were applied as identical affine transformations during training.
- We designed eight experimental models with varying input combinations, ranging from single-sensor inputs to fused multimodal inputs. The input image size is standardized to 256×256 pixels for all models. The number of input channels was varied based on a configuration combining optical data (3 channels), DEM (1 channel), SAR (1 channel), and one-hot encoded land cover data (15 channels) in the channel direction.
- Table III shows the main hyperparameters used for model construction. The loss function employs a combination of BCE (binary cross-entropy) and Dice loss, termed "BCE Dice Loss". This combines BCE, which determines positive/negative on a per-pixel basis and tends to stabilize gradients, with Dice loss, which evaluates the overlap between the predicted mask and the ground truth mask and is effective for class-imbalanced images.
- The total loss (L) is defined as follows [see Eq. (2)]:

$$\Lambda = \langle \cdot \Lambda_{BXE} + ® \cdot \Lambda_{\Delta IXE} \quad (2)$$

where, L_{BCE} represents the BCE loss and L_{DICE} represents the Dice loss. In this study, to unify pixel-level accuracy and region-based overlap, equal weights (α=0.5, β=0.5) were assigned to both components.

TABLE II. EXPERIMENTAL MODELS AND INPUT CHANNEL CONFIGURATIONS

Model	Number of Input Channels
(a) Optical	3
(b) Optical + DEM(TRI)	4
(c) Optical + Land Cover	18
(d) Optical + DEM(TRI) + Land Cover	19
(e) SAR	1
(f) SAR + DEM(TRI)	2
(g) SAR + Land Cover	16
(h) SAR + DEM(TRI) + Land Cover	17

TABLE III. HYPERPARAMETERS

Parameter	Value
Image Size	256×256
Batch Size	8
Loss Function	BCE + Dice
Optimizer	Nadam
Learning Rate Scheduler	ReduceLROnPlateau
Initial Learning Rate	1.0×10^{-4}
Patience (LR Decay)	10 epochs
Decay Factor	0.2
Early Stopping	30 epochs

E. Evaluation Metrics

1) Model Performance Metrics

- To quantitatively evaluate the extraction accuracy of the proposed model, we adopted precision, recall, and F1 score as accuracy metrics. These metrics are calculated

based on the components of the confusion matrix: true positives (TP), false positives (FP), and false negatives (FN) [see Eq. (3) to Eq. (5)]:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4)$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (5)$$

- In addition to comparing accuracy using a single threshold, we created Precision-Recall (PR) curves showing the relationship between precision and recall as the prediction probability threshold was continuously varied from 0.0 to 1.0. We also calculated the area under the curve (AUC). This enabled the comparison of overall model performance independent of any specific threshold. The AUC is commonly used as a performance evaluation metric in landslide susceptibility mapping [12].

2) Visualization of Probability and Uncertainty Maps

- To quantify the model's confidence for each pixel in the segmentation results, probability maps and uncertainty maps were output and visualized. While typical segmentation results output binary values (0 or 1) per pixel, this segmentation model also outputs pixel-level prediction probabilities during testing. Visualizing these alongside the binary results enabled the evaluation of prediction reliability. For calculating both maps, we adopted an inference method based on four-pattern Test-Time Augmentation (4-TTA). Specifically, we applied four transformation patterns to the input image: no transformation (p0), horizontal flip (p1), vertical flip (p2), and horizontal plus vertical flip (p3). Inference is performed on each transformed image. The average (μ) of the four obtained probability values is calculated as the probability map, and the standard deviation (σ) is calculated as the uncertainty map. The probability map indicates the expected probability for each pixel, while the uncertainty map shows that larger standard deviations indicate greater instability in the model's prediction. Fig. 7 shows an overview of the estimation method. p0 to p3 represent the probability values for each transformed image. This method not only reduces prediction instability associated with single-image inference but also visualizes areas of model uncertainty, making it useful for decision-making in practical applications.

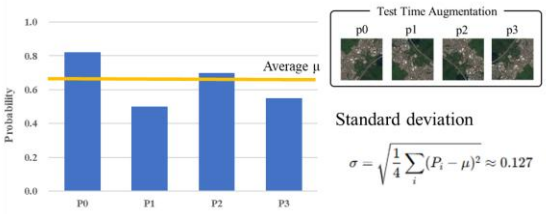


Fig. 7. Overview of inference method using 4-TTA.

IV. Results

A. Quantitative Evaluation of Extraction Accuracy

1) Performance of optical-based models: Table IV shows the evaluation results for each model based on optical images. Note that the values in Table IV were obtained by searching for a threshold where precision and recall were balanced and then running the model using that threshold. The optical-only model (a) achieved P=0.68, R=0.67, F1=0.67, and AUC=0.74. In contrast, the model incorporating DEM (b) showed significant accuracy improvements: P=0.80, R=0.79, F1=0.79, and AUC=0.88. On the other hand, the model incorporating land cover (c) showed P=0.71, R=0.71, F1=0.71, and AUC=0.79, indicating less improvement than the DEM model. The model incorporating all data (d) achieved the best results across all metrics: P=0.81, R=0.82, F1=0.81, and AUC=0.89, demonstrating the most balanced performance among the metrics.

Fig. 8 shows the PR curves for each optical-based model. In all comparisons, adding input data shifted the entire PR curve upward and to the right, indicating improved accuracy. Adding a DEM contributed significantly to a substantial improvement in AUC (+0.14, from 0.74 to 0.88). In contrast, adding land cover results in a smaller improvement in AUC (+0.05, from 0.74 to 0.79). This can be attributed to the fact that many of the ground surface elements represented by land cover were already captured by the spectral information in the optical images. Nevertheless, since both precision and recall improved, land cover is considered to supplement surface features to a certain extent for areas that are difficult to distinguish using optical images alone.

TABLE IV. Results of each Model based on Optical Images

Model	Precision	Recall	F1 score	AUC
(a)	0.68	0.67	0.67	0.74
(b)	0.80	0.79	0.79	0.88
(c)	0.71	0.71	0.71	0.79
(d)	0.81	0.82	0.81	0.89

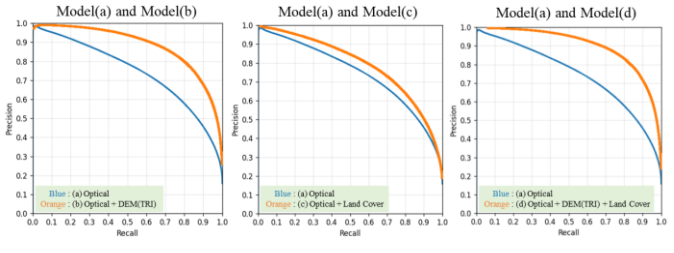


Fig. 8. Comparison of precision–recall (PR) curves for optical-based models with different input configurations.

2) Performance of SAR-based models: Table V shows the evaluation results for each model based on SAR images. As in the case of optical imagery, the values in Table V were obtained by searching for a threshold at which precision and recall were balanced and then applying that threshold. For the SAR-only

model (e), P=0.63, R=0.62, F1=0.63, and AUC=0.69, respectively. In contrast, the model incorporating a DEM (f) showed a substantial improvement, with P=0.75, R=0.73, F1=0.74, and AUC=0.82. The model incorporating land cover (g) also achieved P=0.74, R=0.73, F1=0.73, and AUC=0.81, indicating an improvement comparable to model (f). The model using all inputs (h) achieved the best performance, with P=0.81, R=0.80, F1=0.81, and AUC=0.89, matching the accuracy of the best optical-based model (d). This model also exhibited the most favorable balance among the metrics.

Fig. 9 shows the PR curves for the SAR-based models. As with the optical imagery case, adding input data led to improved performance across the entire curve. In addition, for SAR-based models, unlike the optical imagery case, the inclusion of land-cover data contributed to a substantial improvement in accuracy comparable to that achieved by adding DEM. This was likely because discriminating surface types solely based on microwave backscattering intensity is challenging, as diverse land covers can exhibit similar backscatter signatures depending on surface roughness and dielectric properties. Therefore, the inclusion of land cover data provided essential categorical context, resolving these ambiguities and substantially complementing the SAR-derived spatial features.

TABLE V. RESULTS OF EACH MODEL BASED ON SAR IMAGES

Model	Precision	Recall	F1 score	AUC
(e)	0.63	0.62	0.63	0.69
(f)	0.75	0.73	0.74	0.82
(g)	0.74	0.73	0.73	0.81
(h)	0.81	0.80	0.81	0.89

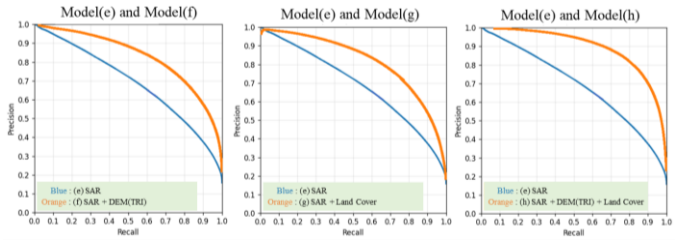


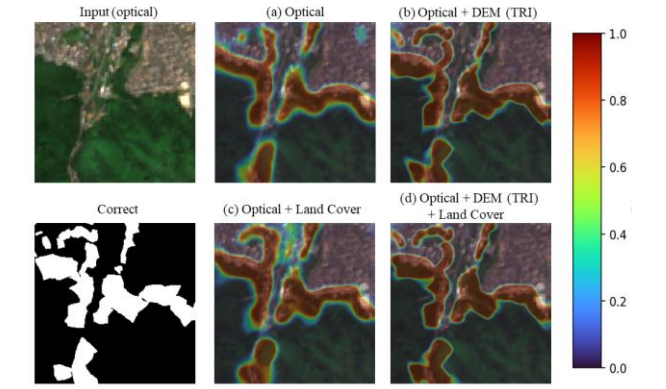
Fig. 9. Comparison of precision–recall (PR) curves for SAR-based models with different input configurations.

B. Visualization of Probability Maps and Uncertainty Maps

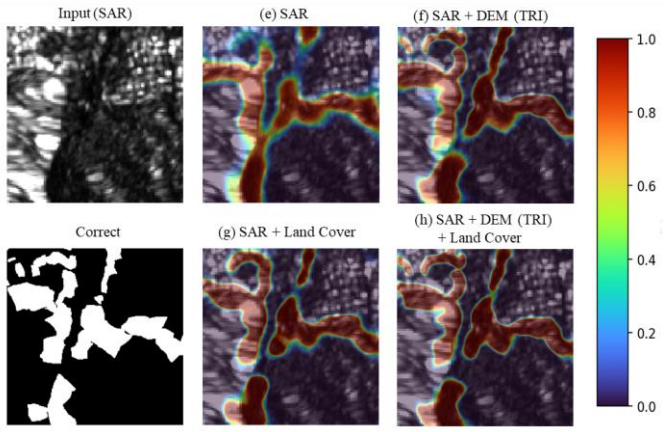
The output results of the probability maps and uncertainty maps are shown in Fig. 10. Comparing the results across all models, a consistent tendency was observed in both categories: high confidence in the central parts of the predicted areas and increased uncertainty along the boundaries. In the probability maps, the expansion of the red regions indicated higher reliability, whereas in the uncertainty maps, a reduction in edge width indicated less ambiguity in the prediction.

For the single-sensor models (a) and (e), many locations exhibited low confidence even in the central parts of the extracted areas, and the area of high-uncertainty edges along the boundaries became larger, indicating greater model instability. In contrast, the models incorporating DEM (b) and (f) showed a substantial improvement in confidence. In the models incorporating land cover (c) and (g), false detections decreased, particularly in urban areas and in locations away from slopes where sediment-related disasters are less likely to occur. In the models incorporating both DEM and land cover (d) and (h), high confidence in the central parts was maintained while the area of high-uncertainty edges near the boundaries became smaller.

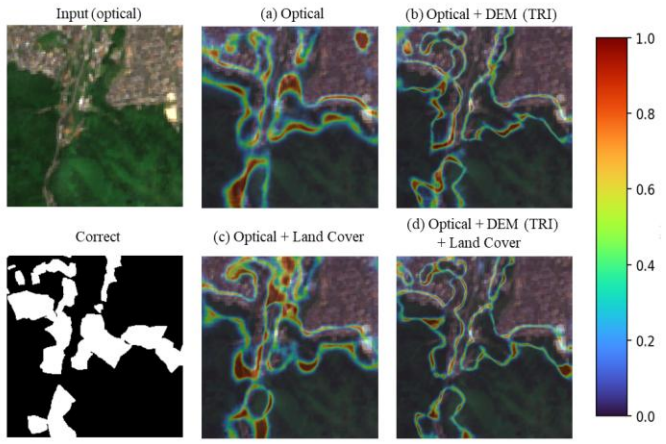
- Thus, presenting deep learning outputs not only as binary results (positive/negative) but also simultaneously as probability maps and uncertainty maps can facilitate an intuitive understanding of the situation for specific areas and can be utilized for rapid and accurate decision-making.



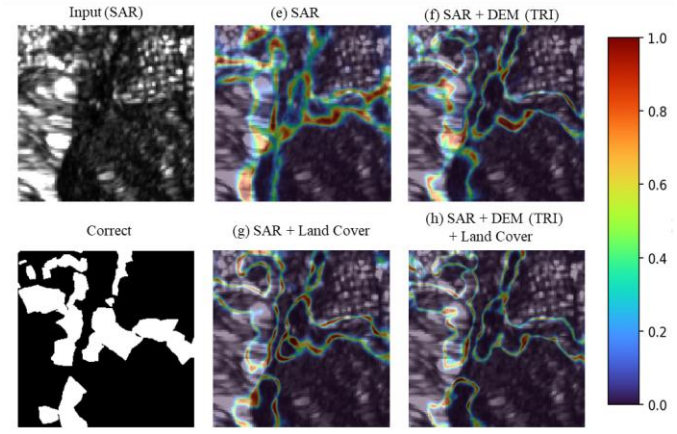
(a) Probability map of the optical-based model.



(b) Probability map of the SAR-based model.



(c) Uncertainty map of the optical-based model.



(d) Uncertainty map of the SAR-based model.

Fig. 10. Probability maps and uncertainty maps.

C. Comparison of Estimation Results before and after Disaster Events

1) Overview of the disaster: Model (d) (optical + DEM + land cover) achieved the highest accuracy for extracting sediment-disaster hazard zones. In this section, we evaluated the practical applicability of this model by applying it to optical images acquired before and after an actual disaster event and analyzing the resulting changes in the probability maps. For comparison, the same analysis was also conducted using the baseline model (a) (optical only). In this study, Nakahata, Yasuura Town, Kure City, Hiroshima Prefecture, which suffered severe damage during The Heavy Rain Event of July 2018 in western Japan, was selected as the validation area. In this district, heavy rainfall triggered sediment-related disasters in multiple mountainous locations, causing road disruptions to settlements and extensive building damage across the area. Fig. 11 shows a photograph of the damage in the nearby Ichihara district[9]. In addition, Fig. 12 shows the optical satellite images acquired for the study area (Nakahata district) before the disaster (June 1, 2018) and after the disaster (July 16, 2018).



Fig. 11. Ichihara district, Kure City.

[9] https://www.city.kure.lg.jp/site/kure-city-reconstruction-memorial-museum/kure-city-reconstruction-memorial-museum-higaijokyo.html

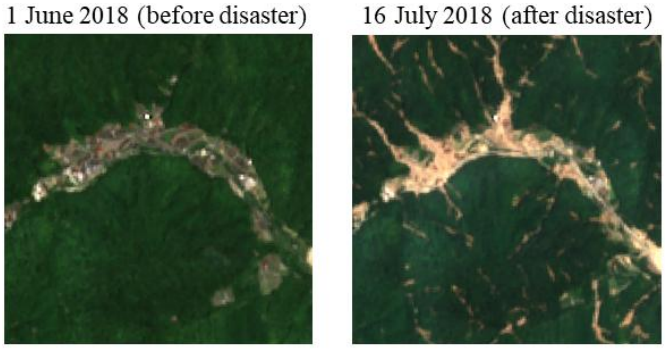


Fig. 12. Optical satellite images of Nakahata, Yasuura Town.

2) Analysis of probability map change: Fig. 13 shows the probability maps produced before and after the disaster, and Fig. 14 shows the difference between the before-disaster and after-disaster probability maps. In the after-disaster probability map (July 16, 2018), both models reflected changes in surface conditions caused by heavy rainfall, with the predicted areas concentrated mainly near the boundary between mountainous and lowland areas. However, the spatial distributions differed clearly between the models.

In model (a), the high-probability regions were spread over a wide area, and sporadic responses were also observed on surrounding slopes. Because the input relied only on spectral information, distinguishing the presence or absence of sediment-related disasters became difficult; as a result, the model responded to apparent changes unrelated to the disaster, leading to unstable extraction. In contrast, in model (d), the high-probability regions were concentrated mainly along boundary areas consistent with the disaster-affected zone, and unnecessary extraction on surrounding slopes is suppressed. This can be interpreted as the model not treating all apparent after-rainfall changes as disasters but instead narrowing the candidate areas by considering topography and land cover information. The ability to output only high-confidence regions is beneficial for practical operations such as rescue activities and risk assessment of secondary disasters.

- Because the DEM and land cover data used in this study are static datasets updated every few years, the same pre-disaster datasets were used for both the before-disaster and after-disaster analyses in this experiment. However, because actual landslides can cause substantial topographic changes, achieving more dynamic and accurate extraction requires topographic and land cover data that reflect conditions immediately after the disaster. In recent years, studies have advanced in directly estimating up-to-date land cover and geospatial data from satellite imagery using deep learning [29,30]. By integrating these techniques into the proposed method, further improvements in the timeliness and accuracy of landslide hazard extraction are expected.
- Prob/uncertainty maps show high center confidence, edge uncertainty reduced by fusion. Pre/post-2018: multimodal pinpoints changes vs. optical scatter. No baselines (e.g., DeepLabV3+, RF on slope/cover) or disjoint tests yet.

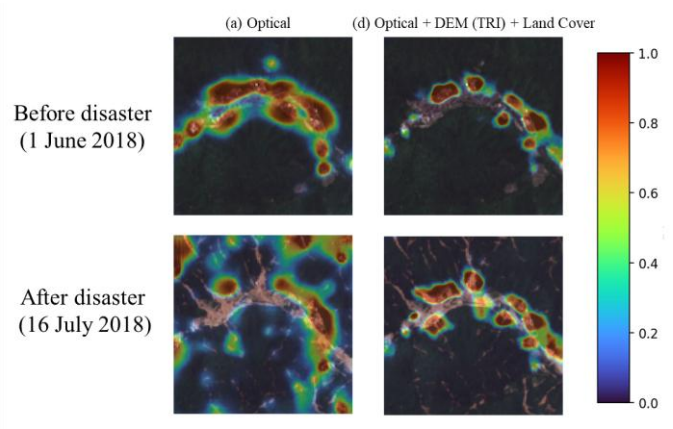


Fig. 13. Comparison of probability maps before and after the disaster.

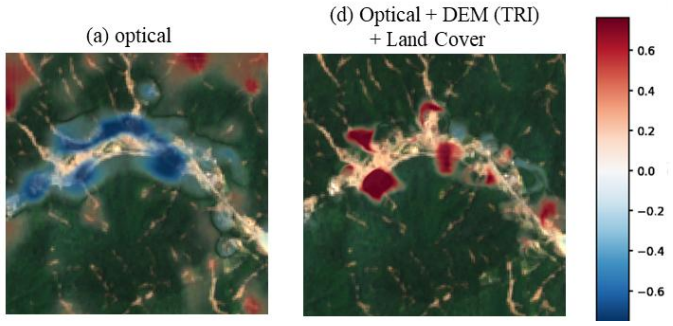


Fig. 14. Differential probability maps (before and after the disaster) visualizing the changes in the probability map.

V. Conclusion

We proposed a deep learning model for extracting sediment disaster hazard zones by integrating satellite imagery and geospatial data. Models were developed by combining diverse information sources, including optical imagery, SAR imagery, DEM, and land cover maps. The accuracy comparison results confirmed that integrating geospatial data substantially improved extraction performance compared with models using a single data source. The best-performing configuration was "optical/SAR + DEM + land cover", achieving an AUC of 0.89. The effectiveness of DEM and land cover was also demonstrated. Because optical and SAR imagery had complementary characteristics that compensated for each other's limitations, obtaining high extraction accuracy with both image types suggested that the proposed approach is highly generalizable and is not constrained by weather conditions or day–night differences. In addition, the case study of The Heavy Rain Event of July 2018 indicated that inputting before-disaster and after-disaster optical imagery may enable the detection of changes in hazard zones.

Multimodal fusion boosts robustness, SAR aiding clouds, but Hiroshima training limits broader claims—no cross-prefecture validation. Predicts susceptibility (pre-event), post-disturbance (change detection); preprocessing detailed, but normalization/seasonality assumed stable.

In conclusion, this advances hazard mapping via ResUNet multimodal fusion (SAR/optical/DEM/LULC, BCE+Dice, uncertainty viz), achieving F1=0.81/AUC=0.89 on Hiroshima data, enabling dynamic updates vs. surveys. Scientific value: quantifies input contributions, operational potential for weather-hit areas. As for the limitations, region-specific (no spatial generalization); random split risks leakage; lacks ML baselines, modern nets (Attention U-Net, DeepLabV3+), CIs; speculative robustness sans scenario tests.

Future Research Works

We plan to develop a framework to generate land cover and DEM products from satellite imagery. Although static DEM and land cover maps are currently used, integrating a mechanism that directly derives these geospatial datasets from up-to-date satellite imagery would enable more dynamic extraction of sediment disaster hazard zones by immediately reflecting changes. In addition, we also plan to incorporate new inputs, such as geological conditions within the target area and real-time rainfall distribution, which are expected to advance the model toward more reliable extraction. Furthermore, we plan to use higher-resolution optical and SAR data to achieve further improvements in accuracy. Namely, spatially disjoint validation/cross-events have to be investigated with baselines (RF/XGBoost, DeepLab); dynamic DEM/LULC generation; geology/rain inputs; higher-res data.

Acknowledgment

The author would like to thank Prof. Dr. Osamu Fukuda of Saga University for his valuable comments and suggestions.

Generative AI Usage Statement

The authors declare that the generative AI tool Gemini 3.0 was used solely for language support and proofreading purposes, including improving grammar, wording, and overall clarity of the manuscript.

The development of scientific ideas, methodologies, analyses, results, and conclusions was carried out entirely by the human authors. All AI-assisted text was carefully reviewed and edited by the authors, who take full responsibility for the accuracy, originality, and integrity of the manuscript.

References

[1] Ministry of Land, Infrastructure, Transport and Tourism, Japan. Landslide disasters in 2024 exceeded the long-term average (since statistics began)—Number of landslide disasters in 2024 released (Press Release, 27 January 2025). Available: https://www.mlit.go.jp/report/press/sabo02_hh_000156.html (accessed on January 14, 2026).

[2] Japan Meteorological Agency. Historical changes in extreme events such as heavy rainfall and extremely hot days (Climate Change Portal). Available: https://www.data.jma.go.jp/cpdinfo/extreme/extreme_p.html (accessed on January 14, 2026).

[3] T. Nishikawa, "Characteristics of sediment runoff occurring at locations not designated as sediment disaster warning areas during the July 2018 heavy rain," Suiri Kagaku (Water Science), vol. 64, no. 2, pp. 43–59, 2020.

[4] H. Saito, W. Murakami, H. Daimaru, and T. Oguchi, "Effect of forest clear-cutting on landslide occurrences: Analysis of rainfall thresholds at Mt. Ichifusa, Japan," Geomorphology, vol. 276, pp. 1–7, 2017.

[5] M. Scaioni, L. Longoni, V. Melillo, and M. Papini, "Remote sensing for landslide investigations: An overview of recent achievements and perspectives," Remote Sensing, vol. 6, no. 10, pp. 9600–9652, 2014.

[6] K. Kanai, S. Kubo, T. Yamane, and P.-J. Chun, "A method for automatic detection of landslide areas from aerial photographs using Mask R-CNN," Artificial Intelligence and Data Science, vol. 2, pp. 223–231, 2021.

[7] W. Qi, M. Wei, W. Yang, C. Xu, and C. Ma, "Automatic mapping of landslides by the ResU-Net," Remote Sensing, vol. 12, no. 15, p. 2487, 2020.

[8] N. Prakash, A. Manconi, and S. Loew, "A new strategy to map landslides with a generalized convolutional neural network," Scientific Reports, vol. 11, p. 9722, 2021.

[9] X. Liu, L. Xu, and J. Zhang, "Landslide detection with Mask R-CNN using complex background enhancement based on multi-scale samples," Geomatics, Natural Hazards and Risk, vol. 15, no. 1, p. 2300823, 2024.

[10] X. Liu et al., "Feature-fusion segmentation network for landslide detection using high-resolution remote sensing images and digital elevation model data," IEEE Transactions on Geoscience and Remote Sensing, vol. 61, p. 4500314, 2023.

[11] K. T. Yoshida, D. Nakayama, and H. Matsuyama, "Construction of landslide detection model using optical satellite images and DEMs by machine learning with multi-instance decision tree model," Journal of the Remote Sensing Society of Japan, vol. 44, no. 4, pp. 317–334, 2025.

[12] M. Ado et al., "Landslide susceptibility mapping using machine learning: A literature survey," Remote Sensing, vol. 14, no. 13, p. 3029, 2022.

[13] J. M. Habumugisha et al., "Landslide susceptibility mapping with deep learning algorithms," Sustainability, vol. 14, no. 3, p. 1734, 2022.

[14] Y. Chen et al., "Mapping post-earthquake landslide susceptibility: A U-Net like approach," Remote Sensing, vol. 12, no. 17, p. 2767, 2020.

[15] A. C. Mondini et al., "Landslide failure detection and mapping using synthetic aperture radar: Past, present and future," Earth-Science Reviews, vol. 216, p. 103574, 2021.

[16] A. C. Mondini et al., "Sentinel-1 SAR amplitude imagery for rapid landslide detection," Remote Sensing, vol. 11, no. 7, p. 760, 2019.

[17] R. Liang et al., "Landslide mapping from post-event single-temporal polarimetric SAR image by a deep learning method exploiting a morphological model," Remote Sensing of Environment, vol. 328, p. 114904, 2025.

[18] K. Arai, Y. Nakaoka, and H. Okumura, "Method for landslide area detection with RVI data which indicates base soil areas changed from vegetated areas," Remote Sensing, vol. 17, p. 628, 2025.

[19] L. Nava, K. Bhuyan, S. R. Meena, O. Monserrat, and F. Catani, "Rapid mapping of landslides on SAR data by attention U-Net," Remote Sensing, vol. 14, p. 1449, 2022.

[20] N. Li et al., "A deep-learning-based algorithm for landslide detection over wide areas using InSAR images considering topographic features," Sensors, vol. 24, no. 14, p. 4583, 2024.

[21] R. Yuan and J. Chen, "A hybrid deep learning method for landslide susceptibility analysis with the application of InSAR data," Natural Hazards, vol. 114, no. 2, pp. 1393–1426, 2022.

[22] X. Yang, D. Chen, Y. Dong, Y. Xue, and K. Qin, "Identification of potential landslide in Jianzha county based on InSAR and deep learning," Scientific Reports, vol. 14, p. 21346, 2024.

[23] Ministry of Land, Infrastructure, Transport and Tourism (MLIT), Water and Disaster Management Bureau, Sabo Department. Sediment Disaster Hazard Locations by Prefecture. Available: https://www.mlit.go.jp/river/sabo/link20.htm (accessed on January 14, 2026).

[24] K. Araiba, T. Nozaki, B.-P. Jeong, and Y. Fukumoto, "Distribution of landslides and their geological characteristics in Japan—Analysis report based on the data of nation-wide designated landslide-threatened areas—," J. Japan Landslide Soc., vol. 44, no. 5, pp. 318–323, 2008.

[25] T. Watakabe and Y. Matsushi, "Lithological controls on hydrological processes that trigger shallow landslides: Observations from granite and hornfels hillslopes in Hiroshima, Japan," Catena, vol. 180, pp. 55–68, 2019.

[26] S. Doshida, "The feature of the landslides by the Hiroshima heavy rain on August 20, 2014," J. Jpn. Soc. Saf. Eng., vol. 57, no. 1, pp. 22–27, 2018.

[27] S. J. Riley, S. D. DeGloria, and R. Elliot, "A terrain ruggedness index that quantifies topographic heterogeneity," Intermountain J. Sciences, vol. 5, no. 1–4, pp. 23–27, 1999.

[28] D. Jha et al., "ResUNet++: An advanced architecture for medical image segmentation," in Proc. 2019 IEEE Int. Symp. Multimedia (ISM), pp. 225–230.

[29] C. F. Brown et al., "Dynamic World, near real-time global 10 m land use land cover mapping," Sci. Data, vol. 9, p. 251, 2022.

[30] J. Gao, J. Liu, and S. Ji, "A general deep learning based framework for 3D reconstruction from multi-view stereo satellite images," ISPRS J. Photogramm. Remote Sens., vol. 195, pp. 446–461, 2023.

AUTHOR'S PROFILE

Kohei Arai, He received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science and Technology of the University of Tokyo from April 1974 to December 1978, and was also with the National Space Development Agency of Japan from January 1979 to March 1990. From 1985 to 1987, he was with the Canada Centre for Remote Sensing as a Post-Doctoral Fellow of the National Science and Engineering Research Council of Canada. He was a professor of Saga University from 1990 to 2017. Also, he was a councilor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology from 1998 to 2000 and was a councilor of Saga University for 2002 and 2003. He also was an executive councilor for the Remote Sensing Society of Japan for 2003 to 2005. He was the Vice Chairman of the Scientific Committee was an Award Committee member of ICSU/COSPAR during. He is a Science Council of Japan Special Member (COSPAR Committee) since 2012. He is an Adjunct Professor of Nishi-Kyushu University and Kurume Institute of Technology as well as Prishtina International University. He wrote 134 books and published 759 journal papers as well as 591 conference papers. He received 66 of awards, including the ICSU/COSPAR Vikram Sarabhai Medal in 2016, and the Science award of the Ministry of Mister of Education of Japan in 2015. He is now Editor-in-Chief of IJACSA. http://teagis.ip.is.saga-u.ac.jp/index.html

A Hybrid Framework Integrating GNN-LSTM-CNN to Map the Impact of MSME User Behavior on Digital Transformation

Jani Kusanti[1]*, Erni Widiastuti[2], Bintara Sura Priambada[3],
Ramadhian Agus Triono Sudalyo[4], Masdava Aviv Masyayissa[5], Rizki Adhi Pratama[6]
Faculty of Electrical Information Engineering, Universitas Surakarta, Indonesia [1, 4, 5, 6]
Faculty of Economics, Universitas Surakarta, Indonesia[2]
Faculty of Law, Universitas Surakarta, Indonesia[3]

***Abstract*—MSMEs need a recommendation system that simultaneously captures the evolution of user intent over time and the relationship structure between entities (users, products, sessions, categories, and security events). The problem with this research is that LSTM excels in sequence, but its performance drops on a rare timeline, a general situation in MSME logs. GNN is strong for cross-entity relationships, but it does not explicitly model temporal dynamics. The gap arises because many pipelines still separate design signals, temporal behavior, and session security, reducing explainability and long-term reliability. Our contribution proposes a calibrated and security-aware hybrid that integrates CNN, heterogeneous GNN with reverse edges, and LSTM for behavioral sequences. Multitask-trained models (BCE for purchase links and λ· BCE for session risk) with L2 regularization and post-practice calibration, chronological data sharing prevents leakage. The goal is to design and evaluate CNN-plus CNN-enhanced GNN-LSTM hybrids to improve the accuracy of recommendations and reduce risk. The results on partner MSME data: ROC-AUC 0.965 (val)/0.946 (test), PR-AP 0.943/0.910; risk ROC-AUC 0.984, PR-AP 0.982, surpassing a CNN-BiLSTM baseline (0.93/0.91). Brier scores 0.161 (links) and 0.176 (risk) enable safer personalization. Going forward, we are focusing on per-segment calibration with ECE/MCE reporting, compute efficiency, multimodal expansion, ablation, and explainability (GNNExplainer, CNN saliency), as well as online retraining and drift monitoring to maintain production performance.**

***Keywords*—CNN; e-commerce; hybrid model; LSTM; MSME; user behavior prediction**

I. Introduction

The competitiveness of MSMEs is increasingly supported by digital transformation, which is understood not only as technology absorption, but as a twin engine of technology assimilation and business model innovation that updates strategy and performance [1]. Digital transformation requires a proactive attitude to build capabilities in data processing, skills, processes, and governance so that digital tools translate into durable value [2],[3],[4],[5]. Digital tools in the form of e-commerce certainly require consequences, especially with risk and resilience: AI can reduce SME business risks and strengthen the resilience of service supply under disruption [6],[7], even as companies have to keep pace with shifting customer expectations and align digital marketing practices accordingly [8],[9].

On the demand side, user experience is crucial. E-commerce displays in the form of visual presentations include aesthetics, accessibility, readability, and interactive features or AR, clearly shape engagement and purchase intent [10],[11],[12]. With an increasingly multimodal data e-commerce view, feature matching captures cross-modal relevance under production constraints, complemented by a lightweight and interpretable relevance architecture and scalable NLP flows for opinion mining [13],[14],[15],[16]. The use of chatbots influences trust through interactivity and perceived humanity, conditioning adoption and further outcomes in digital commerce [17],[18].

User security and privacy are inseparable in e-commerce. The lack of security in e-commerce causes MSMEs to routinely face phishing, bots, card testing, account abuse, and the threat of multimodal fake reviews that erode trust and directly shape user behavior [19],[20],[21],[22]. At the same time, the well-documented privacy-personalization paradox underscores the need for recommenders who pair accuracy with responsible data practices and transparent decision-making [23],[24],[25]. However, despite the rapid advances in AI for e-commerce and the increasing adoption of social and omnichannel tools by SMEs, most production pipelines still optimize conversions while degrading security to a separate layer that ignores how design cues and session-level risk together affect behavior over time [26],[27],[28].

In this study, conducted in [26] [6], the LSTM model is effective at capturing sequential user dynamics central to the recommendation system, enabling MSMEs to infer needs and predict purchase intent for a more personalized experience and higher conversions. However, LSTMs require heavy training and computing costs, large parameter footprints, and prolonged optimization cycle constraints, which clash with the limits of MSME resources [29],[30],[6],[31]. They also inherit the opacity of deep learning, complicating diagnosis and refinement of strategies [15],[29]. The large amount of model data in the training data has an impact on performance; data inadequacy can worsen when MSME logs are sparse, noisy, or incomplete, especially for sequence-dependent architectures such as LSTM [31].

The contribution of this research is to improve the accuracy of modeling the impact of MSME user behavior on the digital transformation agenda. To close the gaps, we designed a unified

*Corresponding author.

architecture that integrates three key sources of information: design and user experience signals extracted from product imagery with CNN, user–product–session relational contexts along with security incidents modeled through GNN-based heterogeneous graphs, as well as long-term behavioral dynamics represented by LSTM. This integration allows the system to increase the relevance of recommendations and maintain security resiliency, so that user trust is maintained.

II. Previous Research Study

Digital transformation in MSMEs is not only about technology absorption but also about how to build business model innovations that reshape performance and strategy [1]. Transformation requires a proactive attitude that demands capabilities, data, skills, processes, and adaptive governance to maintain the conditions of value creation that are often highlighted in Indonesian MSMEs and under the pressure of Industry 4.0 [2],[13],[32]. Meeting these demands requires models that connect design, behavior, and networks. Visual design display, accessibility, and readability are the drivers of purchase intent; CNN can be used to extract these signals from product imagery [11]. Long-term shifts in intent and engagement require LSTMs to robustly model daily or weekly sequences under shocks and seasonality [6]. Meanwhile, purchasing journeys, sessions, categories, and even security events form a relational substrate that GNN can capture, in line with the reality of platformization in the MSME ecosystem [28],[27],[33],[37]. CNN, LSTM, and GNN answered marketing calls to track evolving expectations with measurement-driven, disciplined practices [9]. Multimodal fusion research shows clear advantages of combining images, text, and behavioral signals [13],[14] and emphasizes the need for a lightweight, interpretable relevance model at production scale [15], while review-centric NLP still struggles with domain shifts and multimodal grounding [16].

Security in transactions, privacy, and trust are the basis when MSMEs carry out digital transformation through e-commerce. E-commerce is constantly exposed to phishing, bots, card testing, and account abuse that erodes trust and conversions [21], with multimodal fake reviews further contaminating reputation signals [22]. Paradoxically, personalization requires transparency, user control, and calibrated decisions [23], echoing the approach of preserving trust in adjacent domains [24]. Predictive analytics in MSMEs has progressed from data-mining studies of purchasing patterns to LSTM models (strengthened by exploratory analysis) that improve online sales forecasting [34]. In parallel, GNNs bring structure-aware learning for interaction, supply, and cyberattack networks, supporting anomaly and threat detection to reduce operational risk, though scalability can be challenging for resource-constrained MSMEs [35],[6],[36].

The research gap is that most MSME-oriented recommenders still handle design cues, temporal behavior, and security risk in isolation, which limits explainability and undermines long-horizon reliability. This research contribution is a hybrid CNN-LSTM-GNN architecture that encodes product-design quality from images, models users' temporal dynamics, and propagates information across a heterogeneous graph of users, products, sessions, and categories, including security events. The approach is expressly designed to map and make explainable how user behavior both drives and reflects digital transformation in MSME e-commerce, closing the integration and interpretability gap while remaining suitable for production deployment. while reporting rigorous ranking and reliability metrics ROC-AUC, PR-AP, and probability calibration (Brier/ECE) to enable trustworthy and cost-conscious deployment.

III. Proposed Research Methodology

The methodology for this study is illustrated in Fig. 1.

A. LSTM

LSTM dynamics (single layer), the equation used is shown in Eq. (1):

$$h_t = LSTM(x_t, h_{t-1}),\ \hat{y} = \sigma(Wh_T + b) \quad (1)$$

LSTM in Eq. (1) variable summary (concise): At time t, $\boldsymbol{x_t}$ is the feature vector (e.g., daily user aggregates such as event count, dwell time, discounts), typically $x_t \in \mathbb{R}^F$ (or $\mathbb{R}^{TxF}$ for full sequences). The previous hidden state $\boldsymbol{h_{t-1}} \boldsymbol{\epsilon}\ \mathbb{R}^{\boldsymbol{H}}$ carries memory. After ingesting x_t, the LSTM outputs $h_t \in \mathbb{R}^H$ (with an internal cell state ct, omitted here). Over a sequence of length $\boldsymbol{T}$, $\boldsymbol{h_T}$ summarizes the entire history. The final prediction is computed by a linear map and activation: $\hat{y} = \sigma(Wh_T + b)$, with $W \in \mathbb{R}^{1xH}, b \in \mathbb{R}$ for binary (sigmoid), or $W \in \mathbb{R}^{KxH}, b \in \mathbb{R}^K$, for multi-class (softmax). In short, x_t provides the current signal, h_{t-1} the memory, h_T the sequence summary, and (W, b, σ) convert it to calibrated probabilities.

B. GDBT

The GBDT additive learner (m trees) equation is shown in Eq. (2):

$$F_m(x) = F_{m-1}(x) + vh_m(x),\ \ m = 1, \dots, M \quad (2)$$

GBDT in Eq. (2) is explained as follows: In this boosting formulation, x denotes the feature vector for a single sample. The algorithm proceeds over iterations $m = 1, \dots, M$, where M is the total number of weak learners. At each step, the current ensemble prediction $F_{m-1}(x)$ is updated by adding a new weak learner $h_m(x)$, for example, a small regression tree trained on the residuals/gradients at step m. The update is scaled by the learning rate $v(0 < v \leq 1)$; smaller v makes changes more conservative and typically requires more iterations. The result is the new ensemble prediction $F_m(x)$ after incorporating h_m at iteration m.

C. CNN

CNN encoder (feature map pooled to embedding) is shown in Eq. (3):

$$z^{img} = Pool(\phi_{CNN}(I)) \quad (3)$$

CNN in Eq. (3) is explained as follows: I is the input product image (RGB tensor $HxWx3$). $\phi_{CNN}(.)$ is a convolutional backbone that converts I into a feature map $F \epsilon \mathbb{R}^{H'xW'xC}$ capturing color, texture, layout, and readability. $Pool(.)$ (global average/max/adaptive) compresses F into a fixed-length vector. $z^{img} \epsilon \mathbb{R}^{d_{img}}$ is the resulting image embedding a compact design/UX representation fed to downstream models.

Fusion (weighted blending of branch scores) is shown in Eq. (4):

$$\hat{p} = w_{CNN}\hat{p}_{CNN} + w_{LSTM}\hat{p}_{LSTM} + w_{gdbt}\hat{p}_{gdbt},\ w_k \geq 0, \sum_k w_k = 1 \quad (4)$$

Eq. (4) is explained as follows: here, $\hat{p}$ denotes the final ensemble prediction (e.g., purchase probability or session-risk score), obtained by blending three branch outputs: $\hat{p}_{CNN}$ from the image/design stream, $\hat{p}_{LSTM}$ from the temporal-behavior stream, and $\hat{p}_{GDBT}$ from the tabular/exogenous Gradient Boosting stream. The blend uses non-negative weights $w_{CNN}, w_{LSTM}, w_{GDBT}$ that sum to one, ensuring a convex combination and thus a valid probability. The optional multi-task objective (purchase & risk) is shown in Eq. (5):

$$\mathcal{L} = BCE(\hat{p}_{link}, y_{link}) + \lambda BCE(\hat{p}_{risk}, y_{risk}) + \beta||\theta||_2^2 \quad (5)$$

Eq. (5) is explained as follows: here, $\mathcal{L}$ denotes the total training loss the model minimizes. It has three parts: 1) a binary cross-entropy term for purchase link prediction, $BCE(\hat{p}_{link}, y_{link})$, which measures how well the predicted purchase probability $\hat{p}_{link}$ matches the ground-truth label $y_{link} \epsilon \{0,1\}$; 2) a session-risk binary cross-entropy term, $\lambda BCE(\hat{p}_{risk}, y_{risk})$, scaled by $\lambda \geq 0$ to control its influence, by comparing the predicted risk $\hat{p}_{risk}$ to $y_{risk} \epsilon \{0,1\}$; and 3) an L_2 regularization penalty, $\beta||\theta||_2^2$, where θ are all trainable weights (e.g., from the CNN/LSTM/GNN). The coefficient β controls weight decay to curb overfitting by discouraging excessively large parameters.

D. Evaluation

To evaluate, ROC-AUC is shown in Eq. (6), PR-AP in Eq. (7), Brier score in Eq. (8), and ECE in Eq. (9):

$$\text{Po}\chi\text{–AYX: area under } TPR(t) vs FPR(t) \quad (6)$$

$$\Pi\text{P–A}\Pi\text{: } AP = \int_0^1 P(R)dR \quad (7)$$

Precision@K/Recall@K: on top-K ranked pairs

$$\text{Brier score: } \frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - y_i)^2 \quad (8)$$

$$\text{EXE (calibration): } \sum_b \frac{n_b}{N}|acc(b) - conf(b)| \quad (9)$$

Decision policy (thresholding) is shown in Eq. (10):

$$\hat{y} = 1\{\hat{p} \geq r\},\quad r = argmax\ \mathbb{E}[gain - cost_{FP} - friction_{risk}] \quad (10)$$

TPR(t), in Eq. (6), is explained as follows: true-positive rate at threshold t. FPR(t): false-positive rate at threshold t. Eq. (7) is explained as follows: here, P(R): precision as a function of recall R. Brier in Eq. (8) is explained as follows: score N: number of evaluated examples. $\hat{p}_i$: predicted probability, for example i. $y_i \epsilon \{0,1\}$: ground-truth label, for example i. ECE (Expected Calibration Error) in Eq. (9) is explained as follows b: probability bin (e.g., 0.0–0.1, …, 0.9–1.0). $\boldsymbol{n_b}$: number of examples in bin b. acc(b): empirical accuracy in bin b. conf(b): average predicted probability (confidence) in bin b. Decision policy (thresholding) in Eq. (10) is explained as follows: $\hat{y}=1\{\hat{p}\geq r\}$: predicted class is 1 if probability $\hat{p}$ exceeds threshold r. $r = \arg\max \mathbb{E}[gain - cost_{FP} - friction_{risk}]$: choose the threshold r that maximizes expected net utility, balancing conversion gain, false-positive cost $cost_{FP}$, and user-experience friction from risk controls.

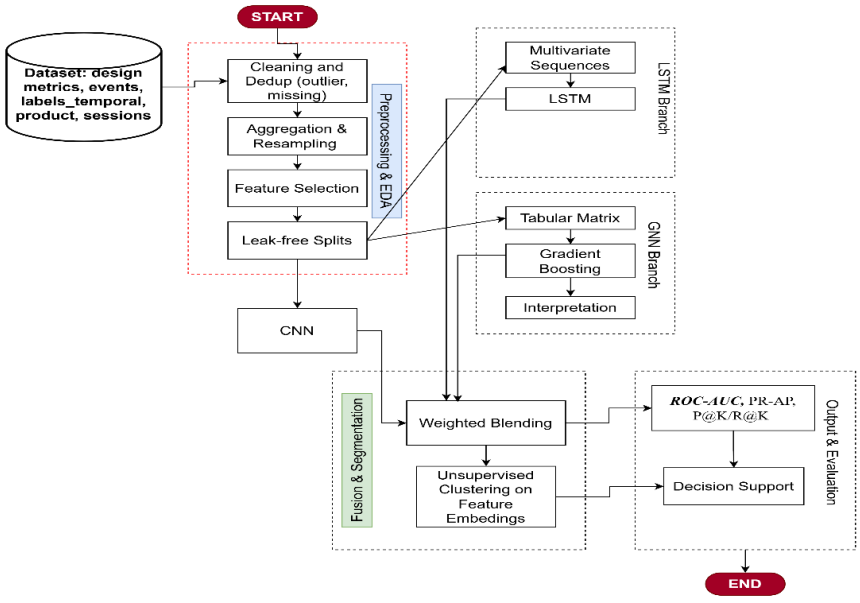


Fig. 1. Proposed research methodology.

IV. Experimental Results and Analysis

The data that has been obtained is stored in the datasets. All data is processed by preprocessing. The next process is carried out by reverse edge, which is meant by reverse edge as follows: Original edge: ('user', 'purchase', 'product') means that the user buys the product. Reverse edge: (product, purchase_rev, user)

means product purchased by the user (inverse direction of information).

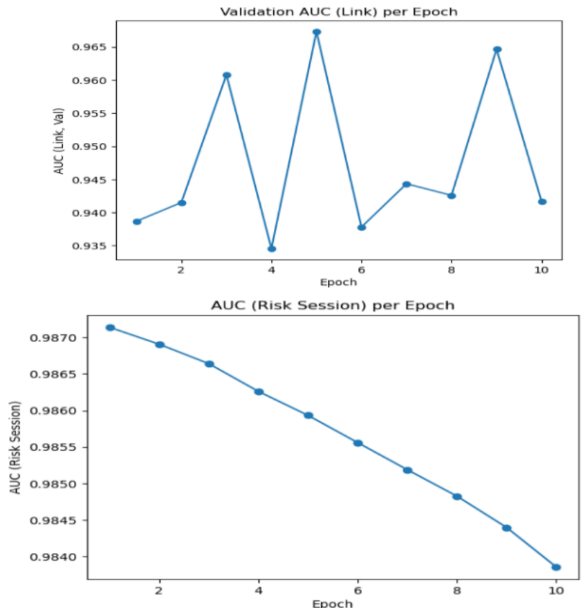


Fig. 2. Tracking and plotting results.

In Fig. 2, Link AUC remains consistently high (~0.94–0.97) with minor saw-tooth fluctuations typical of epoch-wise negative resampling or small validation sets, yielding a stable ≈0.95+ ranking quality for user–product pairs. The security head also stays very strong (risk AUC ~0.984–0.987) with a slight dip; together with a lower BCE, this points to improved calibration. Any small ranking drift is likely negligible and attributable to class imbalance, mild label noise, or train–evaluation distribution differences.

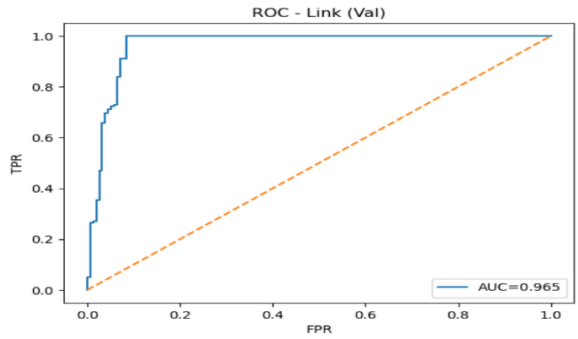


Fig. 3. ROC results - link (Val).

In the ROC–link (Val) curve in Fig. 3, AUC = 0.965 indicates excellent ranking. The curve hugs the top-left, achieving high TPR at low FPR. True buyer pairs are pushed to the top with a few false ranks. Probabilistically, AUC 0.965 ≈ 96.5% chance of a positive score above a negative, leaving ample flexibility to set thresholds for either high-recall or high-precision operation. In Fig. 4, the ROC curve climbs sharply at low FPR and hugs the top-left, indicating strong separation. AUC = 0.946 means the true pair scores higher in ~94.6% of positive–negative pairs. The small drop from validation (AUC ≈ 0.965) signals good generalization without overfitting. Practically, this affords flexible thresholding—low false positives with high TPR—and, alongside PR-AUC/Precision@K, confirms the model reliably prioritizes high-propensity user–product pairs.

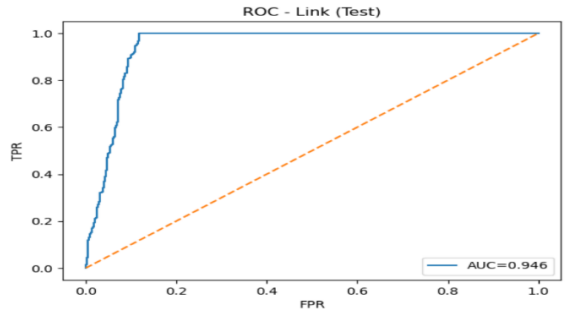


Fig. 4. Hasil ROC - link (Test).

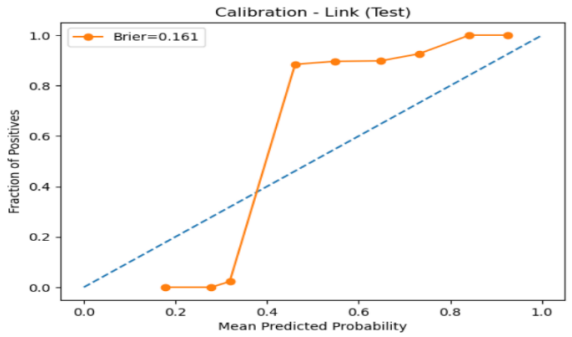


Fig. 5. Calibration - link (Test).

In Fig. 5, the curve shows mild over-confidence at low scores (~0.15–0.30), clear under-confidence in the mid range (~0.35–0.45), and persistent under-confidence at high scores (>0.5–~0.95); predictions near ~0.95–1.0 are very reliable. Brier = 0.161 indicates reasonably good calibration on imbalanced data, with most room for improvement in the middle band. Given the strong ROC/PR, ranking is already solid; light per-segment recalibration (Platt, isotonic, or beta) will make probabilities more proportional and safer for MSME personalization.

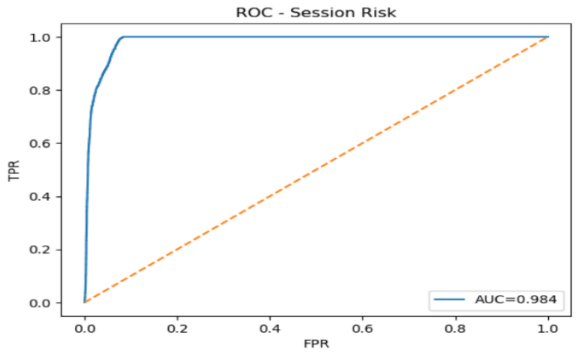


Fig. 6. ROC - session risk.

In Fig. 6, the ROC curve rises steeply at low FPR and hugs the top-left (AUC = 0.984, ≈98.4% pairwise separability). This reflects the security-aware heterogeneous graph, where severity, threat type, and outcomes propagate over session→user→sec_event (with reverse edges). Operationally, sustaining TPR ≥ 0.90 at FPR ≤ 0.05 supports risk-adaptive

friction: higher thresholds to block, lower selective thresholds for CAPTCHA/2FA on borderline cases yielding accurate early warnings at low error cost and stronger security without harming legitimate UX.

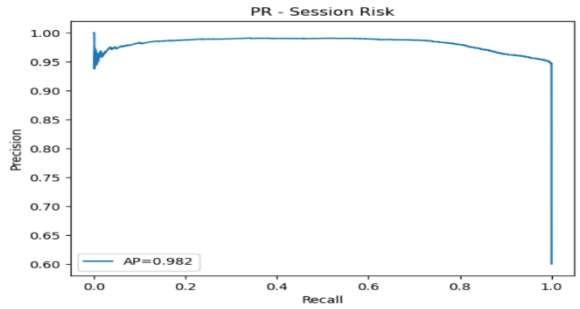


Fig. 7. PR - session risk.

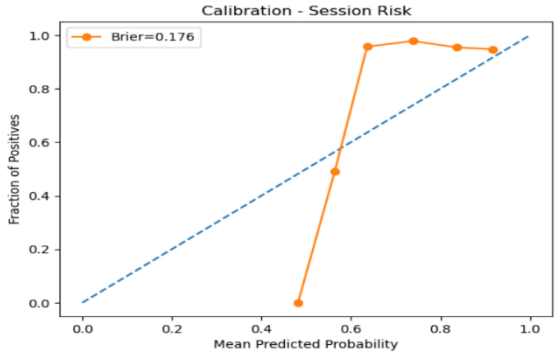


Fig. 8. Calibration - session risk.

In Fig. 7, precision stays ~0.95–1.0 across a wide recall range (AP = 0.982), indicating few false positives and near-perfect ranking. The curve is flat >0.95 through mid–high recall, dropping only near recall ≈1.0 as borderline cases enter. Operate at recall 0.7–0.9 (precision ≥0.95) to enable risk-adaptive friction (selective 2FA/CAPTCHA) with minimal user burden. Results are consistent with the security-aware heterogeneous graph, where severity/type/outcome signals propagate via session–user–sec_event relations (with reverse edges) to deliver accurate early warnings at low cost. The reliability curve in Fig. 8 shows: 1) local over-confidence at ~0.50–0.58 (below the y=x line), 2) real under-confidence at ~0.60–0.70 (curve beyond y=x), and 3) systematic under-confidence in the high range of ~0.70–0.95, while at ~0.95–1.0 the prediction is highly trusted. A Brier score of 0.176 indicates a fairly good calibration in the imbalanced scenario, but needs improvement in the medium probability (~0.55–0.65). Overall, the model excels in the rankings (high PR/ROC), but tends to underestimate the risk at medium–high scores.

In Table I, the hybrid model (CNN + LSTM + heterogeneous GNN with reverse edges) clearly outperforms the CNN–BiLSTM baseline: purchase ROC-AUC rises from 0.93/0.91 (val/test) to 0.965/0.946 with PR-AP 0.943/0.910; the security head reaches ROC-AUC 0.984 and PR-AP 0.982. Calibration is tighter (Brier 0.161 link; 0.176 risk), enabling easier thresholding (e.g., P@K, risk gating) with lower error costs. Gains stem from richer relational context, bidirectional message passing, and multi-task training. Remaining limits: mild mid-range under-confidence and sensitivity to graph/label quality.

TABLE I. COMPARISON OF CNN–BILSTM BASELINE VS. HYBRID CNN+LSTM+GNN ON PARTNER MSME DATA

Aspect	CNN–BiLSTM (Baseline)	Hybrid CNN+LSTM+GNN (Proposed)
ROC-AUC (Purchase)	0.93 (val) / 0.91 (test)	0.965 (val) / 0.946 (test)
PR-AP (Purchase)	—	0.943 (val) / 0.910 (test)
ROC-AUC (Session Risk)	—	0.984
PR-AP (Session Risk)	—	0.982
Calibration (Brier)	— (often less stable for sequence-only models)	0.161 (link) / 0.176 (risk)
Relational context modeling	Limited to temporal user sequences	Yes (user–product–session–category–sec_event) via GNN
Bidirectional message passing	—	Yes (reverse edges; stabilizes embeddings, helps cold/near-cold)
Safe personalization (risk gating)	Limited	Enabled: calibrated thresholding & policy control
Multi-task objective (link + risk)	No	Yes (regularizes representations, reduces conflict)
Key limitations	Sensitive to sparse/noisy sequences; limited explainability	Mild mid-probability under-confidence; performance bounded by graph quality

V. CONCLUSION

This research shows that the hybrid architecture of CNN+LSTM+GNN can provide safe personalization for MSMEs. For the purchase stage, the proposed model achieves a ROC-AUC of 0.965 (val) / 0.946 (test) and a PR-AP of 0.943 / 0.910, exceeding the CNN-BiLSTM baseline (0.93 / 0.91). For session risk detection, performance was close to perfect (ROC-AUC 0.984, PR-AP 0.982). The predicted probability is adequately calibrated (Brier 0.161 for links; 0.176 for risk), making it suitable for operational thresholds. These gains are driven by multi-tasking learning and two-way messaging that exploits relational contexts across users, products, sessions, categories, and security events that generate more accurate recommendations without sacrificing user security or trust. It can be concluded that the CNN+LSTM+GNN hybrid model can be used as an accurate, safe, and measurable recommendation in MSME e-commerce.

Looking ahead, we will pursue multimodal expansion integrating text (titles/reviews), payment/logistics signals, and campaign indicators to enrich decision context, alongside per-segment calibration and cost–benefit evaluation to ensure reliable and cost-effective deployment.

REFERENCES

[1] C. S. de Mattos, G. Pellegrini, G. Hagelaar, and W. Dolfsma, Systematic literature review on technological transformation in SMEs: a transformation encompassing technology assimilation and business model innovation, vol. 74, no. 2. Springer International Publishing, 2024. doi: 10.1007/s11301-023-00327-7.

[2] D. Daryono, "Proactive Digital Transformation Imperative (PDTI): The Role of Capabilities Strategic Digital Business in MSMEs," 2025. doi: 10.1051/e3sconf/202560905010.

[3] H. S. Ikhwan, "Performance reconfiguration in Indonesian MSMEs: Digital transformation, emerging skills, and organizational health," J.

Infrastructure, Policy Dev., vol. 8, no. 3, 2024, doi: 10.24294/jipd.v8i3.3101.

[4] M. Alfarizi, T. Widiastuti, and Ngatindriatun, "Exploration of Technological Challenges and Public Economic Trends Phenomenon in the Sustainable Performance of Indonesian Digital MSMEs on Industrial Era 4.0," J. Ind. Integr. Manag., vol. 09, no. 01, pp. 65–96, Jan. 2023, doi: 10.1142/S2424862223500045.

[5] S. S. Wulandari, "Digital Proficiency and Entrepreneurial Mindset for SME Success through Market Savvy and Tech Literacy," APTISI Trans. Technopreneursh., vol. 7, no. 1, pp. 26–36, 2025, doi: 10.34306/att.v7i1.527.

[6] N. Drydakis, "Artificial Intelligence and Reduced SMEs' Business Risks. A Dynamic Capabilities Analysis During the COVID-19 Pandemic," Inf. Syst. Front., vol. 24, no. 4, pp. 1223–1247, 2022, doi: 10.1007/s10796-022-10249-6.

[7] S. I. Zaman, S. A. Khan, S. Qabool, and H. Gupta, "How digitalization in banking improve service supply chain resilience of e-commerce sector? a technological adoption model approach," Oper. Manag. Res., vol. 16, no. 2, pp. 904–930, 2023, doi: 10.1007/s12063-022-00341-0.

[8] S. Pratap, S. K. Jauhar, Y. Daultani, and S. K. Paul, "Benchmarking sustainable E-commerce enterprises based on evolving customer expectations amidst COVID-19 pandemic," Bus. Strateg. Environ., vol. 32, no. 1, pp. 736–752, 2023, doi: 10.1002/bse.3172.

[9] M. Cioppi, I. Curina, B. Francioni, and E. Savelli, Digital transformation and marketing: a systematic and thematic literature review, vol. 2023, no. 2. Springer International Publishing, 2023. doi: 10.1007/s43039-023-00067-2.

[10] C. Guo and X. Zhang, "The impact of AR online shopping experience on customer purchase intention: An empirical study based on the TAM model," PLoS One, vol. 19, no. 8 August, pp. 1–20, 2024, doi: 10.1371/journal.pone.0309468.

[11] M. W. Iqbal, K. Shinan, S. R. S. Rafique, A. Alourani, M. U. Ashraf, and N. Z. A. Rahim, "Usability and optimization of online apps in user's contex," PeerJ Comput. Sci., vol. 10, pp. 1–27, 2024, doi: 10.7717/PEERJ-CS.2561.

[12] R. Wolniak, K. Stecuła, and B. Aydın, "Digital Transformation of Grocery In-Store Shopping-Scanners, Artificial Intelligence, Augmented Reality and Beyond: A Review," Foods, vol. 13, no. 18, pp. 1–34, 2024, doi: 10.3390/foods13182948.

[13] E. Daskalakis et al., "Applications of Fusion Techniques in E-Commerce Environments: A Literature Review," Sensors, vol. 22, no. 11, 2022, doi: 10.3390/s22113998.

[14] N. Zareena, "State-of-the-Art in Multi-faceted Feature Matching for E-Commerce: A Comprehensive Analysis," Inform., vol. 48, no. 12, pp. 97–106, 2024, doi: 10.31449/inf.v48i12.6007.

[15] Z. Lin, "Deep Bag-of-Words Model: An Efficient and Interpretable Relevance Architecture for Chinese E-Commerce," 2024. doi: 10.1145/3637528.3671559.

[16] N. Malik and M. Bilal, "Natural language processing for analyzing online customer reviews: a survey, taxonomy, and open research challenges," PeerJ Comput. Sci., vol. 10, pp. 1–38, 2024, doi: 10.7717/PEERJ-CS.2203.

[17] Y. Ding and M. Najaf, "Interactivity, humanness, and trust: a psychological approach to AI chatbot adoption in e-commerce," BMC Psychol., vol. 12, no. 1, 2024, doi: 10.1186/s40359-024-02083-z.

[18] K. K. Nirala, N. K. Singh, and V. S. Purani, A survey on providing customer and public administration based services using AI: chatbot, vol. 81, no. 16. Springer US, 2022. doi: 10.1007/s11042-021-11458-y.

[19] S. Saeed, S. A. Altamimi, N. A. Alkayyal, E. Alshehri, and D. A. Alabbad, "Digital Transformation and Cybersecurity Challenges for Businesses Resilience: Issues and Recommendations," Sensors, vol. 23, no. 15, pp. 1–20, 2023, doi: 10.3390/s23156666.

[20] L. Zhou, "Enhancing Phishing Detection in Semantic Web Systems Using Optimized Deep Learning Models," Int. J. Semant. Web Inf. Syst., vol. 20, no. 1, 2024, doi: 10.4018/IJSWIS.361772.

[21] X. Liu et al., "Cyber security threats: A never-ending challenge for e-commerce," Front. Psychol., vol. 13, no. October, pp. 1–15, 2022, doi: 10.3389/fpsyg.2022.927398.

[22] C. Liu, X. He, and L. Yi, "Determinants of multimodal fake review generation in China's E-commerce platforms," Sci. Rep., vol. 14, no. 1, pp. 1–28, 2024, doi: 10.1038/s41598-024-59236-8.

[23] A. I. Canhoto, B. J. Keegan, and M. Ryzhikh, "Snakes and Ladders: Unpacking the Personalisation-Privacy Paradox in the Context of AI-Enabled Personalisation in the Physical Retail Environment," Inf. Syst. Front., vol. 26, no. 3, pp. 1005–1024, 2024, doi: 10.1007/s10796-023-10369-7.

[24] P. Ruotsalainen, B. Blobel, and S. Pohjolainen, "Privacy and Trust in eHealth: A Fuzzy Linguistic Solution for Calculating the Merit of Service," J. Pers. Med., vol. 12, no. 5, 2022, doi: 10.3390/jpm12050657.

[25] Y. K. Dwivedi et al., Exploring the Darkverse: A Multi-Perspective Analysis of the Negative Societal Impacts of the Metaverse, vol. 25, no. 5. Springer US, 2023. doi: 10.1007/s10796-023-10400-x.

[26] R. E. Bawack, S. F. Wamba, K. D. A. Carillo, and S. Akter, Artificial intelligence in E-Commerce: a bibliometric study and literature review, vol. 32, no. 1. Springer Berlin Heidelberg, 2022. doi: 10.1007/s12525-022-00537-z.

[27] W. Hussain and J. M. Merigo, "Onsite/offsite social commerce adoption for SMEs using fuzzy linguistic decision making in complex framework," J. Ambient Intell. Humaniz. Comput., vol. 14, no. 9, pp. 12875–12894, 2023, doi: 10.1007/s12652-022-04157-5.

[28] G. Nikoletos et al., "Digital marketplaces in European research landscape: A systematic literature review," Open Res. Eur., vol. 4, 2024, doi: 10.12688/openreseurope.18657.1.

[29] Z. Su, H. Yang, and J. Ai, FPLV: Enhancing recommender systems with fuzzy preference, vector similarity, and user community for rating prediction, vol. 18, no. 8 August. 2023. doi: 10.1371/journal.pone.0290622.

[30] N. Tyagi and B. Bhushan, "Demystifying the Role of Natural Language Processing (NLP) in Smart City Applications: Background, Motivation, Recent Advances, and Future Research Directions," Wirel. Pers. Commun., vol. 130, no. 2, pp. 857–908, 2023, doi: 10.1007/s11277-023-10312-8.

[31] X. Lu, K. Wijayaratna, Y. Huang, and A. Qiu, "AI-Enabled Opportunities and Transformation Challenges for SMEs in the Post-pandemic Era: A Review and Research Agenda," Front. Public Heal., vol. 10, no. April, pp. 1–11, 2022, doi: 10.3389/fpubh.2022.885067.

[32] M. Alfarizi, "Exploration of Technological Challenges and Public Economic Trends Phenomenon in the Sustainable Performance of Indonesian Digital MSMEs on Industrial Era 4.0," J. Ind. Integr. Manag., vol. 9, no. 1, pp. 65–96, 2024, doi: 10.1142/S2424862223500045.

[33] R. Misra, R. Mahajan, N. Singh, S. Khorana, and N. P. Rana, "Factors impacting behavioural intentions to adopt the electronic marketplace: findings from small businesses in India," Electron. Mark., vol. 32, no. 3, pp. 1639–1660, 2022, doi: 10.1007/s12525-022-00578-4.

[34] P. Bellini, L. A. I. Palesi, P. Nesi, and G. Pantaleo, "Multi Clustering Recommendation System for Fashion Retail," Multimed. Tools Appl., vol. 82, no. 7, pp. 9989–10016, 2023, doi: 10.1007/s11042-021-11837-5.

[35] E. Widiastuti, J. Kusanti, and H. Sulistyowati, "Enhancing Recurrent Neural Network Efficacy in Online Sales Predictions with Exploratory Data Analysis.," Int. J. …, 2025, [Online]. Available: https://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=2158107X&AN=183591984&h=Pq6l0f7lcPb79%2FBu%2BU56JZlg0j7LoOYFCcColHbbiDNCmXZEKclS1Tf1l5p0WQFfNrBjSp9ibJ%2BpMTyl34GhTQ%3D%3D&crl=c

[36] W. Ju et al., "A Comprehensive Survey on Deep Graph Representation Learning," Neural Networks, vol. 173, no. 1, pp. 1–100, 2024, doi: 10.1016/j.neunet.2024.106207.

[37] A. sanatan satpathy, S. kumar Sahoo, A. Mohanty, and P. P. Mohanty, "Strategies for enhancements of MSME resilience and sustainability in the post-COVID-19 era," Soc. Sci. Humanit. Open, vol. 11, no. October 2024, p. 101223, 2025, doi: 10.1016/j.ssaho.2024.101223.

A Computer Vision–Based Method for Determining the Vaccine Injection Position in Pangasius Fingerlings

Nguyen Phuc Truong[1], Luong Vinh Quoc Danh[2], Nguyen Chanh Nghiem[3]
College of Engineering, Can Tho University, Can Tho City, Vietnam[1, 2, 3]
Faculty of Electrical and Electronics Engineering, Vinh Long University of Technology Education, Vinh Long Province, Vietnam[1]

***Abstract*—Pangasius farming in the Mekong Delta is a major component of Vietnam's aquaculture industry, characterized by large-scale production, intensive farming practices, and significant contributions to export revenue. However, vaccination of Pangasius fingerlings is still predominantly performed manually, resulting in low productivity, high labor demand, and inconsistent injection accuracy, which limit large-scale deployment in commercial hatcheries. Therefore, there is an urgent need to develop an automated and accurate vaccination method for Pangasius fingerlings. This study proposes a novel computer vision–based approach for non-contact measurement of Pangasius fingerlings and accurate determination of the vaccine injection position. The proposed method leverages the image processing capabilities of the OpenCV library in combination with statistical morphological characteristics of Pangasius fingerlings to localize the injection position. The Python-implemented algorithm is lightweight and can run on an embedded Raspberry Pi platform, supporting practical in-field deployment. Experimental results demonstrate an average positioning accuracy of 97.65%, confirming the effectiveness of the proposed approach and its potential to serve as a technological foundation for automated vaccination systems in Pangasius aquaculture.**

***Keywords*—Computer vision; fish length measurement; fish vaccination; injection position; OpenCV; Pangasius**

I. Introduction

Pangasius is one of the major freshwater aquaculture products in the Mekong Delta, Vietnam, with a total export turnover of approximately USD 2 billion in 2024 [1]. Given its significant economic importance, effective disease control is essential to ensure the sustainability and competitiveness of the industry. In this context, vaccination of Pangasius fingerlings plays a crucial role in disease prevention, improving production efficiency, reducing the overuse of antibiotics, and minimizing antibiotic residues in fish products. However, vaccine injection in Pangasius fingerlings remains challenging due to their small size and morphological variability, which make accurate determination of the injection position difficult. Currently, the injection process is performed manually, resulting in low productivity, high labor demand, and inconsistent injection accuracy, as shown in Fig. 1. In addition, improper injection may damage fish internal organs and increase post-injection mortality rates. These limitations hinder large-scale and efficient vaccination in commercial hatcheries. To date, no dedicated solution has been reported that enables accurate, automated, and size-adaptive vaccination for Pangasius fingerlings. Therefore, the development of an automated and accurate vaccination method for Pangasius fingerlings is urgently required.



Fig. 1. Manual vaccination of pangasius fingerlings [2].

Motivated by these challenges, this study presents a computer vision–based approach for non-contact measurement of Pangasius fingerlings and accurate determination of the vaccine injection position. The proposed method employs image processing functions from the OpenCV library to determine the fish length. In particular, the vaccine injection position on the fish is determined based on a combination of image processing techniques and statistical data on the size and morphology of Pangasius fingerlings. For the first time, this study establishes a formulation for determining the appropriate vaccine injection position in Pangasius fingerlings using computer vision.

The organization of the present study is as follows: Section II is the review of the related literature, and Section III is the description of the proposed methodology. Section IV presents the experimental results. Section V provides concluding statements and directions for future research.

II. Related Works

Over the past two decades, the application of computer vision to fish length measurement has advanced considerably, supporting automated and non-contact measurement frameworks for modern fisheries and aquaculture systems. Among the earliest studies, White *et al.* [3] proposed the CatchMeter system, an onboard conveyor-based imaging setup that utilizes digital cameras to perform automated species classification and fish length measurement with millimeter-scale accuracy. Similarly, Jamaluddin *et al.* [4] proposed a non-contact fish length measurement system using camera-based image processing, highlighting its advantages in reducing handling stress and improving operational efficiency in aquaculture applications. The authors in [5] proposed a portable computer vision–based system using smartphone cameras and the OpenCV library to automatically measure striped catfish length, achieving an average accuracy of 97.71%. Kwon *et al.* [6] introduced a low-cost, vision-based system using dual smartphone cameras to estimate the body length and weight of olive flounder in indoor tanks with high accuracy. Lee *et al.* [7] developed an automated vision-guided Cartesian robotic system for flatfish vaccination that accurately determines injection positions under varying fish sizes and orientations.

Recent studies have integrated artificial intelligence algorithms and real-time capabilities. Tseng *et al.* [8] employed convolutional neural networks to estimate snout-to-fork length from images of harvested fish, achieving an accuracy of 95.74% in body length estimation. In [9], a stereo vision system was used to capture images of fish swimming freely in sea cages. The collected stereo data were used to train a YOLO model for simultaneous estimation of body length and weight, achieving a mean absolute percentage error of 5.5%. In [10], the authors proposed an automated framework for fish body length estimation by combining the RT-DETR deep learning model with Google ARCore technology. The system leveraged the smartphone Time-of-Flight capability to infer object distance and achieved measurement accuracy exceeding 95%. Rocha *et al.* [11] presented an automated approach for extracting morphological characteristics of freshwater fish using a Mask R-CNN framework in combination with the OpenCV library. The method estimated three body-length metrics, achieving an average relative error of 8.82% for standard length.

Despite extensive research on fish length measurement, most existing studies focus on grading, counting, or biomass estimation, while limited attention has been paid to its role in precise intervention tasks such as vaccination. In particular, research on vision-based determination of vaccine injection positions for fish fingerlings remains scarce. In addition, the substantial computational demands of deep learning frameworks restrict their adoption in cost-sensitive and resource-constraint aquaculture settings, underscoring the need for more efficient and deployable solutions. To address these limitations, this study integrates an OpenCV-based length measurement approach with statistical morphometric analysis to enable accurate and automated determination of vaccine injection positions for Pangasius fingerlings. The proposed processing pipeline is implemented in Python and can operate on embedded platforms such as Raspberry Pi, enabling practical in-field deployment.

III. Methodology

A. Image Collection

An imaging chamber was designed for acquiring images of Pangasius fingerlings. As shown in Fig. 2, the chamber consists of a Basler acA1300-200uc 1.3-megapixel camera [12] and a Raspberry Pi 4 kit with 8-GB RAM acting as the image processing unit. The images captured by the camera were transmitted to the Raspberry Pi kit [13] via a USB 3.0 interface for processing. The image processing algorithms were implemented in Python programming language (version 3.10.0) [14] using the OpenCV library (version 4.5.4.60) [15].

The fish used in this study were purchased from Pangasius farms in Vinh Long Province. The fingerlings had a body length ranging from 105 mm to 170 mm, measured from the head tip to the end of the caudal fin, and their weight ranged from 7.33 g to 35.94 g. The fish were anesthetized for imaging and morphometric measurements to establish a dataset for further analysis. The camera lens used in this study has a fixed focal length of 6 mm and an aperture of f/2.8. The fish images were captured under controlled lighting conditions at a resolution of 1280*x*1024 pixels. Prior to processing, they were resized to 800*x*600 pixels and stored in PNG format, providing a practical tradeoff between computational efficiency and measurement accuracy.

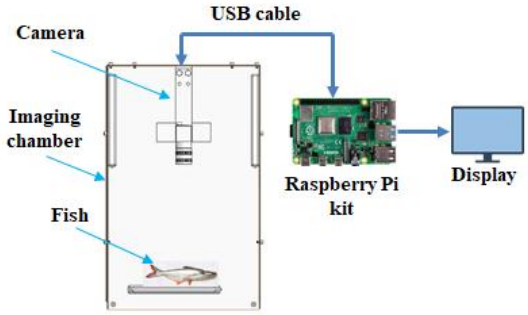


Fig. 2. Image acquisition setup.

B. Determination of the Vaccine Injection Position

The flowchart of the fish image processing procedure for determining the vaccine injection position is shown in Fig. 3. The procedure consists of seven processing steps: convert image to binary one, find contours and draw masks, extract the region of interest, determine fish orientation, find the position of fish's caudal peduncle, calculate the fish length, and determine the vaccine injection position.

1) Convert image to binary one: In this step, the original image is transformed into a binary image. The RGB image is first converted to grayscale using the OpenCV cv2.cvtColor function. Subsequently, Gaussian blurring is applied via cv2.GaussianBlur to suppress noise and smooth edges. Finally, adaptive thresholding is employed using cv2.adaptiveThreshold to convert the grayscale image into a binary image. The results of the RGB-to-binary conversion are illustrated in Fig. 4.

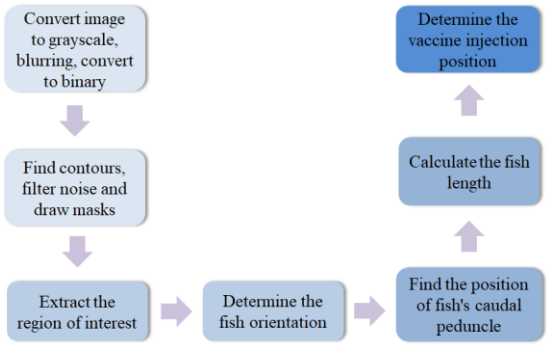


Fig. 3. The flowchart of the image processing algorithms.

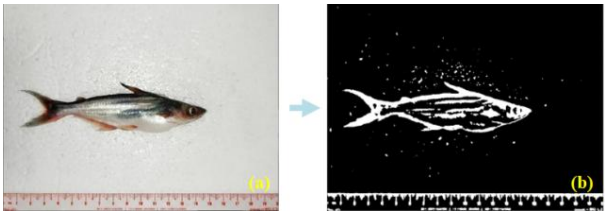


Fig. 4. Conversion from RGB to binary image: a) original RGB image; b) binary image.

2) Find contours and draw masks: In this step, object detection is performed on the binary image using the cv2.findContours function. To suppress noise, the contours.sort_contours function from the Imutils library [16] is employed to sort the detected contours and discard those with small areas. An image mask is then generated using the numpy.zeros function from the NumPy library [17] in combination with the cv2.drawContours function. The outcomes of the image masking process are presented in Fig. 5.

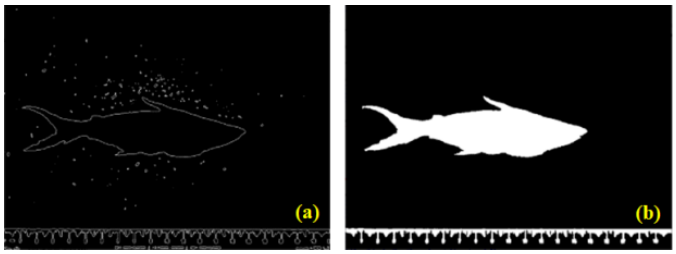


Fig. 5. Image masking: a) finding contours; b) drawing masks.

3) Extract the region of interest: The region of interest (ROI) corresponds to the image area containing the fish and is used to reduce computational cost. A rectangular ROI is extracted using the max and cv2.boundingRect functions. The extracted ROI is illustrated in Fig. 6.

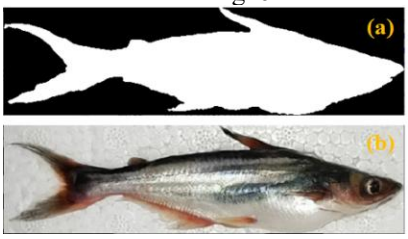


Fig. 6. Extracting the ROI: a) binary image; b) original RGB image.

4) Determine the fish orientation: During image acquisition, the fish were placed on the conveyor belt in a horizontal orientation, with their bodies aligned parallel to the direction of motion. Accordingly, two possible cases can occur: (a) the fish head is oriented to the left and (b) the fish head is oriented to the right, regardless of whether the ventral side is facing upward or downward. As shown in Fig. 7, h1 and h2 denote the heights of the fish image calculated at positions 0.2*w from the left and right boundaries of the ROI, respectively. Therefore, if h1 > h2, the fish head is oriented to the left; otherwise, the fish head is oriented to the right. For subsequent caudal peduncle detection, the ROI is transformed using the cv2.flip() function to ensure that the fish head is consistently oriented to the left, as illustrated in Fig. 8.

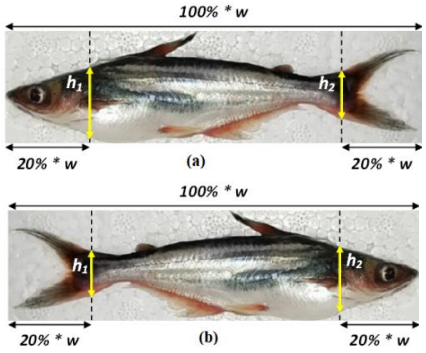


Fig. 7. Illustration of two different fish orientations: a) The fish head is oriented to the left; b) The fish head is oriented to the right.



Fig. 8. Result of changing fish orientation: a) the ROI of the binary image; b) the original image.

5) Find the position of the fish's caudal peduncle: Pangasius has a slender and elongated body that becomes narrower and flatter toward the tail. In addition, the caudal fin is thin and susceptible to damage, which can result in inaccurate length measurements. In this work, as a solution to this problem, the fish length is determined from the head tip to the narrowest region of the tail peduncle.

Fig. 9 illustrates the procedure for locating the narrowest part of the fish caudal peduncle. Specifically, the fish image ROI spanning from 0.7L to 0.9L (where L denotes the fish length) is analyzed to identify the minimum body width. A custom function, referred to as width_measuring, is developed to compute the distance between the first and last white pixels in each image column within the 0.7L÷0.9L region. The column exhibiting the minimum distance indicates the location of the narrowest part of the fish caudal peduncle. The algorithms corresponding to the proposed method are shown in Fig. 10 and Fig. 11. The outcome of this processing step is illustrated in Fig. 12.

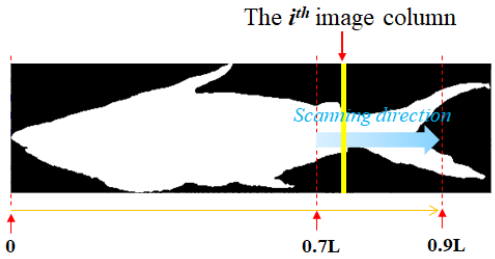


Fig. 9. Scanning for the narrowest position of the fish's caudal peduncle.

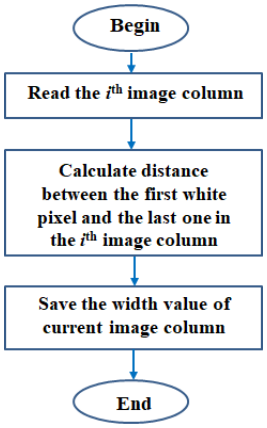


Fig. 10. The flowchart of the width_measuring function.

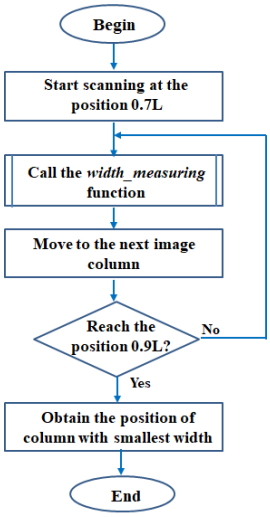


Fig. 11. Algorithm for locating the narrowest region of the fish caudal peduncle.

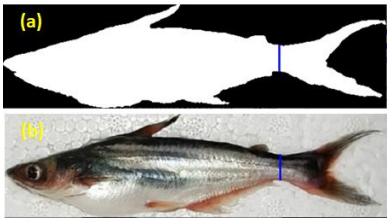


Fig. 12. Location of the narrowest part of the fish's caudal peduncle is marked with blue line in: a) the ROI of binary image and b) the original image.

6) Calculate the fish length: At this stage, the distance between the fish head tip and the location of minimum body width identified in the previous step is computed using the distance.euclidean function from the SciPy library [18]. This distance represents the fish length L, as depicted in Fig. 13, and is computed using the following Eq. (1):

$$L = N * s \quad (1)$$

where,

L: calculated fish length in millimeters.

N: the total number of image pixels located along the straight line from the fish mouth to the midpoint of the tail peduncle.

s: the physical size of a single pixel, equal to 0.264583 mm at an image resolution of 96 dpi.



Fig. 13. Determination of fish length.

7) Determine the vaccine injection position: The optimal injection needle position on the fish body is determined to be at 0.5÷0.75 of the pelvic length, anterior to the base of the pelvic fin [19]. Accordingly, the needle insertion position is point P, as illustrated in Fig. 14. The distance from the fish head tip to the injection point P, denoted as L3, is determined as follows [see Eq. (2)]:

$$L_3 = L_1 - 0.625 * L_2 \quad (2)$$

where,

L_1: the prepectoral length.

L_2: the pelvic length.

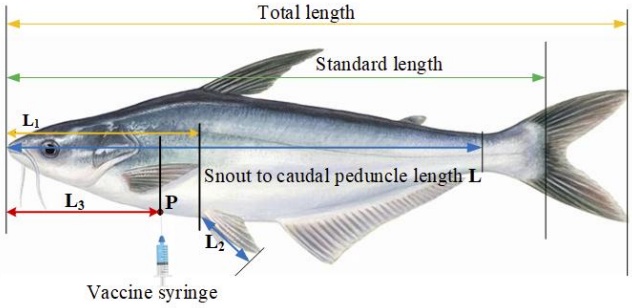


Fig. 14. Illustration of the optimal needle insertion position on the fish body.

In this study, the position of the vaccine injection point P is determined using image processing techniques in combination with experimental measurement data. Accordingly, the length L3 is determined based on Eq. (2) and the manually measured values of lengths L1 and L2.

Table I presents the experimental data from manual measurements of fish length Lm, prepectoral length *L*1, and

pelvic length L2 for 120 randomly selected Pangasius fingerlings. In Table I, R is defined as the ratio of L3 to the manually measured Lm, which corresponds to the length from the snout to the narrowest region of the caudal peduncle. Accordingly, the average value of the ratio R is determined to be 0.41.

TABLE I. MEASURED LENGTH AND WEIGHT DATA OF 120 PANGASIUS FINGERLINGS

	Manual measurement L_m: Snout to caudal peduncle length L_1: Prepectoral length L_2: Pelvic fin length L_3: Injection point w: Manual weight measurement R: Rate factor						**Image-based length measurement**	**Absolute error**	**Accuracy of image-based length measurement** $\left(100-\frac{\lvert L-L_m\rvert\cdot 100}{L_m}\right)$		**Manual measurement** L_m: Snout to caudal peduncle length L_1: Prepectoral length L_2: Pelvic fin length L_3: Injection point w: Manual weight measurement R: Rate factor						**Image-based length measurement**	**Absolute error**	**Accuracy of image-based length measurement** $\left(100-\frac{\lvert L-L_m\rvert\cdot 100}{L_m}\right)$
No.	L_m	L_1	L_2	L_3	w	R	L	$\lvert L-L_m\rvert$	(%)	No.	L_m	L_1	L_2	L_3	w	R	L	$\lvert L-L_m\rvert$	(%)
1	75.40	38.60	9.26	32.81	9.15	0.44	79.16	3.76	95.01	61	93.20	45.63	12.64	37.73	14.02	0.40	97.51	4.31	95.38
2	77.20	36.62	12.11	29.05	8.56	0.38	76.82	0.38	99.51	62	93.44	45.20	13.21	36.94	13.12	0.40	97.24	3.80	95.93
3	77.84	38.08	9.96	31.86	7.33	0.41	78.16	0.32	99.59	63	93.47	45.98	11.67	38.69	13.71	0.41	94.96	1.49	98.41
4	79.24	40.33	10.80	33.58	9.39	0.42	83.53	4.29	94.59	64	94.15	47.50	10.52	40.93	13.08	0.43	99.52	5.37	94.30
5	79.28	37.96	10.70	31.27	9.12	0.39	82.02	2.74	96.54	65	95.00	47.18	10.44	40.66	14.13	0.43	100.32	5.32	94.40
6	79.70	41.03	9.09	35.35	8.33	0.44	83.28	3.58	95.51	66	95.56	46.42	13.16	38.20	12.95	0.40	97.36	1.80	98.12
7	80.10	39.13	11.44	31.98	7.76	0.40	81.99	1.89	97.64	67	95.76	46.70	12.30	39.01	13.08	0.41	99.75	3.99	95.83
8	80.44	40.31	12.47	32.52	8.73	0.40	80.31	0.13	99.84	68	95.81	47.32	12.99	39.20	12.31	0.41	95.76	0.05	99.95
9	81.50	40.97	9.30	35.16	10.02	0.43	83.61	2.11	97.41	69	95.89	47.80	12.13	40.22	16.62	0.42	100.13	4.24	95.58
10	81.55	39.80	10.60	33.18	8.56	0.41	81.50	0.05	99.94	70	96.41	45.53	13.76	36.93	13.35	0.38	101.37	4.96	94.86
11	82.06	41.23	10.24	34.83	8.74	0.42	83.56	1.50	98.17	71	96.44	48.85	13.24	40.58	15.48	0.42	100.20	3.76	96.10
12	82.09	41.55	11.61	34.29	8.26	0.42	81.76	0.33	99.60	72	96.52	48.13	10.72	41.43	14.36	0.43	100.28	3.76	96.10
13	82.76	40.61	8.30	35.42	10.01	0.43	86.04	3.28	96.04	73	97.18	46.57	13.22	38.31	14.81	0.39	97.20	0.02	99.98
14	82.80	39.87	12.24	32.22	10.38	0.39	87.04	4.24	94.88	74	97.33	45.76	13.25	37.48	12.96	0.39	99.75	2.42	97.51
15	82.92	41.24	9.54	35.28	10.12	0.43	86.41	3.49	95.79	75	97.40	48.44	14.38	39.45	13.96	0.41	99.89	2.49	97.44
16	83.38	40.22	9.75	34.13	10.80	0.41	83.23	0.15	99.82	76	97.66	48.01	12.65	40.10	16.41	0.41	101.68	4.02	95.88
17	83.79	41.80	12.84	33.78	9.91	0.40	84.82	1.03	98.77	77	97.93	47.40	13.10	39.21	16.64	0.40	99.22	1.29	98.68
18	84.37	40.56	8.55	35.22	10.97	0.42	87.15	2.78	96.70	78	98.36	47.14	12.40	39.39	15.20	0.40	103.20	4.84	95.08
19	84.75	40.64	9.07	34.97	11.07	0.41	87.61	2.86	96.63	79	98.58	48.68	13.29	40.37	14.04	0.41	101.38	2.80	97.16
20	85.12	41.18	11.51	33.99	11.17	0.40	89.45	4.33	94.91	80	98.71	47.79	11.58	40.55	14.82	0.41	103.81	5.10	94.83
21	85.17	40.68	11.46	33.52	10.39	0.39	86.35	1.18	98.61	81	98.93	47.69	12.94	39.60	14.45	0.40	104.4	5.47	94.47
22	85.49	41.27	10.45	34.74	10.66	0.41	88.37	2.88	96.63	82	100.24	47.80	11.33	40.72	14.18	0.41	103.45	3.21	96.80
23	85.51	42.42	11.00	35.55	11.63	0.42	90.20	4.69	94.52	83	100.85	48.61	12.78	40.62	15.92	0.40	106.44	5.59	94.46
24	85.75	42.93	11.74	35.59	11.02	0.42	92.06	6.31	92.64	84	100.92	48.08	11.14	41.12	14.39	0.41	103.20	2.28	97.74
25	86.00	44.10	9.88	37.93	12.28	0.44	90.75	4.75	94.48	85	101.47	50.91	12.56	43.06	16.16	0.42	108.29	6.82	93.28

26	86.15	42.67	10.06	36.38	10.24	0.42	85.98	0.17	99.80	86	101.81	50.90	13.06	42.74	16.01	0.42	106.72	4.91	95.18
27	86.62	43.36	8.71	37.92	12.05	0.44	90.31	3.69	95.74	87	101.83	48.83	12.83	40.81	16.48	0.40	105.75	3.92	96.15
28	86.73	43.17	12.07	35.63	12.10	0.41	90.79	4.06	95.32	88	102.16	49.72	11.47	42.55	17.92	0.42	108.53	6.37	93.76
29	86.97	42.03	8.69	36.60	11.35	0.42	90.18	3.21	96.31	89	102.28	49.38	13.35	41.04	17.41	0.40	109.01	6.73	93.42
30	87.09	42.72	10.92	35.90	12.17	0.41	91.52	4.43	94.91	90	102.34	50.16	13.20	41.91	17.08	0.41	107.42	5.08	95.04
31	87.09	43.26	11.87	35.84	12.45	0.41	87.74	3.35	96.18	91	103.09	49.25	14.48	40.20	17.60	0.39	108.47	5.38	94.78
32	87.62	42.85	10.27	36.43	10.12	0.42	90.97	0.81	99.08	92	103.19	49.82	11.97	42.34	17.32	0.41	107.60	4.41	95.73
33	87.70	43.60	8.82	38.09	11.97	0.43	88.51	5.73	93.47	93	103.34	50.16	14.34	41.20	18.87	0.40	106.79	3.45	96.66
34	87.71	44.78	11.23	37.76	11.78	0.43	93.44	2.95	96.64	94	103.54	48.96	11.58	41.72	17.83	0.40	108.78	5.24	94.94
35	87.86	43.63	10.20	37.26	11.57	0.42	90.81	4.39	95.01	95	104.13	50.58	15.30	41.02	17.77	0.39	109.16	5.03	95.17
36	87.96	44.38	12.11	36.81	12.64	0.42	92.35	2.50	97.16	96	104.37	51.13	14.23	42.24	17.31	0.40	110.68	6.31	93.95
37	87.98	43.57	11.04	36.67	11.30	0.41	90.48	1.66	98.13	97	105.13	51.41	11.55	44.19	16.82	0.42	110.98	5.85	94.44
38	88.85	43.81	12.51	35.99	12.26	0.40	90.51	2.00	97.75	98	105.18	51.19	14.47	42.15	18.14	0.40	111.14	5.96	94.33
39	88.89	44.11	10.46	37.57	13.37	0.42	90.89	3.81	95.71	99	105.96	51.59	11.90	44.15	18.91	0.42	109.94	3.98	96.24
40	88.89	43.01	12.29	35.33	11.21	0.40	92.72	1.50	98.32	100	106.76	51.55	11.30	44.49	18.27	0.42	112.00	5.24	95.09
41	89.37	43.05	12.86	35.01	11.53	0.39	90.87	5.87	93.43	101	106.96	51.87	11.80	44.50	18.33	0.42	112.72	5.76	94.61
42	89.37	44.75	12.04	37.23	11.91	0.42	95.26	5.07	94.33	102	107.03	51.45	13.84	42.80	18.93	0.40	113.13	6.10	94.30
43	89.45	44.44	13.17	36.21	12.69	0.40	94.52	5.36	94.01	103	107.39	50.91	13.47	42.49	19.26	0.40	113.93	6.54	93.91
44	89.54	42.96	11.82	35.57	13.31	0.40	94.90	5.20	94.20	104	108.76	51.55	11.25	44.52	17.92	0.41	115.25	6.49	94.03
45	89.61	42.37	11.31	35.30	11.47	0.39	94.81	1.57	98.25	105	108.87	53.77	13.18	45.53	21.21	0.42	113.35	4.48	95.89
46	89.95	44.06	11.17	37.08	14.45	0.41	91.52	4.98	94.47	106	109.75	53.13	11.64	45.86	18.38	0.42	116.71	6.96	93.66
47	90.06	44.17	11.23	37.15	13.82	0.41	95.04	4.55	94.96	107	109.91	52.93	14.79	43.69	18.79	0.40	116.08	6.17	94.39
48	90.30	44.95	12.66	37.04	12.04	0.41	94.85	0.87	99.04	108	110.54	55.63	13.79	47.01	20.32	0.43	114.73	4.19	96.21
49	90.32	44.28	11.87	36.86	13.56	0.41	91.19	3.34	96.31	109	111.16	52.87	14.06	44.08	21.22	0.40	114.83	3.67	96.70
50	90.40	44.68	11.84	37.28	14.36	0.41	93.74	5.38	94.07	110	112.86	54.77	15.23	45.25	23.35	0.40	117.89	5.03	95.54
51	90.67	43.21	12.02	35.70	12.17	0.39	96.05	0.48	99.47	111	114.45	53.92	13.96	45.20	22.82	0.39	120.12	5.67	95.05
52	91.04	44.24	11.12	37.29	11.74	0.41	91.52	4.50	95.08	112	116.19	54.93	12.70	46.99	25.22	0.40	122.90	6.71	94.22
53	91.40	43.69	12.13	36.11	12.94	0.39	95.90	4.70	94.86	113	116.79	55.90	16.21	45.77	23.10	0.39	121.57	4.78	95.91
54	91.42	45.51	12.11	37.94	12.99	0.41	96.12	1.83	98.00	114	117.16	57.49	13.21	49.23	24.15	0.42	122.50	5.34	95.44
55	91.63	45.24	12.52	37.42	14.17	0.41	93.46	4.47	95.13	115	121.54	57.94	16.76	47.47	26.76	0.39	129.66	8.12	93.32
56	91.85	43.80	10.65	37.14	12.49	0.40	96.32	3.90	95.76	116	122.28	59.76	17.62	48.75	34.55	0.40	131.14	8.86	92.75
57	91.88	44.76	12.03	37.24	14.15	0.41	95.78	4.63	94.96	117	122.58	59.36	16.01	49.35	32.26	0.40	129.71	7.13	94.18
58	91.91	45.69	13.34	37.35	13.49	0.40	96.54	2.45	97.35	118	125.85	61.67	16.86	51.13	28.92	0.41	132.14	6.29	95.00
59	92.50	47.11	12.82	39.10	13.58	0.42	94.95	4.19	95.47	119	126.70	63.03	16.54	52.69	35.26	0.42	133.88	7.18	94.33
60	92.52	45.34	11.63	38.07	13.15	0.41	96.71	2.27	97.55	120	133.27	61.74	15.72	51.92	35.94	0.39	138.95	5.68	95.74

As shown in Table I, the absolute error between the fish length measured using image processing L and that obtained by manual measurement L_m ranges from a minimum of 0.02 mm to a maximum of 8.86 mm, with an average error of 3.92 mm (i.e., 4%). This error may result from inaccuracies in manual measurements and in pixel-count calculations on the fish image.

Based on the statistical data in Table I, the correlation between the manually measured fish length L_m and the image-derived fish length L was obtained using the Matlab curve fitting toolbox [20] with a coefficient of determination R^2 = 0.9856, as shown in Fig. 15 and expressed by Eq. (3):

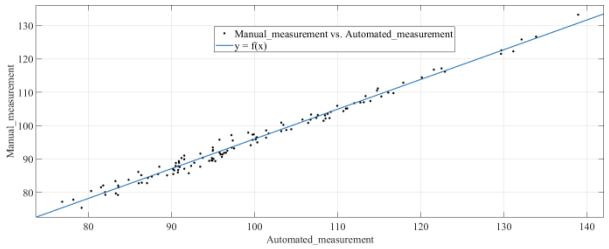


Fig. 15. Correlation function between L_m and L.

$$L_m = 0.8907 * L + 6.975 \quad (3)$$

In summary, the vaccine injection point on the fish P is defined by the distance from the fish head tip to P, denoted as $L_{injection}$, which is computed as follows:

$$L_{injection} = 0.41 * (0.8907 * L + 6.975) \quad (4)$$

Based on the fish length determined using image processing techniques and Eq. (4), the microprocessor can control the vaccine syringe to move to the determined injection point.

8) Fish weight estimation: From the statistical survey results of fish length L_m and the corresponding weight W in Table I, the fish weight can be estimated as a function of fish length with a coefficient of determination R^2 = 0.9502, and is expressed by the following Eq. (5):

$$W = 0.0000617y^3 - 0.01268y^2 + 1.1y - 29.07 \quad (5)$$

where, y is the fish length calculated from Eq. (3).

IV. Experimental Results

To verify the effectiveness and accuracy of the proposed method, 65 Pangasius fingerlings with weights ranging from 6.52 g to 41.3 g were randomly selected for validation. The fingerlings were reared in tanks and anesthetized prior to measurement. Fish size measurements were then conducted using two methods: using a digital caliper and an image-based method, as shown in Fig. 16. The comparison results between image-based measurements and manual measurements are presented in Table II. The experimental results indicate that the calculated fish length data exhibit an average absolute error of 0.94 mm, with accuracy ranging from 96.79% to 99.96%, and an average accuracy of 99.05%. The injection position estimation results show an average absolute error of 1 mm and an average accuracy of 97.65%. In addition, the fish weight estimation results achieve an average accuracy of 93.75% with an average absolute error of 2.66 g. The average processing time from image capture to result display is 0.25 seconds.

The accuracy of fish length, injection position, and weight estimations is computed as given in Eq. (6), Eq. (7), and Eq. (8), respectively:

$$Fish\ length\ accuracy = 100 - \frac{|L - L_m|}{L_m} * 100 \quad (6)$$

$$Injection\ accuracy\ (\%) = 100 - \frac{|L_{injection} - L_3|}{L_3} * 100 \quad (7)$$

$$Weight\ accuracy\ (\%) = 100 - \frac{|W_i - W_m|}{W_m} * 100 \quad (8)$$

TABLE II. Experimental Measurement Results

No.	Snout to caudal peduncle length measurement		Weight		Prepectoral length	Pelvic fin length	Injection point		Absolute errors			Accuracy (%)		
	Manual (L_m)	Image-based (L)	Manual (W_m)	Image-based (W_i)	Manual measurement (L_1)	Manual measurement (L_2)	Manual measurement (L_3)	Image-based measurement $L_{injection}$	$\lvert L - L_m \rvert$	$\lvert W_i - W_m \rvert$	$\lvert L_{injection} - L_3 \rvert$	$100 - \frac{\lvert L - L_m \rvert}{L_m} * 100$ Length	$100 - \frac{\lvert W_i - W_m \rvert}{W_m} * 100$ Weight	$100 - \frac{\lvert L_{injection} - L_3 \rvert}{L_3} * 100$ Injection point
1	75.21	74.30	6.55	7.97	37.56	9.69	31.50	30.46	0.91	1.42	1.04	98.79	78.32	96.70
2	79.19	81.26	8.98	9.69	38.89	9.50	32.95	33.32	2.07	0.71	0.36	97.39	92.09	98.88
3	79.77	78.26	6.52	8.93	39.09	10.73	32.38	32.09	1.51	2.41	0.30	98.11	63.04	99.10
4	81.54	83.09	10.11	10.18	40.30	9.07	34.63	34.07	1.55	0.07	0.56	98.10	99.31	98.38
5	82.19	81.46	9.25	9.75	40.46	10.46	33.92	33.40	0.73	0.50	0.53	99.11	94.59	98.47
6	83.38	82.87	9.94	10.12	40.34	11.64	33.07	33.98	0.51	0.18	0.91	99.39	98.19	97.25
7	84.54	83.82	10.88	10.38	40.84	10.73	34.13	34.37	0.72	0.50	0.23	99.15	95.40	99.30
8	85.43	84.54	11.01	10.58	42.38	11.56	35.16	34.66	0.89	0.43	0.49	98.96	96.09	98.58

9	85.97	87.34	11.21	11.36	43.39	11.59	36.15	35.81	1.36	0.15	0.34	98.41	98.66	99.06
10	86.68	87.72	11.54	11.50	41.80	12.72	33.85	35.96	1.04	0.04	2.11	98.80	99.65	93.77
11	87.53	85.13	10.86	10.74	42.86	10.54	36.27	34.90	2.40	0.12	1.37	97.26	98.90	96.22
12	87.86	85.54	10.63	10.86	42.52	11.30	35.46	35.07	2.32	0.23	0.38	97.36	97.84	98.90
13	88.47	88.75	11.16	11.81	42.91	12.06	35.37	36.39	0.28	0.65	1.02	99.68	94.18	97.12
14	89.36	90.23	11.96	12.27	43.52	12.16	35.92	36.99	0.87	0.31	1.07	99.03	97.41	97.02
15	89.51	87.62	11.04	11.47	43.13	12.38	35.39	35.92	1.89	0.43	0.53	97.89	96.11	98.50
16	89.97	90.88	13.93	12.48	44.03	9.65	38.00	37.26	0.91	1.45	0.74	98.99	89.59	98.05
17	90.39	87.49	11.62	11.43	43.61	11.69	36.30	35.87	2.90	0.19	0.43	96.79	98.36	98.82
18	91.20	92.06	14.08	12.87	44.06	10.92	37.24	37.75	0.86	1.21	0.51	99.06	91.41	98.63
19	91.82	91.44	14.59	12.67	42.16	11.81	34.78	37.49	0.38	1.92	2.71	99.59	86.84	92.21
20	92.38	92.79	13.28	13.12	44.92	10.40	38.42	38.04	0.41	0.16	0.38	99.56	98.80	99.01
21	92.99	93.34	13.93	13.31	44.96	10.33	38.50	38.27	0.35	0.62	0.24	99.62	95.55	99.40
22	93.94	92.19	12.69	12.91	46.89	14.48	37.84	37.80	1.75	0.22	0.04	98.14	98.27	99.89
23	95.65	94.41	14.35	13.68	48.81	11.71	41.49	38.71	1.24	0.67	2.78	98.70	95.33	93.30
24	95.89	94.90	13.86	13.86	47.76	10.66	41.10	38.91	0.99	0.00	2.19	98.97	100.00	94.67
25	96.95	95.66	13.93	14.13	49.31	12.22	41.67	39.22	1.29	0.20	2.45	98.67	98.56	94.12
26	97.49	98.21	14.56	15.11	47.16	11.69	39.85	40.27	0.72	0.55	0.41	99.26	96.22	98.95
27	98.37	97.80	12.53	14.94	48.74	13.47	40.32	40.10	0.57	2.41	0.22	99.42	80.77	99.45
28	99.65	99.61	15.13	15.67	49.37	9.93	43.16	40.84	0.04	0.54	2.32	99.96	96.43	94.62
29	101.27	101.85	16.89	16.62	48.81	13.49	40.38	41.76	0.58	0.27	1.38	99.43	98.40	96.58
30	102.12	102.45	16.15	16.88	49.52	13.62	41.01	42.00	0.33	0.73	1.00	99.68	95.48	97.59
31	102.75	103.13	17.42	17.19	50.16	13.10	41.97	42.28	0.38	0.23	0.31	99.63	98.68	99.26
32	103.36	102.67	16.40	16.98	51.37	14.49	42.31	42.10	0.69	0.58	0.22	99.33	96.46	99.50
33	104.82	104.15	16.94	17.66	49.18	14.33	40.22	42.70	0.67	0.72	2.48	99.36	95.75	93.83
34	106.21	105.63	17.58	18.36	53.19	15.55	43.47	43.31	0.58	0.78	0.16	99.45	95.56	99.63
35	106.60	106.47	18.92	18.78	54.72	14.99	45.35	43.65	0.13	4.14	1.70	99.88	99.26	96.25
36	107.28	107.61	20.82	19.35	52.09	13.12	43.89	44.12	0.33	1.47	0.23	99.69	92.94	99.48
37	108.91	109.09	19.90	20.13	54.27	15.48	44.60	44.73	0.18	3.77	0.13	99.83	98.84	99.71
38	109.06	109.76	18.96	20.49	52.14	13.73	43.56	45.00	0.70	1.53	1.44	99.36	91.93	96.69
39	109.91	110.87	23.78	21.11	56.71	15.96	46.74	45.46	0.96	6.67	1.28	99.13	88.77	97.26
40	110.15	111.28	21.10	21.34	55.93	15.95	45.96	45.62	1.13	3.76	0.34	98.97	98.86	99.26
41	110.29	111.18	21.45	21.28	56.97	15.12	47.52	45.58	0.89	4.17	1.94	99.19	99.21	95.92
42	110.69	110.22	20.30	20.74	55.13	16.88	44.58	45.19	0.47	0.44	0.61	99.58	97.83	98.63
43	110.73	111.38	21.05	21.40	54.75	15.26	45.21	45.67	0.65	3.65	0.46	99.41	98.34	98.98
44	110.85	110.53	25.27	20.92	54.97	16.74	44.51	45.32	0.32	8.35	0.81	99.71	82.79	98.18
45	111.80	111.22	21.53	21.31	55.94	15.77	46.08	45.60	0.58	4.22	0.48	99.48	98.98	98.96
46	113.63	115.01	23.90	23.58	58.52	14.35	49.55	47.15	1.38	4.32	2.40	98.79	98.66	95.16
47	113.70	114.65	22.57	23.35	57.94	15.59	48.20	47.01	0.95	3.22	1.19	99.16	96.54	97.53
48	113.76	113.54	25.22	22.67	58.83	16.09	48.77	46.55	0.22	6.55	2.22	99.81	89.89	95.45
49	114.12	113.41	23.19	22.59	56.61	16.31	46.42	46.50	0.71	4.60	0.08	99.38	97.41	99.83
50	114.77	113.46	26.06	22.62	57.48	16.48	47.18	46.52	1.31	7.44	0.66	98.86	86.80	98.60
51	115.06	115.14	25.73	23.66	56.15	14.82	46.89	47.21	0.08	2.07	0.32	99.93	91.95	99.32
52	115.63	114.93	23.98	23.53	58.26	17.18	47.52	47.12	0.70	4.45	0.40	99.39	98.12	99.16
53	116.53	116.81	25.40	24.75	59.67	15.97	49.69	47.89	0.28	4.65	1.80	99.76	97.44	96.38
54	117.41	116.57	25.02	24.59	57.98	16.85	47.45	47.79	0.84	4.43	0.34	99.28	98.28	99.28
55	117.62	116.32	27.81	24.42	58.67	16.39	48.43	47.69	1.30	7.39	0.74	98.89	87.81	98.47
56	118.43	119.12	27.26	26.33	60.36	15.43	50.72	48.84	0.69	4.93	1.88	99.42	96.59	96.29
57	118.72	117.04	27.45	24.90	59.50	15.73	49.67	47.99	1.68	6.55	1.68	98.58	90.71	96.62
58	119.96	121.64	25.10	28.17	60.53	17.22	49.77	49.87	1.68	0.93	0.10	98.60	87.77	99.80
59	120.54	117.49	26.23	25.20	56.90	15.47	47.23	48.17	3.05	1.03	0.94	97.47	96.07	98.01
60	121.02	120.19	30.49	27.09	62.02	17.16	51.30	49.28	0.83	7.40	2.02	99.31	88.85	96.06
61	122.04	121.45	31.74	28.02	62.10	17.08	51.43	49.79	0.59	7.72	1.64	99.52	88.28	96.81
62	123.63	122.39	34.61	28.74	61.04	19.15	49.07	50.18	1.24	9.87	1.11	99.00	83.04	97.74
63	123.85	123.67	31.59	29.74	62.22	16.92	51.65	50.70	0.18	5.85	0.95	99.85	94.14	98.16
64	125.28	126.73	41.30	32.27	60.13	13.88	51.46	51.96	1.45	9.03	0.50	98.84	78.14	99.03
65	128.95	130.15	37.89	35.33	66.64	17.10	55.95	53.36	1.20	6.56	2.59	99.07	93.24	95.37



Fig. 16. Conducting fish length measurements: a) Pangasius fingerlings tank; b) Fish were anesthetized prior to measurement; c) Manual measurement using digital caliper; d) Image-based measurement using computer vision.

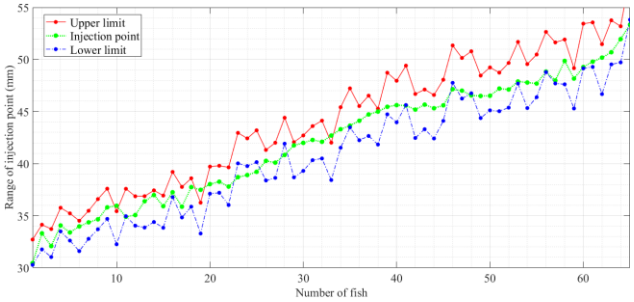


Fig. 17. Calculated results of the vaccine injection position.

The results of the vaccine injection position estimation and the practical injection tolerance range are illustrated in Fig. 17. The experimental data show that most of the calculated injection positions fall within the allowable limits, except for three cases with deviations greater than 1 mm, which are 1.32 mm, 1.24 mm, and 1.08 mm, respectively. In practice, it has also been observed that some Pangasius fingerlings have different pelvic fin lengths despite having the same body length. This may be one of the causes of the observed discrepancies. Besides, the use of a camera equipped with a short-focal-length lens (6 mm, f/2.8) introduces spatial nonuniformity in pixel density, which also contributes to measurement deviations. However, these errors remain within acceptable limits for practical implementation. The proposed method offers a more accurate and reliable approach to automated injection than manual vaccination, where high-intensity human operation can introduce errors that compromise post-injection fish health.

V. Conclusion and Future Work

This study has proposed and experimentally validated a novel computer vision–based approach for accurately determining the vaccine injection position in Pangasius fingerlings. By leveraging the image processing capabilities of the OpenCV library in combination with statistical morphological data of Pangasius fingerlings, the proposed method achieved an average accuracy of 97.65%. The open-source Python-based pipeline operates on embedded platforms, enabling practical in-field deployment. This work is the first to establish a computer-vision–based formulation for locating the vaccine injection position in Pangasius fingerlings. The results demonstrate the feasibility of applying computer vision techniques to support automated vaccination tasks for Pangasius fingerlings.

The estimation accuracy of the image-based approach can be further improved by optimizing the illumination conditions within the imaging chamber and by selecting a camera lens with a focal length appropriate for the fish size. Future work will focus on integrating the proposed vision-based measurement approach with mechatronic actuators and an embedded control system toward a fully automated vaccination platform. Such integration is expected to increase throughput, improve injection consistency, and substantially reduce labor costs in commercial Pangasius hatcheries.

Acknowledgment

The authors would like to sincerely thank Assoc. Prof. Tu Thanh Dung, College of Aquaculture and Fishery, Can Tho University, for her valuable advice on vaccination techniques for Pangasius fingerlings.

Declaration on Generative AI

The authors acknowledge using ChatGPT to improve the clarity and grammar of the manuscript. The content, analysis, and conclusions remain the sole responsibility of the authors.

References

[1] Vietnam News. Vietnam's pangasius exports expected to reach $2b in 2024. https://vietnamnews.vn/economy/1687381/viet-nam-s-pangasius-exports-expected-to-reach-2b-in-2024.html

[2] Hien La. Vaccines and a vision for sustainable development in pangasius farming. https://tepbac.com/tin-tuc/full/vac-xin-va-tam-nhin-phat-trien-ben-vung-trong-nghe-nuoi-ca-tra-33297.html (in Vietnamese)

[3] D.J. White, C. Svellingen, N.J.C. Strachan, "Automated Measurement of Species and Length of Fish by Computer Vision". Fisheries Research, Volume 80, Issues 2–3, 203-210 (2006).

[4] Jamaluddin, M. H., Siong Seng, C., Shukor, A. Z., Ali Ibrahim, F., Miskon, M. F., Mohd Aras, M. S., Md Ghazaly, M., & Ranom, R., "The Effectiveness of Fish Length Measurement System using Non-Contact Measuring Approach". Jurnal Teknologi, 77(20) (2015).

[5] L. H. Phong, N. P. Truong, L. V. Q. Danh, V. H. Nam, N. T. Tung, and T. T. Dung, "A Portable System for Automated Measurement of Striped Catfish Length Using Computer Vision," in Ubiquitous Intelligent Systems, Singapore, P. Karuppusamy, F. P. García Márquez, and T. N. Nguyen, Eds., 2022// 2022: Springer Nature Singapore, pp. 607-618.

[6] Kwon, Inyeong, Hang Thi Phuong Nguyen, Paththige Waruni Prasadini Fernando, Hieyong Jeong, Sungju Jung, and Taeho Kim. "Prototype Mobile Vision System for Automatic Length Estimation of Olive Flounder (Paralichthys olivaceus) in Indoor Aquaculture" Journal of Marine Science and Engineering, 2025, 13, no. 6: 1167. https://doi.org/10.3390/jmse13061167.

[7] Lee D. G., Bong-Jin Cha, Seong-Wook Park, Mun-Gyeong Kwon, Guo-Cheng Xu, Hee-Je Kim, "Development of a vision-based automatic vaccine injection system for flatfish,". Aquacultural Engineering, Volume 54, 2013, pages 78-84, ISSN 0144-8609, https://doi.org/10.1016/j.aquaeng.2012.12.001.

[8] Tseng C. H, Ching-Lu Hsieh, Yan-Fu Kuo. "Automatic measurement of the body length of harvested fish using convolutional neural networks," Biosystems Engineering, Volume 189, 2020, Pages 36-47, ISSN 1537-5110.

[9] Tonachella, N., Martini, A., Martinoli, M. et al. "An affordable and easy-to-use tool for automatic fish length and weight estimation in

mariculture". Sci Rep 12, 15642 (2022). https://doi.org/10.1038/s41598-022-19932-9.

[10] Kang, Seung-Beom, Seung-Gyu Kim, Sang-Hyun Lee, and Tae-Ho Im. "A Study on the Automation of Fish Species Recognition and Body Length Measurement System," Fishes, 2024, no. 9: 349. https://doi.org/10.3390/fishes9090349

[11] Rocha, W.S., da Fonseca, T.F.C., Watanabe, C.Y.V. et al. "Automatic measurement of fish from images using convolutional neural networks,". Multimedia Tools and Applications, 84, 6327–6347 (2025). https://doi.org/10.1007/s11042-024-19180-1

[12] Basler ace. acA1300-200uc. https://www.baslerweb.com/en/shop/aca1300-200uc/

[13] Raspberry Pi 4. https://www.raspberrypi.com/products/raspberry-pi-4-model-b/

[14] Python. https://www.python.org/downloads/

[15] OpenCV library. https://opencv.org

[16] Imutils library. https://pypi.org/project/imutils

[17] Numpy library. https://numpy.org

[18] Scipy library. https://scipy.org

[19] Pharmaq. Vaccine Package Insert. (2018). https://pharmaq-cdn-e9g8bdeng6hhcsee.z01.azurefd.net/media/njgph0sn/pil-ajpanga2-vn-v3-english.pdf

[20] Matlab. Curve fitting toolbox. https://www.mathworks.com/products/curvefitting.html

Exposure-Based Media Mix Modeling Using Machine Learning and Genetic Algorithms

An Audience Reach Optimization Framework for Advertising Budget Allocation

Thejan Dulara, Indra Mahakalanda, Prasanga Jayathunga
University of Moratuwa, Katubedda 10400, Sri Lanka

***Abstract*—Media budget allocation remains a persistent challenge in the advertising industry. Inefficient spending and biased planning decisions often reduce campaign effectiveness. Advertisers struggle to balance investments across television, radio, press, and digital platforms while managing diminishing returns. This study proposes a data-driven media mix determination model that integrates supervised machine learning with genetic algorithm-based optimization. The objective is to maximize audience reach while maintaining cost efficiency. Unlike traditional media mix models that rely on aggregated medium-level performance, this study adopts an audience exposure-based modelling approach. Facebook and YouTube are used as digital media platforms in this study. Television and digital models are trained using exposure-based reach measures, such as 1 plus and 2 plus reach. Machine learning models, including decision trees, random forests, XGBoost, and LightGBM, are evaluated to capture complex and nonlinear relationships between spend and exposure-based reach. Smoothed reach response curves are used to identify efficiency levels and saturation points for each medium. A genetic algorithm is then applied to derive the optimal budget allocation across media under efficiency, reach, and cost constraints. The model is trained using real advertising data from the Sri Lankan market, ensuring practical relevance and applicability. Although the analysis is based on a country-specific dataset, the model is transferable to markets of similar scale. This study contributes to the literature by introducing an exposure-driven media mix modelling approach that improves media budget planning accuracy and supports more effective advertising decision-making.**

***Keywords*—Media mix determination; saturation points; audience reach; audience exposure**

I. Introduction

Advertisers seek to reach the maximum possible audience while using their budgets efficiently [1]. However, allocating budgets across television, radio, print, and digital media remains a persistent challenge [1]. Each medium exhibits different response behaviors, and excessive spending on a single channel often results in diminishing returns [2]. These challenges increase the risk of inefficient budget allocation and reduced campaign effectiveness.

A data-driven approach that combines machine learning based prediction and optimization techniques can help address this problem [3]. Predictive models enable advertisers to understand how audience reach responds to varying spend levels across media. Optimization techniques then support the identification of an optimal media mix that maximizes reach at the lowest possible cost [3]. Media Mix Modeling (MMM) provides a structured framework for this purpose. Existing MMM approaches generally follow two directions. One focuses on quantitative objectives, such as maximizing reach while minimizing cost [4]. The other incorporates qualitative considerations, including campaign fitness, creative alignment, and historical effectiveness [4].

Most traditional MMM studies rely on aggregated spend to reach relationships at the media channel level [5]. Such approaches may overlook the effects of repeated audience exposure and the nonlinear nature of media response [6]. To address this limitation, this study adopts an exposure-based perspective. Television and digital media performance are modeled using exposure-based reach measures, such as 1 plus and higher reach levels, which reflect the proportion of the audience exposed to advertising multiple times [7]. This allows the model to learn directly from audience exposure behavior rather than general media-level aggregates.

Accordingly, this study develops a media mix determination model that integrates supervised machine learning techniques with genetic algorithm-based optimization. The primary focus is on quantitative performance factors. Unlike traditional aggregated media mix models that rely mainly on total spend to outcome relationships, the proposed framework models exposure-based reach metrics at different frequency thresholds. It incorporates measures such as 1 plus and higher exposure levels to capture audience response more precisely. This approach allows the model to reflect nonlinear response behavior and diminishing marginal effectiveness in a more direct manner. To the best of our knowledge, this Exposure-Based Media Mix Modelling framework, which combines exposure level modeling with machine learning and genetic algorithm optimization, represents a novel contribution to media planning research.

The remainder of this study discusses related concepts, defines the research problem and objectives, presents the methodology and results, and concludes with a discussion of implications.

II. Related Concepts

A. Media Mix Modeling

Media Mix Modelling (MMM) is a widely used analytical approach that supports advertisers in allocating budgets across multiple media channels to improve sales performance and

return on investment [8]. It enables the assessment of how individual media channels contribute to overall campaign outcomes within an evolving media environment [8]. Despite its usefulness, advertisers face several challenges when applying MMM, including limited data availability, interactions among media channels, and unpredictable consumer behaviour [9] [8]. To address these challenges, machine learning models such as Random Forest and gradient boosting techniques are increasingly applied. These models can capture complex response patterns by accounting for nonlinear effects, seasonality, and external economic factors [10]. As a result, MMM has become a critical tool for data-driven and cost-effective advertising decision-making [8].

MMM approaches can broadly be categorized into regression-based, Bayesian, and machine learning based frameworks. Traditional regression-based MMM typically relies on parametric specifications that incorporate advertising characteristics such as carryover and diminishing returns through transformations like ad stock and Hill functions [11]. These models offer clear interpretability, as regression coefficients can be directly translated into attribution metrics such as Return on Advertising Spend. However, their performance depends on correct functional form specification and sufficient variation in the underlying data [11].

Bayesian MMM extends regression frameworks by incorporating prior knowledge into the estimation process [12], [11]. Instead of producing single-point estimates, Bayesian approaches generate probability distributions for model parameters and attribution metrics, allowing uncertainty to be explicitly quantified [12]. This can be useful when data are sparse or highly correlated. In contrast, machine learning-based MMM methods emphasize predictive flexibility and nonlinear pattern recognition, often with fewer structural assumptions [10], [11]. While these approaches may reduce reliance on strict parametric forms, they may require additional techniques to ensure interpretability and stable attribution measurement [11].

B. The Concept of Diminishing Returns and Saturation Points in Media Advertising

An important concept in Media Mix Modeling is the principle of diminishing returns. Diminishing returns describe the nonlinear relationship between advertising spend and incremental audience response. As spending increases, each additional unit of investment generates progressively smaller gains in reach or engagement [13]. In empirical media response analysis, this behavior is commonly represented through saturation curves, such as S-curves or Hill transformations, which model how response flattens as spend rises [13].

For example, empirical observations in performance channels show that initial investment levels may produce high marginal returns, while additional increments of spend generate lower incremental gains. A retailer may observe a return on investment of 4:1 at moderate spend levels, declining to 2:1 or lower as spend increases further [13]. This decline reflects audience saturation, budget inefficiency, or limited incremental demand. The flattening of the response curve, therefore, represents the transition from efficient growth to diminishing effectiveness.

Saturation occurs when marginal return falls below a practical decision threshold or below the marginal return of alternative media channels [14]. In mathematical terms, the saturation point corresponds to the region where the first derivative of the response function approaches zero. At this stage, additional spending contributes negligible incremental reach or sales. Identifying this point is essential for rational budget allocation, as continued investment beyond saturation results in inefficient use of resources.

Different media channels exhibit distinct saturation behaviors. Television typically requires higher investment before saturation due to its broad reach potential, whereas digital platforms such as online video may reach saturation earlier because of targeting limits and audience overlap [2]. Moreover, saturation interacts with audience reach and frequency. When a campaign already reaches a high proportion of the addressable audience at multiple exposure levels, further impressions produce minimal incremental response [13].

Recognizing diminishing returns and saturation patterns is therefore fundamental to effective media planning. By estimating response curves and identifying efficiency thresholds, advertisers can avoid over investment in saturated channels and reallocate budgets toward media with remaining growth potential [14]. This empirical understanding forms the analytical foundation for exposure-based modeling and optimization in media mix determination.

C. Machine Learning Applications in Advertising

Machine learning techniques are well-suited for modeling the complex and nonlinear relationships commonly observed in advertising data. Models such as Random Forest, Support Vector Regression, and Bayesian Networks have been used to predict audience reach and identify saturation points across media channels [15]. These techniques also support adaptive decision-making by learning from historical performance data [15]. In addition, machine learning enables dynamic budgeting strategies, allowing advertisers to adjust media investments in response to observed performance changes, thereby improving efficiency and return on investment [16].

D. Audience Reach and Cost Efficiency

Audience reach and cost efficiency are fundamental metrics in advertising evaluation. Reach measures the proportion of the target audience exposed to advertising, while cost efficiency reflects how effectively budgets are converted into audience contact. Common indicators include Cost Per Click, return on investment, and incremental reach [17]. These metrics guide advertisers in reallocating budgets toward higher-performing channels. Prior research also highlights the efficiency gains achieved by shifting budgets from traditional media to digital platforms such as YouTube and display advertising networks, particularly when reach extension and targeting precision are prioritized [18].

E. Audience Exposure and Exposure Level-Based Planning

Traditional reach-based analysis often treats exposure as a binary outcome, without accounting for repeated audience contact. Exposure-based planning extends this concept by measuring reach at different exposure thresholds, such as 1

plus, 2 plus, or higher levels, which represent the proportion of the audience exposed to advertising at least a specified number of times [19]. Exposure frequency plays a critical role in advertising effectiveness, as repeated exposure influences recall, persuasion, and behavioral response. Modeling exposure levels allows advertisers to capture diminishing returns associated with repeated contact and to better understand audience response dynamics [20]. Exposure-based planning has been widely discussed in media planning literature and is commonly applied in television and digital advertising measurement frameworks. However, a detailed review of the theoretical literature on exposure frequency and marginal effectiveness is not included, as it is beyond the scope of this study.

F. Optimization Algorithms for Media Budget Allocation

Optimization techniques translate analytical insights into actionable media plans. Linear Programming and Genetic Algorithms are among the most commonly applied optimization methods in media budget allocation [21]. Linear Programming is effective for static problems with fixed constraints, whereas Genetic Algorithms offer greater flexibility for complex and nonlinear planning scenarios [21]. Genetic Algorithms can adapt to dynamic constraints, nonlinear media responses, and multiple objectives, making them well-suited for exposure-based media planning and budget optimization.

Genetic Algorithms are particularly useful in marketing environments characterized by large search spaces and nonlinear response behaviours [22]. Unlike deterministic allocation rules or simple heuristics, GA-based approaches iteratively evolve candidate solutions through selection, crossover, and mutation processes, allowing efficient exploration of multiple allocation combinations [22]. This makes them suitable for media planning problems where marginal returns differ across exposure levels, and constraints interact dynamically. Therefore, GA was selected as an appropriate optimization framework for handling the nonlinear and multi-constraint structure of the proposed exposure-based media mix model.

Together, these concepts form a comprehensive foundation for understanding and improving media mix determination using modern data-driven and optimization-based approaches.

III. Scope of the Study

This study addresses the problem of developing a media mix determination model for marketing campaigns that can predict optimized budget allocation while maximizing audience reach and maintaining cost efficiency. Advertisers face increasing difficulty in planning budgets across multiple media platforms due to nonlinear response behavior and diminishing returns at higher spend levels. This challenge becomes more complex when audience exposure effects are considered. Therefore, this study focuses on building a quantitative, data-driven framework that integrates exposure-based reach modeling with optimization techniques to support effective media planning decisions.

The objectives of this study are to identify key efficiency characteristics and optimal allocation strategies across media channels under different exposure levels. Specifically, the study aims to identify the maximum efficiency point and the zero efficiency point for each advertising medium across multiple exposure levels, including television, digital platforms, radio, and press. It further seeks to determine the reach and required budget corresponding to selected efficiency levels for each medium and exposure level. In addition, the study aims to estimate the budget and efficiency required to achieve a specified target reach under different exposure conditions. Another objective is to calculate the total achievable reach when all media operate at predetermined efficiency levels for selected exposure combinations. Finally, the study aims to optimize budget allocation across all media under predefined exposure levels in order to maximize total reach and identify the most efficient budget distribution for achieving target audience objectives.

IV. Methodology

A. Data

The data used in this study were provided by Third Shift Media (Pvt) Ltd, a leading advertising and media planning agency in Sri Lanka. The dataset consists of 18 separate datasets covering four main media channels: Television, Digital, Radio, and Press. Digital media data is limited to the Facebook and YouTube platforms. Within the television medium, ten separate datasets are provided. These datasets represent audience exposure levels from 1 plus to 10 plus reach. For Facebook and YouTube, exposure-based datasets corresponding to 1 plus, 4 plus, and 6 plus reach levels are available. These exposure thresholds were selected because they represent commonly applied frequency benchmarks in industry practice for digital campaign planning and performance evaluation. For radio and press, general audience exposure datasets are used without further exposure level separation.

Each dataset contains two variables: spend level and the corresponding audience exposure reach. Spend level represents the advertising expenditure for a given medium and is measured in Sri Lankan Rupees. Audience exposure reach represents the percentage of the total audience reached at a specific budget level for the corresponding exposure definition. This structure allows the analysis to capture how different levels of advertising exposure respond to increasing media spend across platforms.

B. Method of Analysis

The methodological framework of this study begins with a structured data preprocessing phase, as illustrated in Fig. 1. The raw datasets were examined for missing values, duplicate records, and outliers. No such issues were identified, and therefore, no additional data cleaning was required. Each dataset was then sorted by spend level to improve interpretability and ensure consistent model training.

A key preprocessing challenge was the inconsistency in population bases across media platforms. Television, radio, and press datasets were based on a population of 17.6 million. In contrast, Facebook exposure data were based on a population of 8.05 million, while YouTube exposure data reflected a population of 7.47 million. To ensure comparability across all

media, Facebook and YouTube datasets were scaled proportionally to match the 17.6 million population base.

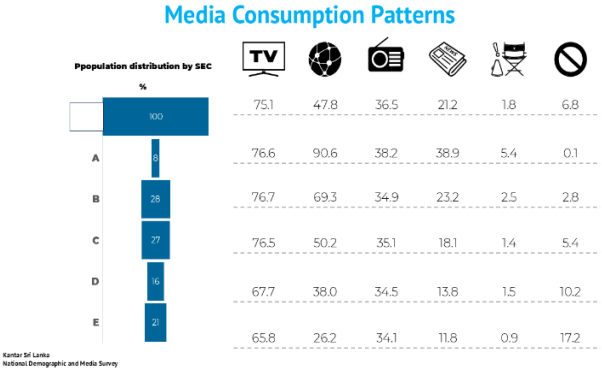


Fig. 1. Media consumption patterns (National Demographic and Media Survey by Kantar Sri Lanka).

The primary objective of the analysis was to identify efficiency levels and saturation points for each medium and exposure level. Since the relationship between advertising spend and audience exposure-based reach is nonlinear and exhibits diminishing returns, four supervised machine learning models were evaluated. These models were Decision Tree, Random Forest, LightGBM, and XGBoost. For each exposure-based dataset across all four media, all four models were trained independently. Model performance was evaluated using Mean Squared Error and R squared metrics on both training and testing sets. The model with the highest R squared value and lowest Mean Squared Error was selected for subsequent analysis.

To support high-resolution analysis, a continuous range of spend values was generated within the observed minimum and maximum spend levels of each dataset. Predicted exposure-based reach values were obtained across this range using the selected model, forming a dense response curve. Since raw predictions could contain minor fluctuations, a Gaussian smoothing filter was applied. This ensured stable derivative estimation in later steps.

The first Numerical derivatives of the smoothed reach curves were computed to quantify marginal reach per unit spend, defined as efficiency. Efficiency values were normalized to a percentage scale, where 100 percent represented the point of maximum marginal reach. Two key thresholds were identified from each efficiency curve. The first was the maximum efficiency point. The second was the saturation point, defined as the point where efficiency fell below 2 per cent of its maximum value, indicating negligible incremental reach.

These efficiency curves enabled two complementary budgeting strategies. In the efficiency-based approach, a user-defined efficiency threshold, such as 80 per cent was selected, and the corresponding budget and reach were identified after the maximum efficiency point. In the reach-based approach, a target reach level was specified, and the corresponding budget and efficiency were derived from the prediction curve.

This process was repeated for all 18 exposure-based datasets across the four media channels. The next phase involved estimating the combined reach under a multi-channel media planning scenario. Media layering order was determined using the National Demographic and Media Survey by Kantar Sri Lanka (Fig. 2). Based on audience penetration levels, media were prioritized as television first, followed by digital platforms in the order of Facebook and YouTube, then radio, and finally press.

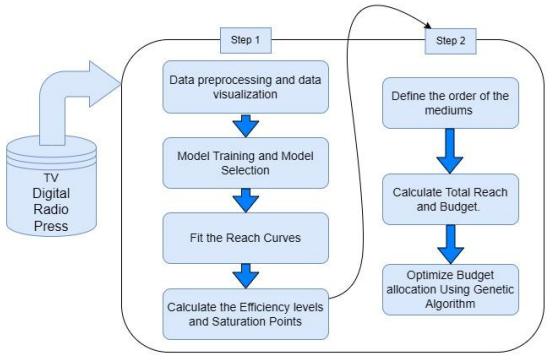


Fig. 2. Research process.

Direct aggregation of reach across channels would lead to duplication errors due to overlapping audiences. To address this issue, realistic duplication assumptions were introduced. It was assumed that 60 per cent of the Facebook audience overlapped with television viewers. 90 per cent of the YouTube audience overlapped with audiences reached through television and Facebook. Similarly, 90 per cent of the radio audience overlapped with audiences reached through television and digital platforms. Finally, 95 per cent of the press audience overlapped with all other media. These assumptions were based on observed media consumption patterns.

To incorporate duplication effects, digital, radio, and press models for all exposure levels were retrained using adjusted datasets that excluded overlapping audiences. Television models were used to estimate the initial reach, followed sequentially by Facebook, YouTube, radio, and press. All modeling steps, including prediction, smoothing, derivative calculation, and efficiency-based analysis, were repeated using the adjusted datasets.

For each medium and exposure level, the budget and reach corresponding to a selected efficiency threshold were extracted. By summing results across media, the total budget required and cumulative reach were estimated. The model also allows users to compute achievable reach under different exposure combinations, such as television 1 plus with Facebook 4 plus and YouTube 4 plus, alongside radio and press at selected efficiency levels.

To further optimize budget allocation under practical constraints, a Genetic Algorithm was applied. The algorithm aimed to maximize total reach while minimizing total spend, or alternatively, achieve a target reach at the lowest possible cost. The initial population consisted of candidate budget plans across selected exposure levels of television, digital, radio, and press, subject to total budget constraints. Each plan was evaluated using trained media models to predict reach. Fitness scores incorporated penalties for exceeding budgets or failing to meet reach targets.

The algorithm evolved solutions over multiple generations, typically between 30 and 50. Selection retained high-performing plans, crossover combined strong solutions, and mutation introduced controlled variation. A repair mechanism ensured all solutions satisfied constraints. Initial populations were seeded using insights from efficiency analysis to improve convergence.

The final output included the optimal budget allocation, predicted reach for each medium, and corresponding efficiency levels. This enabled the Genetic Algorithm to solve different planning scenarios, including fixed budget allocation and target reach achievement.

V. Results

To predict exposure level-specific reach as a function of media spend, machine learning models were selected based on their predictive performance. Decision Tree models were selected for all television exposure level models and for radio, as these achieved the highest R squared values with low Mean Squared Errors. Random Forest models were selected for all digital exposure level models, including Facebook and YouTube, as well as for the press. These models consistently outperformed alternatives across exposure levels.

TABLE I. Results of Selected Machine Learning Models Before Addressing Audience Duplication.

Medium	Selected Model	Frequency	R2	MSE
TV	Decision Tree	1+	0.9853	1.0705
TV	Decision Tree	2+	0.9919	0.8514
TV	Decision Tree	3+	0.9969	0.3978
TV	Decision Tree	4+	0.9983	0.2606
TV	Decision Tree	5+	0.9986	0.2528
TV	Decision Tree	6+	0.9990	0.1913
TV	Decision Tree	7+	0.9992	0.1688
TV	Decision Tree	8+	0.9993	0.1500
TV	Decision Tree	9+	0.9994	0.1433
TV	Decision Tree	10+	0.9994	0.1442
Facebook	Random Forest	1+	0.9999	0.0004
Facebook	Random Forest	4+	0.9999	0.0028
Facebook	Random Forest	6+	0.9999	0.0015
YouTube	Random Forest	1+	0.9998	0.0235
YouTube	Random Forest	4+	0.9999	0.0120
YouTube	Random Forest	6+	0.9999	0.0097
Radio	Decision Tree	-	0.9964	0.1317
Press	Random Forest	-	0.9942	0.0301

Model performance was evaluated using R squared and Mean Squared Error metrics on both training and testing datasets. Results for all exposure level models before addressing audience duplication are summarized in Table I. Performance results after adjusting for duplication are presented for digital, radio, and press models in Table II. Since television served as the base medium, duplication adjustment was not required for television models. To assess potential overfitting, R squared values were calculated separately for training and testing datasets for each model, and the absolute difference between them was computed. Across all exposure level models, this difference remained below a threshold of 0.05. Therefore, no substantial divergence between training and testing performance was observed, indicating stable generalization and a limited risk of overfitting or underfitting in the analyzed advertising datasets.

TABLE II. Results of Selected Machine Learning Models After Addressing Audience Duplication.

Medium	Selected Model	Frequency	R2	MSE
Facebook	Random Forest	1+	0.9999	0.00007
Facebook	Random Forest	4+	0.9999	0.0004
Facebook	Random Forest	6+	0.9999	0.00024
YouTube	Random Forest	1+	0.9998	0.0002
YouTube	Random Forest	4+	0.9999	0.0001
YouTube	Random Forest	6+	0.9999	0.00009
Radio	Decision Tree	-	0.9964	0.0013
Press	Random Forest	-	0.9942	0.0001

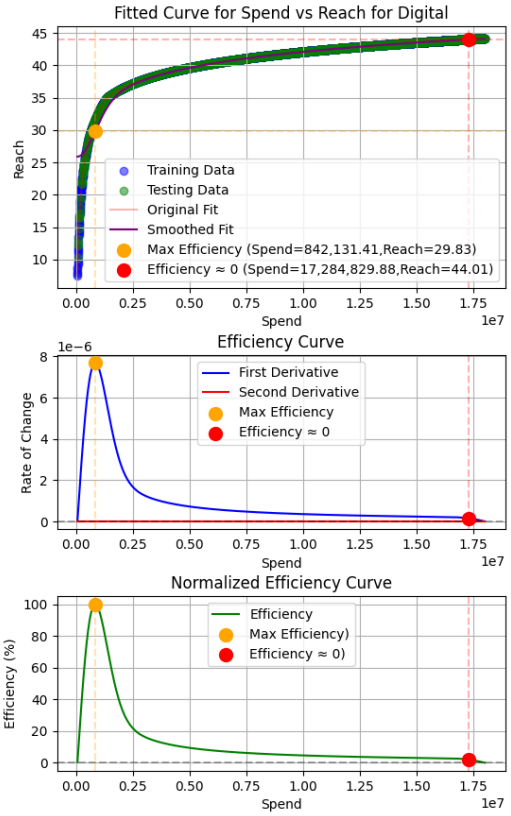


Fig. 3. Reach curve, efficiency curve, and normalized efficiency curve for the Facebook medium at the 1+ exposure level before duplication adjustment.

Based on the selected models, reach response curves were generated for all media and exposure levels under both duplication-adjusted and non-adjusted conditions. For each model, reach curves, efficiency curves, and normalized efficiency curves were derived. Fig. 3 illustrates the model outputs for a representative Facebook digital media case at the 1+ exposure level before addressing duplication. Fig. 4 presents the reach curve after duplication adjustment. The plots highlight the points of maximum efficiency and saturation.

Budget and reach outcomes obtained using both efficiency-based and reach-based targeting approaches, before and after addressing duplication scenarios, are summarized across all exposure levels and media in Table III to Table XX.

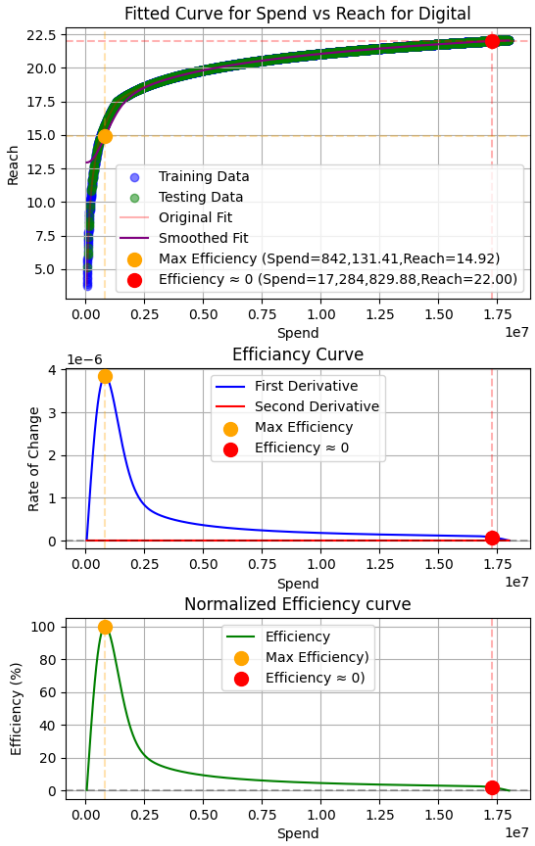


Fig. 4. Reach curve, efficiency curve, and normalized efficiency curve for the Facebook medium at the 1+ exposure level after duplication adjustment.

TABLE III. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 1+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125	74.38
20%	21.90	65.74
40%	14.55	62.53
60%	11.52	60.19
80%	9.35	57.83
100%	6.14	53.18

TABLE IV. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 2+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	74.14
20%	20.96	63.95
40%	14.51	60.20
60%	11.60	57.24
80%	9.45	54.14
100%	6.22	47.96

TABLE V. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 3+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	73.67
20%	20.87	62.10
40%	14.80	57.32
60%	11.89	54.32
80%	9.68	50.50
100%	6.35	42.90

TABLE VI. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 4+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	73.25
20%	21.42	60.67
40%	15.22	55.83
60%	12.20	51.70
80%	9.91	47.26
100%	6.49	38.49

TABLE VII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 5+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	79.98
20%	23.01	60.01
40%	15.88	54.15
60%	12.69	49.52
80%	10.27	44.55
100%	6.69	34.79

TABLE VIII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 6+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	72.58
20%	24.00	58.85
40%	16.59	52.66
60%	13.31	47.78
80%	10.77	42.44
100%	6.98	31.87

TABLE IX. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 7+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	72.13
20%	25.664	58.21
40%	17.330	51.26
60%	13.90	46.14
80%	11.24	40.49
100%	7.26	29.31

TABLE X. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 8+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	71.77
20%	27.31	57.74
40%	18.13	50.20
60%	14.59	44.96
80%	11.79	39.10
100%	7.57	27.41

TABLE XI. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 9+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	71.31
20%	29.12	57.38
40%	19.22	49.34
60%	15.42	43.89
80%	12.49	37.77
100%	7.92	25.62

TABLE XII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE TV 10+ EXPOSURE LEVEL.

Efficiency	Spend (LKR - Mn)	Reach % (Before)
0%	125.00	70.95
20%	30.61	56.87
40%	20.14	48.52
60%	16.16	42.95
80%	13.05	36.76
100%	8.31	24.29

TABLE XIII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE FACEBOOK 1+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	17.99	44.07	22.03
20%	2.62	37.23	18.62
40%	1.87	35.64	17.82
60%	1.54	34.37	17.19
80%	1.27	32.93	16.46
100%	0.84	29.83	14.92

TABLE XIV. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE FACEBOOK 4+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	1.32	29.5	11.80
20%	1.22	29.15	11.66
40%	1.03	27.32	10.93
60%	0.86	24.45	9.78
80%	0.72	21.5	8.6
100%	0.28	8.67	3.47

TABLE XV. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE FACEBOOK 6+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	1.68	28.05	11.22
20%	1.63	27.97	11.19
40%	1.53	27.22	10.89
60%	1.32	24.95	9.98
80%	1.09	21.72	8.69
100%	0.42	8.77	3.51

TABLE XVI. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE YOUTUBE 1+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	23.7	23.3	2.33
20%	6.02	18.3	1.83
40%	3.31	16.6	1.66
60%	2.65	15.8	1.58
80%	2.18	14.9	1.49
100%	1.49	13.3	1.33

TABLE XVII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE YOUTUBE 4+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	23.7	18.3	1.83
20%	5.74	13.7	1.37
40%	3.81	11.8	1.18
60%	3.1	10.5	1.05
80%	2.54	9.1	0.91
100%	1.7	6.2	0.62

TABLE XVIII. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES AT THE YOUTUBE 6+ EXPOSURE LEVEL BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	23.7	17.3	1.73
20%	7.61	13.3	1.33
40%	5.1	11.1	1.11
60%	3.96	9.3	0.93
80%	3.16	7.5	0.75
100%	2.12	4.5	0.45

TABLE XIX. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES FOR RADIO BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	10.00	39.02	3.9
20%	1.95	30.94	3.09
40%	1.13	28.02	2.8
60%	0.89	26.47	2.65
80%	0.73	24.95	2.5
100%	0.48	21.94	2.19

TABLE XX. BUDGET AND REACH FOR EFFICIENCY-BASED AND REACH-BASED TARGETING APPROACHES FOR PRESS BEFORE AND AFTER DUPLICATION SCENARIOS.

Efficiency	Spend (LKR - Mn)	Reach % (Before	Reach % (After)
0%	13.1	27.46	1.37
20%	1.61	23.22	1.16
40%	1.25	22.39	1.12
60%	1.05	21.58	1.08
80%	0.88	20.58	1.03
100%	0.59	18.37	0.92

Using the efficiency curves, total achievable reach and required budget were computed for selected efficiency thresholds. Table XXI presents a planning scenario with efficiency targets of 40 per cent for television 1 plus exposure, 45 per cent for Facebook 4 plus exposure, 50 per cent for YouTube 4 plus exposure, 60 per cent for radio, and 70 per cent for press. These results demonstrate how exposure level selection and efficiency targets influence total reach and budget requirements.

TABLE XXI. COMBINED BUDGET AND REACH OUTCOMES

Model	Efficiency	Spend (LKR)	Reach (%)
TV 1+	40%	14,556,393.14	62.53
Facebook 4+	45%	1,006,975.7	10.76
YouTube 4+	50%	3,427,182.72	1.12
Radio	60%	896,829.68	2.65
Press	70%	974,032.4	1.06
Total		**20,861,413.64**	**78.12**

To further enhance budget allocation decisions, a Genetic Algorithm was applied. In the first scenario, the algorithm optimized a fixed total budget of LKR 20 million across television 1 plus, Facebook 4 plus, YouTube 4 plus, radio, and press. The resulting optimal allocation and predicted reach are reported in Table XXII. In the second scenario, the algorithm was configured to achieve a target reach of 85 per cent for the same exposure level combination. The resulting budget allocation and medium-wise reach outcomes are presented in Table XXIII. These results confirm the effectiveness of the proposed model in supporting exposure-based media planning and optimization.

TABLE XXII. GA RESULTS FOR TARGET BUDGET OF LKR 20 MILLION SPEND.

Model	Efficiency	Spend (LKR)	Reach
TV 1+	16.88%	12,509,660.63	62.049%
Facebook 4+	2.68%	1,347,501.75	11.877%
YouTube 4+	52.21%	2,702,591.28	1.009
Radio	5.97%	2,563,603.05	3.239%
Press	25.71%	876,643.29	1.103
Total		**20,000,000.00**	**79.278%**

TABLE XXIII. GA RESULTS FOR TARGET REACH OF 85% OF TOTAL AUDIENCE.

Model	Efficiency	Spend (LKR)	Reach
TV 1+	4.8%	27,901,264.60	67.52%
Facebook 4+	6.5%	1,310,219.54	11.84%
YouTube 4+	27.4%	3,702,008.25	1.19%
Radio	5.2%	3,036,680.57	3.32%
Press	12.8%	1,080,621.53	1.13%
Total		**37,030,794.49**	**85.00%**

VI. DISCUSSION

This study aimed to develop a practical and data-driven media mix determination model suitable for real-world advertising decision-making using Sri Lankan media data. The proposed framework supports key planning tasks by identifying performance patterns, efficiency levels, and optimal budget allocations across media channels. By incorporating exposure-based reach measures, the model extends traditional media mix modeling approaches and provides a more realistic representation of audience response behavior.

In line with the study objectives, the analysis identified the maximum efficiency points and saturation points for television, digital platforms including Facebook and YouTube, radio, and press across different exposure levels. These results provide valuable insights for advertisers by highlighting spend regions that yield high marginal returns and those that result in inefficient spending. The study further addressed efficiency-based and reach-based planning objectives by estimating the budget required to achieve specified efficiency levels and target reach values under different exposure conditions. This enables advertisers to answer practical planning questions related to expected reach outcomes and budget requirements.

The study also demonstrated how total achievable reach can be estimated by operating all media at predefined efficiency levels for selected exposure combinations. This provides advertisers with clear guidance on how exposure level selection influences cumulative reach outcomes. Budget optimization objectives were addressed through the application of a Genetic Algorithm, which identified optimal budget allocations under constraints such as fixed budgets and target reach levels. The optimization results show that exposure-aware planning can improve budget efficiency and overall campaign reach.

Although the analysis is based on industry data from Sri Lanka, the model is not restricted to a single media plan or organization. The framework can be adapted for advertising budget allocation decision support in small and medium-scale companies as well as larger conglomerates operating in similar market contexts. Overall, the proposed approach offers a simple yet robust tool for exposure-driven media planning and informed budget optimization.

VII. Conclusion and Implications

This study presents a practical and data-driven framework for media mix determination that integrates exposure-based reach modeling with machine learning and genetic algorithm optimization. By focusing on audience exposure levels rather than aggregated channel performance, the proposed approach provides more realistic insights into media response behavior and budget efficiency. The model supports advertisers in identifying efficient spending zones, estimating achievable reach under different exposure conditions, and optimizing budget allocation across multiple media platforms.

Despite its practical contributions, the study has several limitations. The proposed framework focuses exclusively on quantitative performance measures and does not incorporate qualitative factors such as campaign fitness, brand positioning, creative effectiveness, or prevailing market trends. Therefore, advertisers should use the model as a decision support tool rather than a standalone planning solution. Strategic judgment and contextual knowledge remain essential when finalizing media plans.

The model was developed using a duplication-adjusted audience scenario based on observed media consumption patterns. However, audience overlap varies depending on campaign objectives, target segments, and media sequencing strategies. As a result, the duplication parameters applied in this study may not fully reflect all real-world scenarios. In addition, the statistical stability of the results depends on the quality, consistency, and historical depth of the advertising datasets used. The findings may therefore be sensitive to data limitations, exposure definitions, and duplication adjustments implemented during the modeling process.

Future research may integrate qualitative considerations through structured multi-criteria decision-making approaches such as the Analytic Hierarchy Process (AHP). An AHP-based extension could incorporate expert evaluations of qualitative campaign attributes and combine them with the quantitative optimization outputs. This would allow media planning decisions to reflect both measurable performance indicators and strategic brand-level considerations, resulting in a more comprehensive decision framework.

Finally, the model framework is designed in a way that it can be retrained using industry-specific or campaign-specific data, although external validation across different markets has not been conducted within this study. This allows the framework to be adapted for different markets, exposure definitions, and planning objectives. Future extensions may integrate qualitative indicators or dynamic market signals to further enhance decision-making accuracy and practical relevance.

Acknowledgment

The authors would like to acknowledge Third Shift Media (Pvt) Ltd for providing industry data, domain knowledge, and professional support that contributed to this study. The authors also acknowledge the University of Moratuwa for the academic guidance and research environment. Co-authors are gratefully acknowledged for their contributions to this work. In addition, the authors acknowledge the use of ChatGPT version 5.2 for language refinement and clarity improvement during manuscript preparation.

References

[1] "Key Insights from the WFA 2025 Media Budgets Survey," https://www.abintus.consulting/blog/key-insights-from-the-wfa-2025-media-budgets-survey

[2] "Profit Ability 2: The new business case for advertising," https://www.thinkbox.tv/research/thinkbox-research/profit-ability-2-the-new-business-case-for-advertising

[3] M. True, "Optimizing ad spend with AI-powered marketing mix modeling," https://www.fastcompany.com/91250176/optimizing-ad-spend-with-ai-powered-marketing-mix-modeling

[4] Y. Fareniuk and G. Chornous, "Optimization of Media Strategy via Marketing Mix Modeling in Retailing. ," https://www.journals.vu.lt/ekonomika/article/view/27502/30587

[5] J. Browining, "What is Media Mix Modeling (MMM)?," https://www.northbeam.io/blog/media-mix-modeling-mmm-guide

[6] R. Dew, N. Padilla, and A. Shchetkina, "Your MMM is Broken: Identification of Nonlinear and Time-varying Effects in Marketing Mix Models," Aug. 2024

[7] V. R. Lane, "Frequency (Marketing)," https://en.wikipedia.org/wiki/Frequency_%28marketing%29?utm , pp. 80–91, doi: 10.1509/jmkg.64.2.80.17996

[8] Y. Liu, J. Laguna, M. Wright, and H. He, "Media mix modeling – A Monte Carlo simulation study," Journal of Marketing Analytics, vol. 2, no. 3, pp. 173–186, Sep. 2014, doi: 10.1057/jma.2014.3

[9] D. Chan and M. Perry, "Challenges and Opportunities in Media Mix Modeling," https://www.semanticscholar.org/paper/Challenges-and-Opportunities-in-Media-Mix-Modeling-Chan-Perry/c53c3ff6241c31a05d16415c014d4f9ce2cc9d29

[10] S. Kisilevich, "Improving Marketing Mix Modeling Using Machine Learning Approaches | Towards Data Science," https://towardsdatascience.com/improving-marketing-mix-modeling-using-machine-learning-approaches-25ea4cd6994b/

[11] L. Scharfe, J. Antje, and A. Malcherek, "Regression, Bayesian and Machine Learning Methods for Media Mix Modeling-A Comparative Study Motivation and Research Objectives."

[12] K. D. Systems, "Why Bayesian Marketing Mix is Superior to Traditional Approaches?," https://keends.com/blog/why-is-bayesian-regression-modeling-approach-superior-to-traditional-methods/

[13] "Diminishing returns in marketing: when your next euro delivers less than your last - Analytical Alley," https://www.analyticalalley.com/knowledge-hub/diminishing-returns-curve-marketing

[14] M. Lascano, "Saturation Curves in MMM - Indaru," https://www.indaru.com/mmm-saturation-curves/

[15] Y. Li, "Advertising Optimization and Feature Analysis Based on Machine Learning," INSTICC, Aug. 2024, pp. 70–76. doi: 10.5220/0012822000004547

[16] J. A. Choi and K. Lim, "Identifying machine learning techniques for classification of target advertising," Sep. 2020, Korean Institute of Communications Information Sciences. doi: 10.1016/j.icte.2020.04.012

[17] K. Ganesh, C. Thanu Sri, K. Purnendra Reddy, H. Niyotwizeye Gisele, and C. Gloria, "Effectiveness of Online Advertising in Reaching Target Audiences," 2024. [Online]. Available: www.multiresearchjournal.com

[18] Y. Jin, S. Shobowale, J. Koehler, and H. Case, "The Incremental Reach and Cost Efficiency of Online Video Ads over TV Ads."

[19] "Reach and frequency," https://www.thinkbox.tv/how-to-use-tv/spots/reach-and-frequency

[20] G. Broussard, "How advertising frequency can work to build online advertising effectiveness," International Journal of Market Research, vol. 42, pp. 439–457, 2000, doi: 10.1177/147078530004200406

[21] "Webinar: Media Mix Modeling and Optimization," https://www.bayesia.com/bayesialab/tutorials/Webinar-Media-Mix-Modeling-and-optimization

[22] "Genetic Matching Algorithms: Genetic Algorithms in Marketing: Crafting the Perfect Ad Campaign - FasterCapital," https://fastercapital.com/content/Genetic-Matching-Algorithms--Genetic-Algorithms-in-Marketing--Crafting-the-Perfect-Ad-Campaign.html#Success-Stories-in-Genetic-Algorithm-Marketing

An Ensemble Learning Framework with Metaheuristic Optimization for Credit Card Fraud Detection

Agung Nugroho, Muhtajuddin Danny, Ismasari Nawangsih
Dept. of Informatics Engineering, Universitas Pelita Bangsa, Bekasi, Indonesia

***Abstract*—Credit card fraud detection is a major challenge in the financial system due to the characteristics of highly unbalanced data. This study proposes an ensemble learning approach combined with hyperparameter optimization using a Genetic Algorithm to improve the performance of fraud transaction detection. The results of the experiment showed that Random Forest achieved the best performance with a perfect Recall of 1.00 and an F1-Score of 0.903, outperforming the Stacking and Bagging models. Although the optimization significantly increases the training time, this method manages to accelerate the inference time to 0.0290 seconds, making it very feasible to apply to real-time banking security systems that require instant validation. This study confirms the effectiveness of integrating ensemble learning and metaheuristic optimization in dealing with the problem of unbalanced data.**

***Keywords*—Fraud detection; ensemble learning; Genetic Algorithm; Random Forest; real-time detection**

I. Introduction

The rapid increase in digital transactions has changed the global economic landscape, but at the same time, widened the gap for cybercriminal activities. Recent reports show that financial losses due to credit card fraud continue to increase as criminals adapt to conventional security protocols [1][2]. Conventional detection is difficult to rely on anymore because criminals always change their action patterns so that they are unreadable and appear like transactions in general [3]. Accurate early detection becomes crucial, as system failures not only impact direct monetary losses but also erode consumer confidence in the digital banking ecosystem as a whole [4].

The main challenge in detecting credit card fraud lies in the characteristics of extreme class imbalance, where the number of legitimate transactions far exceeds the number of fraudulent transactions with a ratio that can reach 1:1000 or more [5]. In addition, fraud behavior is dynamic (concept drift), where attack patterns are constantly changing to avoid rule-based detection. Traditional machine learning models often fail to address this imbalance, tending to result in high false negative rates that are particularly risky for financial institutions [6].

In recent years, researchers have explored the use of single machine learning models and deep learning to address this problem. However, recent studies show that single models such as Support Vector Machine, Logistic Regression, or Decision Tree often overfit or get stuck in local optima when dealing with high-dimensional data [7] and tend to produce high false negatives when faced with class imbalances between normal and fraudulent transactions [5]. Data imbalance is a major challenge because fraudulent transactions amount to less than 1% of the total data [8].

To overcome the limitations of a single model, the ensemble learning approach has become a prime choice due to its ability to combine the results of multiple algorithms to obtain more stable and accurate classification performance [9]. Approaches such as soft voting and ensemble stacking have been shown to improve detection accuracy and lower false positive rates compared to individual models [10]. The combination of several algorithms, such as Random Forest, AdaBoost, and XGBoost, also provides superior results in handling highly variable transaction data [11].

Nonetheless, the performance of the ensemble model relies heavily on the selection of precise hyperparameters and the selection of relevant features. Searching for parameters manually or using conventional grid search methods has proven to be inefficient and often fails to find the optimal configuration within a large search space [12]. Therefore, metaheuristic optimization algorithms such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Butterfly Optimization Algorithm (BOA) are used to automate the process of finding parameters and selecting the most relevant features [13]. This approach not only improves efficiency but also strengthens the model's ability to deal with high-dimensional datasets [14].

In addition, the integration of data balancing techniques such as SMOTE-ENN and K-means resampling has been proven to improve ensemble learning capabilities in dealing with extreme data imbalances [8]. This approach ensures the model not only focuses on the majority class, but is also able to recognize fraudulent transaction patterns with high accuracy [15]. The combination of metaheuristic optimization and ensemble learning results in models that are adaptive to data dynamics and changes in fraud patterns [16].

This study proposes a new framework that integrates Ensemble Learning with Metaheuristic optimization to detect credit card fraud more precisely. Most previous research has focused on improving accuracy without considering computational efficiency or generalizations on dynamic data [17]. Therefore, this study aims to develop an Ensemble Learning Framework with Metaheuristic Optimization that is able to balance detection performance, efficiency, and adaptability to changing transaction patterns.

II. METHODOLOGY

The methodology proposed in this study is systematically designed to build a robust fraud detection model, address the challenge of data imbalance through the SMOTE technique, and optimize performance through an ensemble stacking approach integrated with metaheuristic optimization. The overall workflow, as visualized in the research flowchart in Fig. 1, consists of five main stages: data collection, preprocessing, unbalanced data handling, model development, and matrix evaluation.

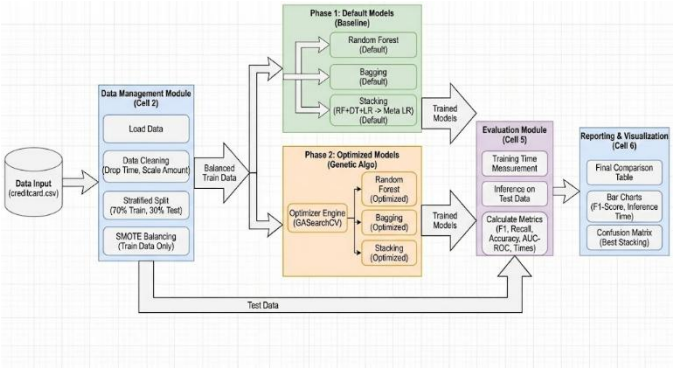


Fig. 1. Research flow.

A. Dataset

The initial stage begins with data input, where the target dataset, i.e., credit card transaction data, is loaded into the analysis environment. This dataset contains credit card transactions that will be the basis for learning for machine learning models. This dataset is sourced from Kaggle, which collected credit card transaction data from cardholders in Europe over two days in September 2013. The data covers a total of 284,807 transactions, of which only 492 transactions were identified as fraud [18]. The characteristics of this dataset are categorized as highly unbalanced due to the significant disparity in proportion between legitimate transactions and fake transactions. This class imbalance is a crucial challenge that is deeply considered in the development of this machine learning model. This dataset was chosen because it has been widely used as a benchmark in many studies on credit card fraud detection, allowing the results of this study to be objectively compared with previous studies [2][19].

B. Data Preprocessing

The preprocessing stage aims to ensure the quality and consistency of data before it is used for model training. First, data cleaning is carried out by removing irrelevant features such as time columns (Time) that do not contribute to fraud patterns, and performing feature scaling using the StandardScaler method to normalize numerical variables such as Amount. After that, the dataset was divided using a stratified split technique with a proportion of 70% training data and 30% test data, to ensure that the distribution of fraud and non-fraud classes remained balanced in each subset [20]. This process is important so that the model does not experience bias against the majority class during training or validation [21].

C. Handling Imbalance Data

One of the most critical challenges in fraud detection is the highly unbalanced nature of data (class imbalance), where the number of fraudulent transactions is much smaller than normal. If left unchecked, the model tends to be biased towards the majority class and fails to detect fraud. To overcome this, the Synthetic Minority Over-sampling Technique (SMOTE) is applied exclusively to the training data only. SMOTE works not by simply duplicating minority data, but by creating new synthetic samples based on interpolation between adjacent minority samples [22]. This approach results in a Balanced Train Data that allows the model to learn the characteristics of fraud more effectively without causing data leakage to the test data. This method helps to expand the search space for fraud patterns without copying data repeatedly [23].

SMOTE helps balance minority class data by synthesizing new samples, but can trigger overfitting if not implemented correctly. In order for the model to have good generalization capabilities, SMOTE is carried out through several stages: calculating the spacing of minority data, determining the percentage of SMOTE and k-value, and finally creating artificial data [24]. Details of this mechanism can be seen in Eq. (1):

$$x_{syn} = x_i + (x_{knn} - x_i) \times \delta \quad (1)$$

where, x_{syn} is the set of synthetic data to be generated, x_i is the data to be replicated, x_{knn} is the data and value closest to the data to be replicated, and δ is a random value between 0 and 1.

D. Model Development

The model development stage utilizes training data that has been balanced and carried out in two experimental phases using the Ensemble Learning approach. The ensemble method is known to improve generalization and robustness of predictions by combining several base models [25].

Phase 1: Default Models (Baseline). This phase aims to build performance benchmarks. The three ensemble architectures of Random Forest, Bagging, and Stacking are trained using the default hyperparameters. Specifically for the Stacking model, the architecture built consists of Random Forest, Decision Tree, and Logistic Regression as base-learners, with additional Logistic Regression serving as meta-learners to aggregate predictions. This approach was chosen because of its ability to combine the advantages of heterogeneous models to produce more stable and accurate predictions than single models [11].

Random Forest – can reduce overfitting by combining multiple decision trees [26][27]. Random Forest is configured with n_estimators=100, random_state=42, and n_jobs=-1. This configuration ensures reproducible results, efficient training, and stable performance in fraud detection.

Bagging Classifier – used to reduce model variance by using bootstrap sampling on the training data. This makes the model more stable and prevents overfitting [28].

Stacking Classifier – a combination of predictions from several base algorithms (base learners) through meta-learners. The purpose of this method is to take advantage of each algorithm so that its performance is more optimal (see Algorithm 1).

Algorithm 1: Stacking Model

Input: Training data after SMOTE balancing
Output: Trained stacking model
1: **Begin**
2: Initialize base learners $B=\{h_1, h_1, \ldots, h_3\}$
3: Train base learners: each h_k is trained on D'_{train} to generate predictions $\hat{y}_k$
4: Construct meta-features: for each sample x_i
$$M(x_i) = [h_1(x_i),\ h_2(x_i), \ldots, h_k(x_i)]$$
5: Train the meta-learner g using the pairs $(M(x_i), y_i)$
6: Final prediction: for a new sample x_j, the stacking prediction is $S(x_j) = g([h_1(x_j),\ h_2(x_j), \ldots, h_k(x_j)])$
7: **End**

Phase 2: Optimized Models (Genetic Algorithm). This phase focuses on improving the performance of the baseline model through hyperparameter optimization. The Optimizer Engine used is based on Genetic Algorithm (GA) through GASearchCV. GA mimics the natural selection process to iteratively evolve a population of parameter combinations to find the optimal configuration that maximizes fitness functions (e.g., F1-Score) for Random Forest, Bagging, and Stacking models.

GA works on the principle of natural selection, where the best combination of parameters is obtained through the process of selection, crossover, and mutation. The fitness function value used is 5-fold cross-validation. This algorithm allows the search for global solutions in complex parameter spaces, resulting in models with better generalizations of new data [16]. The combination of ensemble learning with metaheuristic optimization, such as GA, has been proven to be able to reduce false positive rates without sacrificing inference time significantly [29].

TABLE I. GA HYPERPARAMETER SETTINGS

Parameter	Setting
Population × Generations	6 × 3
Selection	Tournament
Crossover / Mutation	0.8 / 0.1
CV Strategy	5-fold stratified
Fitness	F1-score
Seed	42

Table I presents the detailed configuration of the Genetic Algorithm employed for hyperparameter optimization. The GA operated with a population size of six individuals across three generations, yielding a total of 90 model evaluations when accounting for 5-fold cross-validation. Tournament selection was adopted to maintain competitive pressure among candidate solutions, while crossover (0.8) and mutation (0.1) probabilities were set to balance global exploration and local exploitation of the search space. Elitism ensured that the best solution from each generation was preserved, contributing to rapid convergence as observed in the experimental results. Stratified 5-fold cross-validation was used as the fitness evaluation strategy to maintain class imbalance proportions and reduce overfitting risk. A fixed random seed (42) was applied to guarantee experimental reproducibility.

E. Matrix Evaluation

The final stage is a comprehensive evaluation within the Evaluation Module. It is important to emphasize that all trained models (both from Phase 1 and Phase 2) were evaluated using the original Test Data that was not touched by the SMOTE process. This ensures performance measurements reflect the model's ability to deal with unbalanced real-world data. Evaluation not only measures computational efficiency (training time and inference time), but also performance metrics that are crucial for unbalanced data cases. In addition to Accuracy, the analysis relies heavily on the metrics of Precision, Recall, F1-Score (harmonized average of precision and recall), and Area Under the ROC Curve (AUC-ROC) to provide a complete picture of the model's ability to minimize false negatives (fraud that escapes) and false positives (false alarms). The final results are presented in the form of comparison tables, bar graphs, and confusion matrices. This analysis helps identify the trade-offs between accuracy and efficiency in the implementation of a real-time credit card fraud detection system [30]. All equations can be defined in Eq. (2) to Eq. (5):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} x\ 100 \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$Precission = \frac{TP}{TP+FP} \quad (4)$$

$$F1-score = 2\ \frac{precission * recall}{precission + recall} \quad (5)$$

TP and TN show the amount of positive and negative data that were correctly predicted. Meanwhile, FP and FN described misclassifications in each of these categories. The accuracy metric measures the extent to which target and non-target samples are precisely predicted, reflecting the ability of the test data to represent patterns in the training data [31]. There are two key elements that affect the level of accuracy, namely the amount of data and the imbalance conditions in the dataset used.

III. RESULTS

This study evaluated the performance of ensemble learning, namely Random Forest, Bagging, and Stacking in detecting credit card fraud through comparison of standard conditions and Genetic Algorithm (GA) optimization. Data imbalances were addressed with SMOTE on the training set, while the test data remained in the original distribution to maintain the relevance of reality. The evaluation focused on the Recall and F1-Score metrics in addition to accuracy, precision and AUC-ROC, given the importance of detecting fraudulent transactions on unbalanced data.

The analysis was conducted using public data from Kaggle on European cardholder credit card transactions as of September 2013. The two-day data collection consisted of 284,807 transactions with 492 of them being fraudulent. With a percentage of fraud classes that only reaches 0.172%, this dataset is classified as highly unbalanced. Fig. 2 presents a distribution graph for each of these classes.

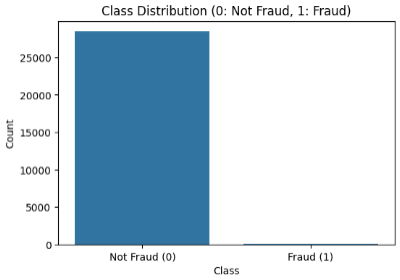


Fig. 2. Class data distribution.

The data is then divided into test data (30%) and training data (70%). In the training data, the SMOTE technique was applied to balance the classes. Initially, out of 199,364 training data, there were only 344 samples of fraud classes. However, after the balancing process, the number of each class (fraudulent and non-fraudulent) became equally large, at 199,020 samples. Details of the comparison before and after SMOTE can be seen in Table II and the graph in Fig. 3.

TABLE II. DATA TRAIN WITH SMOTE

Label (y)	*Row data*	*Row data with SMOTE*
0	199020	199020
1	344	199020

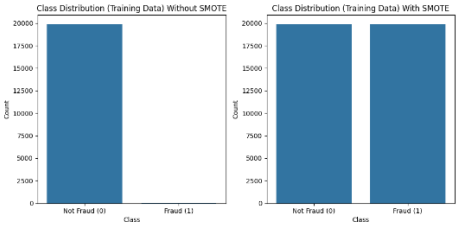


Fig. 3. Comparison data train with SMOTE.

A. Baseline Model Performance

Based on the initial experiment without optimization (baseline), there was a significant difference in performance between the three ensemble algorithms. Referring to Table III, the Stacking model shows the most superior performance with a perfect Precision of 1.00 and an F1-Score of 0.88, which indicates a high ability to minimize false positives. Random Forest (RF) also shows competitive performance with an Accuracy of 0.9995 and a Recall of 0.928, making it a fairly reliable baseline model. In contrast, the Bagging model showed the lowest performance with a Recall of only 0.50 and an F1-Score of 0.48, which indicates its inability to detect half of the total fraud cases that exist.

TABLE III. BASELINE MODEL PERFORMANCE (WITHOUT GENETIC ALGORITHM OPTIMIZATION)

Model	Accuracy	Recall	Precision	F1-Score	AUC-ROC
Random Forest	0.999532	0.928571	0.812500	0.866667	0.999883
Bagging	0.998245	0.500000	0.466667	0.482759	0.783282
Stacking	0.999649	0.785714	1.000000	0.880000	0.999908

B. Optimization of Genetic Algorithms for Model Performance

The application of the Genetic Algorithm aims to find the optimal combination of hyperparameters in each ensemble model. Based on the data in Table IV, the average fitness value for Random Forest increased from 0.9989 in the initial generation to 0.9998 in the 2nd generation with a decreasing standard deviation, indicating that the algorithm successfully converged to the optimal solution. The Random Forest and Stacking models show relatively rapid convergence, with consistent max fitness values since the first generation, while the Bagging model requires a wider variety of parameters before achieving optimal performance.

TABLE IV. RESULTS OF GENETIC ALGORITHM OPTIMIZATION

Model	Generation	Evaluations	Fitness (Mean)	Fitness Std	Fitness Max	Fitness Min
Random Forest (Optimized)	0	6	0.998947	0.00183917	0.999849	0.994836
	1	12	0.999778	0.00005692	0.999849	0.999699
	2	12	0.999841	0.00001184	0.999849	0.999824
Bagging (Optimized)	0	6	0.998746	0.00089185	0.998971	0.998394
	1	12	0.998959	0.00008410	0.999122	0.998846
	2	12	0.999063	0.00008763	0.999197	0.998971
Stacking (Optimized)	0	6	0.999774	0.00000561	0.999824	0.999699
	1	12	0.999820	0.00000936	0.999824	0.999799
	2	12	0.999824	0.00000000	0.999824	0.999824

The application of Genetic Algorithm-based hyperparameter optimization has a significant positive impact, especially on the Random Forest and Stacking models. Based on Table V, Random Forest (Optimized) managed to achieve a perfect Recall of 1.00 (up from 0.928) and an increase in F1-Score to 0.903, making it the best model for detecting all fraudulent transactions without false negatives. The Stacking (Optimized) model also experienced an increase in Recall to 0.857 from the previous 0.785, with a very high AUC-ROC of 0.99995. However, a different phenomenon occurs in the Bagging model, where the optimization actually lowers its performance (the F1-Score drops to 0.387), which indicates that the hyperparameter configuration found by GA may be causing overfitting or not matching the characteristics of the model.

Fig. 4 and Fig. 5 reinforce the numerical results in the comparison table, where Random Forest Optimized shows the most significant F1-Score improvement over other models. The Stacking model also showed consistent improvements, while the Bagging Optimized experienced a decrease in performance. This confirms that the effectiveness of GA optimization is highly dependent on the characteristics of the base model used in the ensemble.

TABLE V. MODEL PERFORMANCE COMPARISON (DEFAULT VS. OPTIMIZED)

Model	Accuracy	Recall	Precision	F1-Score	AUC-ROC
Random Forest	0.999532	0.928571	0.812500	0.866667	0.999883
Random Forest (Optimized)	0.999649	**1.000000**	0.823529	**0.903226**	**0.999908**
Bagging	0.998245	0.500000	0.466667	0.482759	0.783282
Bagging (Optimized)	0.997776	0.428571	0.352941	0.387097	0.783282
Stacking	0.999649	0.785714	**1.000000**	0.880000	0.999908
Stacking (Optimized)	0.999649	0.857143	0.923077	0.888889	**0.999954**

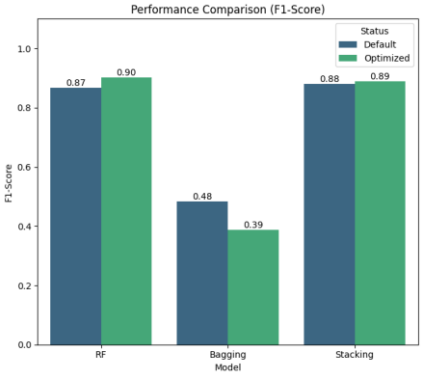


Fig. 4. Performance comparison (F1-score).

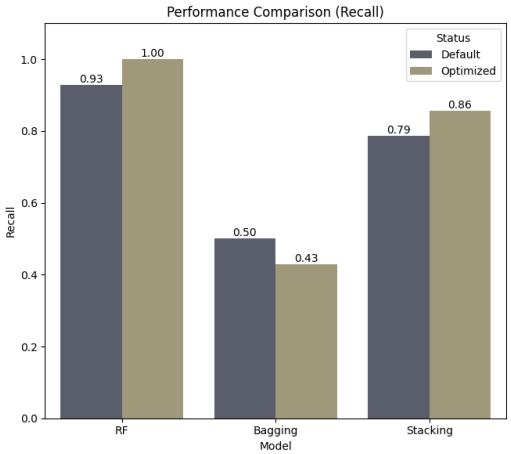


Fig. 5. Performance comparison (Recall).

C. Genetic Algorithm Convergence Analysis

Based on Fig. 6, it can be seen that fitness values increase rapidly in the early generation and then tend to stabilize in the next generation. This pattern shows that the Genetic Algorithm is able to find near-optimal solutions in a relatively small number of generations. The stable convergence also indicates that the GA parameters used have been quite effective in balancing the exploration and exploitation of the solution space.

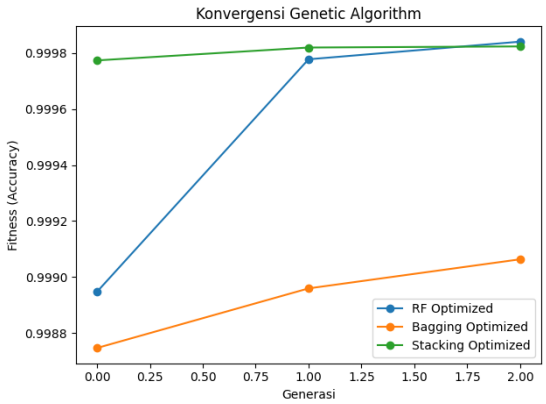


Fig. 6. Genetic Algorithm convergence.

D. Confusion Matrix Analysis and Classification Error Patterns

The confusion matrix analysis in Fig. 7 shows that the optimized Random Forest and Stacking models are able to significantly reduce the number of false negatives, which is a crucial aspect of the fraud detection system. The lack of fraudulent transactions that are misclassified as normal transactions indicates that the optimized model is more sensitive to the minority class. However, there are still a small number of false positives to be aware of, as they can impact the user experience in a real system.

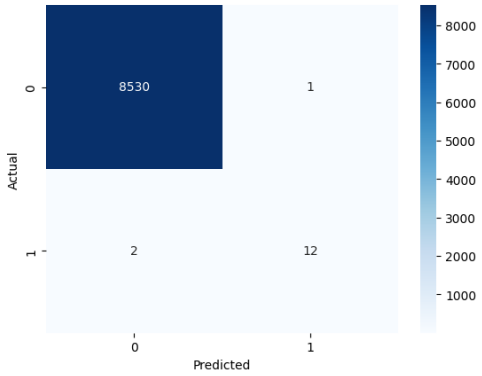


Fig. 7. Confusion matrix.

E. Computational Efficiency and Implementation Feasibility

The results of the computational efficiency evaluation in Table VI and Fig. 8 show that the optimization of the Genetic Algorithm significantly increases training time, especially in Random Forest and Bagging.

TABLE VI. TRAINING TIME AND MODEL INFERENCE

Model	Training Time (s)	Inference Time (s)
Random Forest	21.84	0.0428
Random Forest (Optimized)	434.27	0.0290
Bagging	14.88	**0.0105**
Bagging (Optimized)	1466.00	**0.0104**
Stacking	76.20	0.0691
Stacking (Optimized)	**2288.88**	0.0407

The compute experiment was run on the Google Colab platform with support for an NVIDIA A100 GPU, 16 GB of RAM, and a dual-core Intel Xeon virtual CPU. The entire model is built using the Scikit-learn library with a single-batch inference scheme.

The implementation of GA brings a significant trade-off between training time and inference time. Efficiency data shows a massive spike in training time; for example, Random Forest increased from 21.84 seconds (default) to 434.27 seconds after optimization, and Bagging jumped drastically to 1466 seconds. Nonetheless, it's interesting to note that optimization actually speeds up the time of inference or prediction. The Random Forest inference time is reduced from 0.0428 seconds to 0.0290 seconds, and Stacking (assumed from the context of the efficiency table) also shows a similar efficiency pattern during the prediction phase. This shows that although the initial computational costs (training) are very expensive, the resulting model is more compact and faster when used to detect transactions in real-time.

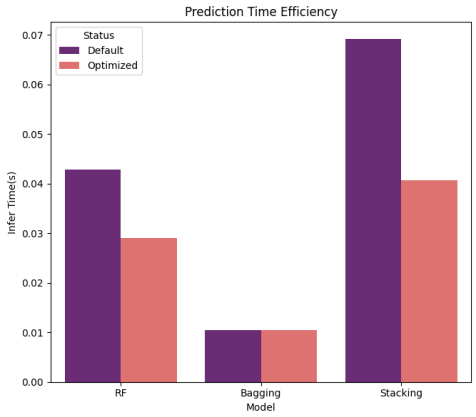


Fig. 8. Inference time.

Overall, the results of the experiment show that the combination of ensemble learning and Genetic Algorithm is effective in improving fraud detection performance, especially in the Random Forest and Stacking models. Improved Recall and F1-Score prove that this approach is capable of overcoming the challenge of data imbalance. However, performance improvements must be considered along with higher computing costs, so the selection of the best model needs to consider the needs of the system and available resources.

This study proves that hyperparameter optimization using a Genetic Algorithm can improve ensemble learning performance in detecting credit card fraud in unbalanced datasets. Random Forest Optimized emerged as the best model with a balance between detection performance and inference efficiency, making it the most viable candidate for practical implementation.

IV. Conclusion

This study concludes that the integration of Ensemble Learning with Genetic Algorithm (GA) optimization is proven to significantly improve fraud detection capabilities, especially in the Random Forest (Optimized) model, which managed to achieve peak performance with a perfect Recall of 1.00 and an F1-Score of 0.903. These results indicate the success of the model in identifying all fraudulent transactions without False Negatives, although the experiment also notes that this metaheuristic optimization has limitations because it is not effective for the entire algorithm, as seen in the performance degradation of the Bag model. From the computational aspect, it was found that there was a strategic trade-off where the use of GA did trigger a spike in training time, but at the same time was able to drastically cut the inference time to only 0.0290 seconds in the Random Forest model. Thus, this combination of maximum detection accuracy and high prediction time efficiency confirms that the model is very feasible and reliable to be implemented into real-time-based banking security systems that demand high speed and precision in the validation of each transaction.

Acknowledgment

This research was carried out with the support of Universitas Pelita Bangsa and the Ministry of Higher Education, Science and Technology in the form of competitive grant funding with contract number 002/7/KP.H/UPB/2025.

References

[1] T. A. Gaav, H. U. Adoga, and T. Moses, "Recent Advances in Credit Card Fraud Detection: An Analytical Review of Frameworks, Methodologies, Datasets, and Challenges," J. Futur. Artif. Intell. Technol., vol. 2, no. 3, pp. 343–369, Sep. 2025, doi: 10.62411/faith.3048-3719-251.

[2] A. R. Khalid, N. Owoh, O. Uthmani, M. Ashawa, J. Osamor, and J. Adejoh, "Enhancing Credit Card Fraud Detection: An Ensemble Machine Learning Approach," Big Data Cogn. Comput., vol. 8, no. 1, p. 6, Jan. 2024, doi: 10.3390/bdcc8010006.

[3] Dipali Sanjay Khedkar and Prof. Rashmi Kulkarni, "Credit Card Fraud Detection using Machine Learning," Int. J. Adv. Res. Sci. Commun. Technol., pp. 1–8, Apr. 2025, doi: 10.48175/IJARSCT-25301.

[4] C.-T. Chen, C. Lee, S.-H. Huang, and W.-C. Peng, "Credit Card Fraud Detection via Intelligent Sampling and Self-supervised Learning," ACM Trans. Intell. Syst. Technol., vol. 15, no. 2, pp. 1–29, Apr. 2024, doi: 10.1145/3641283.

[5] K. K. Mohbey, M. Z. Khan, and A. Indian, "Credit Card Fraud Prediction Using XGBoost," Int. J. Inf. Retr. Res., vol. 12, no. 2, pp. 1–17, Jul. 2022, doi: 10.4018/IJIRR.299940.

[6] S. Sheokand and S. Beniwal, "Hybrid metaheuristic optimization based credit card fraud detection," 2025, p. 050004. doi: 10.1063/5.0299060.

[7] F. K. Alarfaj, I. Malik, H. U. Khan, N. Almusallam, M. Ramzan, and M. Ahmed, "Credit Card Fraud Detection Using State-of-the-Art Machine Learning and Deep Learning Algorithms," IEEE Access, vol. 10, pp. 39700–39715, 2022, doi: 10.1109/ACCESS.2022.3166891.

[8] I. D. Mienye and Y. Sun, "A Deep Learning Ensemble With Data Resampling for Credit Card Fraud Detection," IEEE Access, vol. 11, pp. 30628–30638, 2023, doi: 10.1109/ACCESS.2023.3262020.

[9] M. R. Baker, Z. N. Mahmood, and E. H. Shaker, "Ensemble Learning with Supervised Machine Learning Models to Predict Credit Card Fraud Transactions," Rev. d'Intelligence Artif., vol. 36, no. 4, pp. 509–518, Aug. 2022, doi: 10.18280/ria.360401.

[10] A.-A. Al-Maari, M. Abdulnabi, Y. Nathan, A. Ali, U. Ali, and M. Khan, "Optimized Credit Card Fraud Detection Leveraging Ensemble Machine Learning Methods," Eng. Technol. Appl. Sci. Res., vol. 15, no. 3, pp. 22287–22294, Jun. 2025, doi: 10.48084/etasr.10287.

[11] S. Kodati, N. Nallametti, K. Veeranjaneyulu, N. Sreekanth, B. M. Rao, and G. Jagadesh, "An Ensemble Machine Learning Approach for Enhancing Credit Card Fraud Detection," J. Neonatal Surg., vol. 14, no. 4, pp. 32–40, Mar. 2025, doi: 10.52783/jns.v14.2436.

[12] S. S. Reddy, K. Amrutha, V. MNSSVKR Gupta, K. VSSR Murthy, and V. V. R. M. Rao, "Optimizing Hyperparameters for Credit Card Fraud Detection with Nature-Inspired Metaheuristic Algorithms in Machine

Learning," J. Inst. Eng. Ser. B, vol. 106, no. 6, pp. 2005–2030, Dec. 2025, doi: 10.1007/s40031-025-01207-2.

[13] P. Sure, S. Pandey, T. Parnami, Gaurang, and A. Saxena, "Feature Selection Techniques for Enhancing Credit Card Fraud Detection Performance: A Hybrid Metaheuristic Approach Using Nature-Inspired Algorithms," Int. J. Res. Appl. Sci. Eng. Technol., vol. 12, no. 2, pp. 1185–1194, Feb. 2024, doi: 10.22214/ijraset.2024.58549.

[14] Y. Wang, "A Data Balancing and Ensemble Learning Approach for Credit Card Fraud Detection," in 2025 4th International Symposium on Computer Applications and Information Technology (ISCAIT), IEEE, Mar. 2025, pp. 386–390. doi: 10.1109/ISCAIT64916.2025.11010591.

[15] D. H. M. de Souza and C. J. Bordin, "Ensemble and Mixed Learning Techniques for Credit Card Fraud Detection," pp. 1–7, 2021, [Online]. Available: http://arxiv.org/abs/2112.02627

[16] E. Ileberi and Y. Sun, "A Hybrid Deep Learning Ensemble Model for Credit Card Fraud Detection," IEEE Access, vol. 12, pp. 175829–175838, 2024, doi: 10.1109/ACCESS.2024.3502542.

[17] T. M. Museba and K. V. Vanhoof, "An Adaptive Heterogeneous Ensemble Learning Model for Credit Card Fraud Detection," Adv. Sci. Technol. Eng. Syst. J., vol. 9, no. 3, pp. 1–11, May 2024, doi: 10.25046/aj090301.

[18] "Kaggle." [Online]. Available: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

[19] N. Baisholan, J. E. Dietz, S. Gnatyuk, M. Turdalyuly, E. T. Matson, and K. Baisholanova, "A Systematic Review of Machine Learning in Credit Card Fraud Detection Under Original Class Imbalance," Computers, vol. 14, no. 10, p. 437, Oct. 2025, doi: 10.3390/computers14100437.

[20] M. Azim Mim, N. Majadi, and P. Mazumder, "A soft voting ensemble learning approach for credit card fraud detection," Heliyon, vol. 10, no. 3, p. e25466, Feb. 2024, doi: 10.1016/j.heliyon.2024.e25466.

[21] L. Bonde and A. K. Bichanga, "Improving Credit Card Fraud Detection with Ensemble Deep Learning-Based Models: A Hybrid Approach Using SMOTE-ENN," J. Comput. Theor. Appl., vol. 2, no. 3, pp. 383–394, Feb. 2025, doi: 10.62411/jcta.12021.

[22] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," J. Artif. Intell. Res., vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[23] K. H. Ahmed, S. Axelsson, Y. Li, and A. M. Sagheer, "A credit card fraud detection approach based on ensemble machine learning classifier with hybrid data sampling," Mach. Learn. with Appl., vol. 20, p. 100675, Jun. 2025, doi: 10.1016/j.mlwa.2025.100675.

[24] Asniar, N. U. Maulidevi, and K. Surendro, "SMOTE-LOF for noise identification in imbalanced data classification," J. King Saud Univ. - Comput. Inf. Sci., vol. 34, no. 6, pp. 3413–3423, 2022, doi: 10.1016/j.jksuci.2021.01.014.

[25] A. Mohammed and R. Kora, "A comprehensive review on ensemble deep learning: Opportunities and challenges," J. King Saud Univ. Comput. Inf. Sci., vol. 35, pp. 757–774, 2023, doi: 10.1016/j.jksuci.2023.01.014.

[26] R. Sharma and D. Minhas, "Comparative Performance of Random Forest versus Gradient Boosting Machines in Detecting Financial Fraud," in 2024 7th International Conference on Circuit Power and Computing Technologies (ICCPCT), IEEE, Aug. 2024, pp. 1027–1030. doi: 10.1109/ICCPCT61902.2024.10673140.

[27] S. S. Suganya, S. Nishanth, and D. Mohanadevi, "Ensemble Learning Approaches for Fraud Detection in Financial Transactions," in 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), IEEE, Dec. 2023, pp. 805–810. doi: 10.1109/ICACRS58579.2023.10404382.

[28] L. Yuningsih, G. A. Pradipta, D. Hermawan, P. D. W. Ayu, D. P. Hostiadi, and R. R. Huizen, "IRS-BAG-Integrated Radius-SMOTE Algorithm with Bagging Ensemble Learning Model for Imbalanced Data Set Classification," Emerg. Sci. J., vol. 7, no. 5, pp. 1501–1516, Oct. 2023, doi: 10.28991/ESJ-2023-07-05-04.

[29] S. Iqbal, K. M. Awan, S. Kamal, and Z. U. Rehman, "Interpretable Ensemble Learning Models for Credit Card Fraud Detection," Appl. Sci., vol. 15, no. 22, p. 12073, Nov. 2025, doi: 10.3390/app152212073.

[30] I. Almubark, "Advanced Credit Card Fraud Detection: An Ensemble Learning Using Random Under Sampling and Two-Stage Thresholding," IEEE Access, vol. 12, pp. 192079–192089, 2024, doi: 10.1109/ACCESS.2024.3519335.

[31] M. Gong, "A N OVEL P ERFORMANCE M EASURE FOR M ACHINE," Int. J. Manag. Inf. Technol., vol. 13, no. 1, pp. 1–19, 2021, doi: 10.5121/ijmit.2021.13101.

A Lossless Medical Image Compression Framework Using Logic Minimization

Swathi Pai M[1], Jacob Augustine[2], Pamela Vinitha Eric[3]
Research Scholar-School of Computer Science and Engineering, Presidency University Bengaluru, 560119, Karnataka, India [1]
Department of CSE (AIML), PES University, Bengaluru, 560059, Karnataka, India [2]
School of Computer Science and Engineering, Presidency University Bengaluru, 560119, Karnataka, India [3]

***Abstract*—Medical image compression is an active research area owing to the growth of the volume of medical image data in digital form. A method for lossless compression of medical images was proposed using logic minimization. The grayscale medical image is split into bit planes, and each bit plane is divided into blocks of fixed size, e.g., 8×4. The binary bit stream resulting from each of these blocks is treated as the output of a Boolean function, and logic minimization is attempted for a compact form. If this step fails to give a compact representation, the bits are stored as such. In this study, an extendable framework for lossless compression of medical images is presented. The bit plane is adaptively divided using a Quadtree to capture large uniform areas as leaf nodes. The non-uniform blocks at the leaf node are subjected to a logic minimization approach, as in the case of fixed-size blocks in previous related work. To improve the result further, the original image is gray-coded as a pre-processing step. On the bit plane, an XOR operation is done for the current block with the neighboring block for redundancy removal. This framework allows further exploration with the incorporation of other Boolean function representation techniques to enhance the compression.**

***Keywords*—Medical image compression; lossless compression; Quadtree; logic minimization; bit plane encoding; XOR operation; gray coding**

I. Introduction

Medical images such as X-rays [1], CT scans [2], and MRIs [3] contain critical diagnostic information and therefore must be stored and transmitted without any loss of information [4]. As healthcare systems increasingly rely on digital imaging, the volume of medical data has grown manyfold, creating challenges in storage, network bandwidth, and real-time clinical access. This has seen the lossless compression methods gain prominence in this area since they are used to maintain image fidelity as well as to decrease size and thereby enable efficient telemedicine workflows, long-term archival, and quicker retrieval of information in hospital information systems [5]. Close to perfect compression ratios and at the same time containing no loss is an unsolved research problem, particularly when it comes to heterogeneous medical imagery, with different textures and intensities. This study proposes a lossless framework of medical image compression that builds upon Quadtree-based [6] spatial decomposition together with Boolean logic minimization. The gray coding method is used on the input grayscale image in order to increase the inter-pixel correlation across bit planes [7]. A hierarchical Quadtree structure is then used to process each bit plane, starting with a 64×64-block then a 32×32 block, and then further on 16×16, and then finally an 8×8 block, depending on block uniformity. The leaf-level code produced during this decomposition is coded with the ESPRESSO logic minimizer in order to leverage the redundancy between binary patterns [8]. There are also blocks that fail to minimize well; these blocks are considered to be incompressible. In such situations, to further reduce entropy, an XOR-based prediction step is then used, with each incompressible block being XORed with its spatial neighbor to extract further regularity. The blocks that have been XOR-processed are then minimized with ESPRESSO, and blocks that return compression coded in this manner. The blocks that fail to yield compression shall be coded as incompressible. The suggested hybrid loss-less compression scheme, consist of gray-coding, multi-level Quadtree partitioning, logic minimization, and XOR-based refinement, is a customized scheme for medical images [9]. With the rise in the need for efficient handling of medical data in a modern healthcare setting, this stratified structure will strive to achieve maximum compression ratio, as well as pixel-perfect reconstruction to meet the increasing demands. The compression can also be optimized by incorporating any other methods to the incompressible blocks at the leaf level through this framework.

II. Related Work

A Quadtree is a hierarchical representation or structure, which recurrently divides an image into smaller and smaller parts [10, 11, 12]. Each block is checked at every stage to find out if the pixels of every block are of the same binary value. In such a way, the block turns into a leaf, otherwise it is divided into four equal quadrants, as illustrated in Fig. 1.

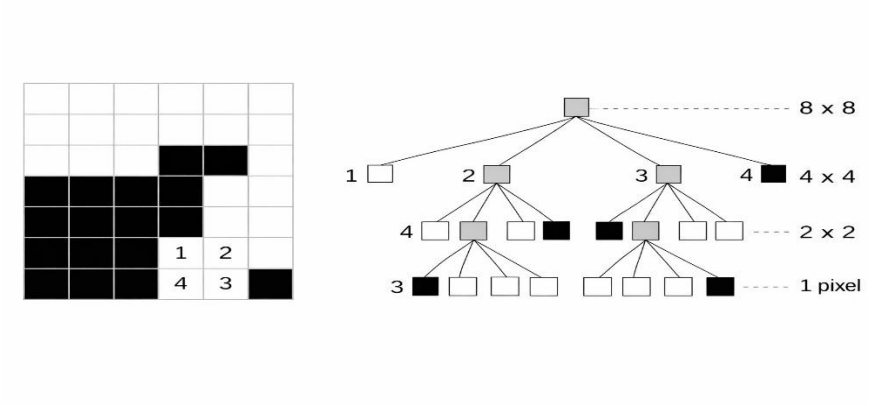


Fig. 1. Quadtree representation.

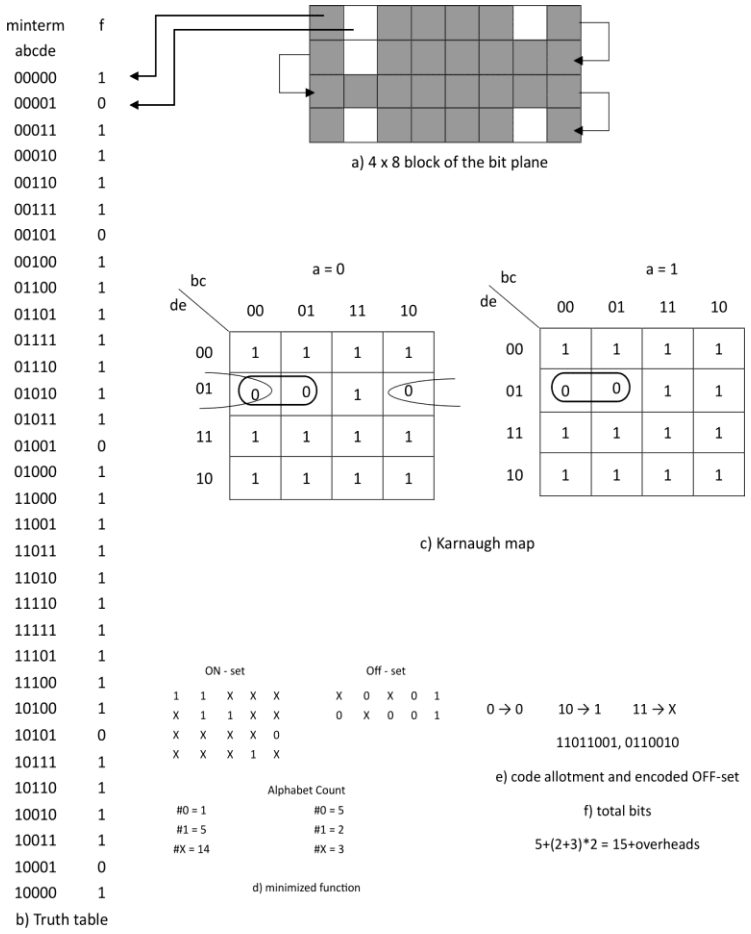


Fig. 2. Function generation and logic coding.

The adaptive representation enables the representation of large homogenous areas by a few nodes, which does not require the storage of redundant pixel data. Thus, for a binary image of size $2^n \times 2^n$, the entire region forms the root of the Quadtree, and subdivisions continue until each leaf contains only 0s or only 1s. Images with large homogeneous areas naturally stop at higher levels of the tree, producing a compact description. In this framework, the grayscale input image (8 bits per pixel) is first separated into eight bit-planes, from the least significant bit s_0 to the most significant bit s_7. Each bit-plane is there- fore a binary image that captures one layer of intensity information.

The Quadtree process is then applied to every bit-plane independently. Initially, each plane is divided into non-overlapping 64 × 64 blocks. This block size is chosen under the assumption that uniform regions are unlikely to extend beyond this scale. Each of these 64 × 64 blocks becomes a Quadtree root and is examined for uniformity. If the block is not homogeneous, it is subdivided to 32 × 32, and subsequently to 16 × 16 and 8 × 8, depending on the spatial variation of pixels [13].

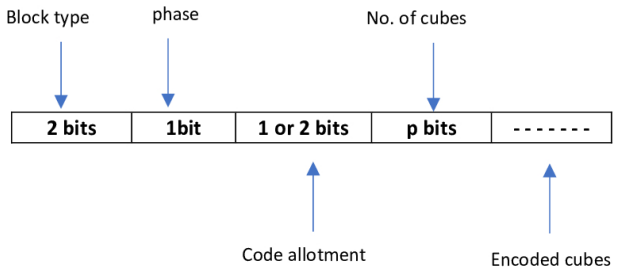


Fig. 3. Format of compressed image.

At the 8×8 stage, a change is incorporated. Rather than the Quadtree process, any non-uniform 8×8 block is further divided vertically to two 8 x 4 blocks. This is refined to capture to vertical structures and narrow column-oriented features that are seen in medical images. In lower bit-planes (s0–s3), these 8 × 4 regions often contain fine-grain details resembling noise, while in the higher bit-planes (s4–s7), they tend to preserve strong anatomical edges [14]. This stage prepares the binary data for logic minimization more effectively. Each 64 × 64 region is tested to ensure the Quadtree-based representation is actually beneficial. If the compressed representation produced by the Quadtree requires more bits than simply storing the block row-wise, the original block is kept instead. A single flag bit is stored to indicate whether Quadtree coding was used:

- 0 – the block is compressed using Quadtree rules
- 1 – the block is stored directly as raw pixels

Two bits are used at each internal node of the Quadtree to indicate the nature of the region:

- 00 – region is entirely black
- 01 – region is entirely white
- 10 – region requires further subdivision
- 11 – leaf node (special handling begins at the 8 × 8 level)

For a mixed region, subdivision continues until the 8 × 8 scale is reached. At this point, the region is split into two 8 × 4 blocks, and each 8 × 4 block is classified using two bits:

- 00 – block is all black
- 01 – block is all white
- 11 – block contains mixed pixels and must be encoded

For mixed 8 × 4 blocks, the corresponding 32-bit pattern is encoded using Boolean logic minimization through the ESPRESSO algorithm. If ESPRESSO produces a representation shorter than the raw block, the minimized form is stored and marked using:

- 0 – logic coding successful

If the minimizer is unable to reduce the block size, raw pixel data is stored instead:

- 1 – logic coding unsuccessful; retain original bits

Such a combination of multi-level Quadtree decomposition, non-uniform 8x8 areas being treated with special care and Boolean minimization allows the representation of structured and unstructured regions in the image to be made efficient and ensure that compression is no higher than the cost of storing the raw block.

A. Logic Coding

Jacob et al. [15] proposed an extension of the block-based binary image coding to the use of logic minimization principles. A grayscale image is split in their method into single bit-planes and each bit-plane is divided into blocks n x m size, both dimensions being powers of two [16]. Each block would belong to one of three groups, namely, all-black, all-white, and interracial. Short predetermined codes are assigned to uniform blocks, whereas mixed blocks get translated into switching functions and simplified using ESPRESSO logic minimizer [17]. The technique makes sure that only minimization causes compression of the number of bits; in case the original block cannot be compressed, the original block is kept [18].

The pixel positions are mapped to minterms by a process known as gray coding, such that adjacent pixels in the image can be translated to adjacent ones in the truth table [19]. This neighboring adds the probability of cube combining in the entity of minimization. In order to maintain this adjacency between rows, the algorithm switches forward and reverse scanning of rows [Fig. 2(a)]. Logic minimization [20] is sometimes capable of encoding a block of 32 pixels with a significantly smaller number of bits than necessary, e.g., 32 bits to around 15 bits, together with any overhead. The conversion of a 4x8 block into a five-variable switching function, the associated Karnaugh map groupings [21], and the number of alphabets in an ON-set and OFF-set cube, respectively, are shown in Fig. 2 [22]. There is always a frequency of cube occurrences and, based on these, prefix codes like 0, 10, and 11 are given to cube types [Fig. 2(e)]. The format of encoding of every block of bit-planes is shown in Fig. 3. A two-bit header identifies the block type:

- 00 for all-black blocks
- 01 for all-white blocks
- 10 for blocks successfully minimized
- 11 for blocks where minimization fails

In the case of incompressible blocks (type 11), original nm bits are simply stored at the end of the header, [23]. Another bit is used to show whether the best ON-set or OFF-set representation has been chosen in compressible blocks. A further code allocation step (summarized in Table I) uses a p-bit field, whose size varies from 1 to 6 depending on the block's minimization characteristics.

TABLE. I. PREFIX CODE ALLOTMENT TO SYMBOLS

Allotment Indicator	Code Allotment		
	0	1	X
0	0	10	11
10	11	0	10
11	10	11	0

The pixel values are replicated through an annual

assessment of the minimized cubes on all minterms of the block during decoding. One of the pixels is given a one in the end of a minterm held by any cube in the ON-set; a zero in the end of a minterm held by the OFF-set, or at the end of a minterm not subsumed by any RAM-ON cube. This subsuming cube- based reconstruction approach guarantees that the representation by the compression creates the binary framework of the block accurately (once the binary structure) [24].

III. Proposed Methodology

The suggested approach (illustrated in Fig. 4) starts with changing the input grayscale picture into its Gray-coded counterpart, such that adjoining intensity levels are a single step apart [25]. The changes make the binary transitions seen within the bit-planes more continuous to allow an even more compressible representation. The source pixel addresses with adjacent pixels represented by one bit difference are mapped to pixel positions using gray coding to allow those pixels to be packed in a narrow band together. The significance of this property is that it conserves the locality of space when a pixel block is viewed as a Boolean function [26]. As an example, a pixel value of 120 (binary 01111000) gets converted to gray code 01000100, the neighboring value 121 (binary 01111001) becomes 01000101, indicating that there is a change in the value of only a single bit between neighboring pixels. The Gray-coded image is then decomposed into eight bit-planes, each containing the 0^{th} to 7^{th} bit of all pixels. Since each bit-plane is now a binary image, further processing is performed independently on every plane. To capture spatial uniformity effectively, each bit-plane is divided into non- overlapping blocks of size 64×64. Every block serves as the root of a quadtree and is recursively split into 32 × 32, 16 × 16, and finally 8 × 8 regions. The splitting of an image recursively is shown in Fig. 5.

A block becomes a leaf whenever its pixels are uniform; otherwise, subdivision continues. This hierarchical partitioning concentrates detailed processing only on non-uniform regions while compactly representing large homogeneous areas. Once the 8 × 8 level is reached, the block is considered the fundamental unit for logic-based encoding.

Each non-uniform block at the finest level is treated as a Boolean switching function whose output corresponds to the binary values of the $n \times m$ pixels in the block [28]. The block is mapped to a function of $\log_2(nm)$ variables, and the widely used ESPRESSO logic minimizer is employed to reduce the number of cubes required to represent this function. Blocks for which minimization yields a reduction in bit count are stored in compressed form, while the remaining ones are labeled as incompressible and preserved in their original pixel format.

To further reduce the number of blocks that remain incompressible after the first round of logic minimization, an XOR-based refinement step shown in Fig. 6 is applied. This step uses a left-predictor model, where each block is predicted using its immediate left neighbor. Let $B(i,j)$ be the current block in row i and column j, and $B(i, j-1)$ be the block to its left. The residual block is computed as:

$$R(i,j) = B(i,j) \oplus B(i,j-1) \quad (1)$$

The first block in every row, $B(i, 0)$, has no left neighbor; therefore, prediction cannot be performed and the block is retained in its original form [29]. When neighboring blocks share similar structures, which is typical in medical images, the XOR operation cancels out the common patterns and produces a residual with a larger number of zeros. For example, consider a left block L and the current block C from a bit plane:

L = 10110110

C = 10100111

The XOR residual R is obtained as

$$R = L \oplus C = 00010001.$$

The resulting residual block contains long runs of zeros, which makes it more suitable for subsequent minimization using the logic-based compression stage. The residual block $R(i, j)$ is therefore passed through ESPRESSO for a second minimization attempt. In the proposed Scheme, heterogeneous leaf blocks are encoded using a compact header structure to indicate the applied transformations. For leaf blocks of size 8 × 8, a one-bit XOR header is first used to signal whether left predictive XOR has been applied between the current block and its immediate left neighbor, and logically minimized.

- 1: indicates that the XOR transformation is performed
- 0: denotes that the original block values are retained

After the XOR operation, the resulting block is subjected to Boolean logic minimization [27] using the ESPRESSO algorithm. After that, a one-bit storage type header is employed to show the state of whether the block is held as raw pixels or as minimized Boolean cubes.

- 1: represents raw storage
- 0: denotes logic-minimized representation

This combined encoding of Gray encoding, adaptive Quadtree partitioning, and separable steps of logic minimization with refinement of XOR is to improve compressibility without any relationships to the image being altered during the encoding process to maintain the lossless input of the reconstructed image.

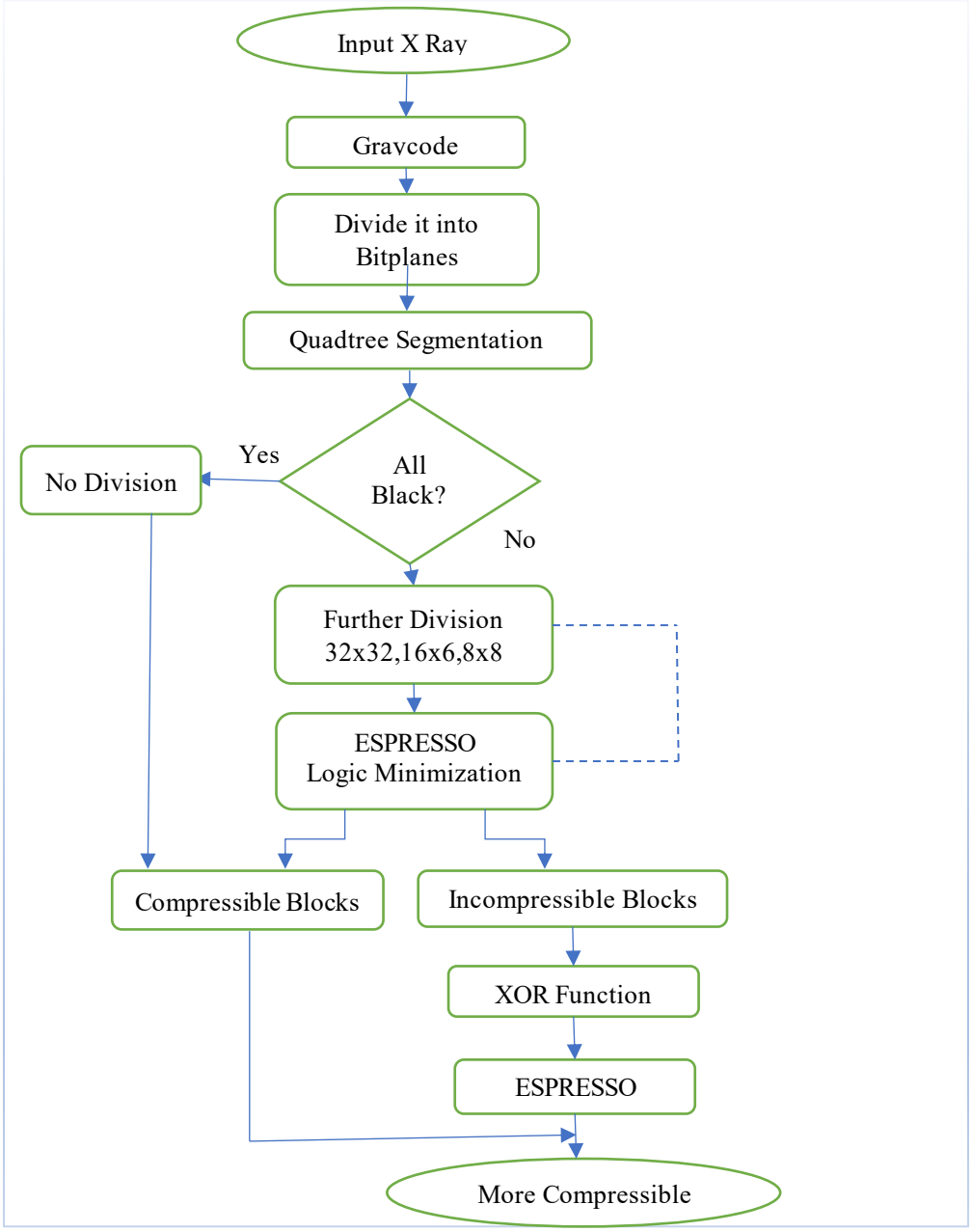


Fig. 4. Flowchart for the related work.

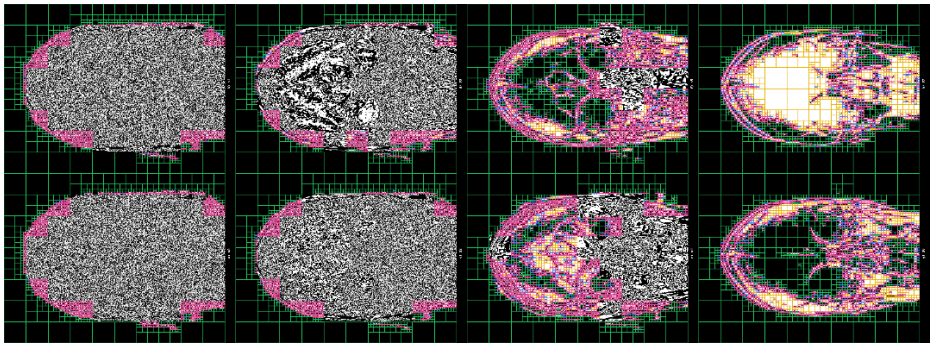


Fig. 5. Recursive Quadtree division.

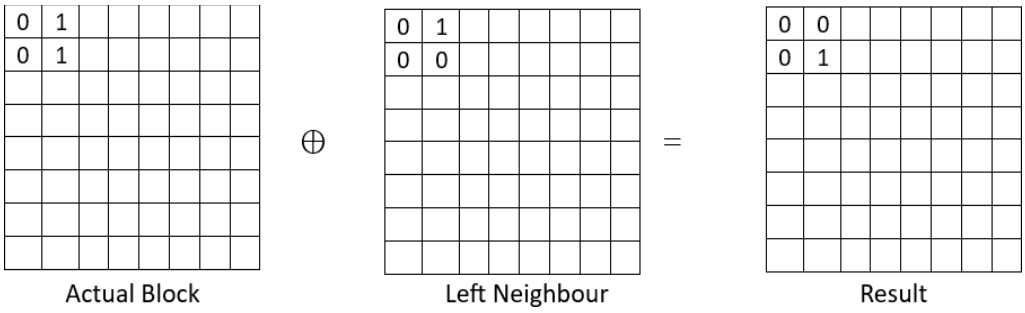


Fig. 6. XOR function.

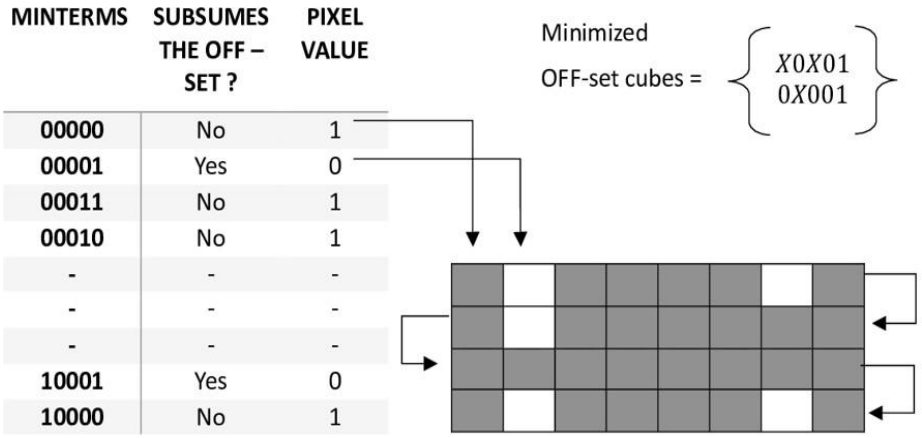


Fig. 7. Decompression scheme.

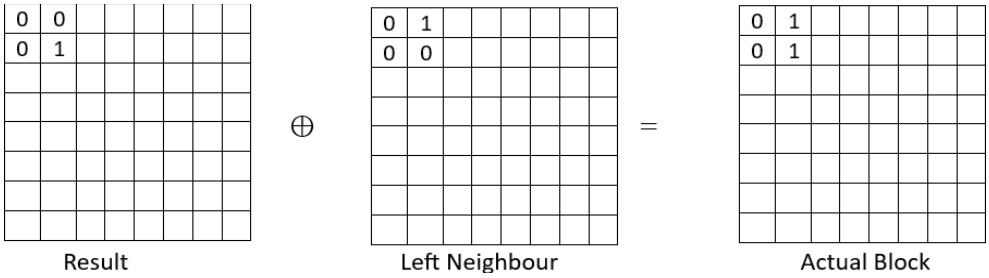


Fig. 8. Reverse XOR function.

A. Decompression Process

During decompression, each bit plane is reconstructed by decoding the compressed blocks in the same sequence used during encoding [29]. For blocks classified as all-black, all-white, or incompressible, recovery is direct since their pixel values were stored without applying logic minimization. For logic-coded blocks, the minimized switching function associated with the block is first rebuilt. The encoded data provide the block type, the phase bit, and the symbol-to-code mapping, from which the minimized ON-set or OFF-set cubes are recovered. Once these cubes are obtained, the switching function is expanded back to its truth-table form using cube subsuming: each minterm generated in Gray-code order is checked against the recovered cubes. A minterm that is contained within any cube of the ON-set is assigned a pixel value of 1, whereas a minterm subsumed by a cube of the OFF-set is assigned 0. Minterms not covered by the minimized cubes receive the complementary value. These values are then placed in their respective positions in the block, producing the reconstructed binary region and is depicted in Fig. 7. For blocks that underwent XOR-based refinement during compression, an additional reversal step is applied, as shown in Fig. 8. Because XOR is self-inverting, the original block C_i is reconstructed by XOR-ing the decoded residual block $R\hat{}i$ with its left predictor Li:

$$Ci = Li \oplus R\hat{}i \qquad (2)$$

Only the first block of each row omits this step, as it was stored directly during compression [30]. By combining logic-coded reconstruction with XOR reversal, each bit plane is recovered exactly as in the original grayscale image.

IV. Results and Discussion

The experimental evaluation was carried out on grayscale X-ray images (shown in Fig. 9) to examine the compression performance of the proposed multi-stage coding scheme which was carried out using Python version *3.13.3* on a Linux platform. Various images which are used here are collected

from Kaggle.com.

The compression efficiency was evaluated using the compression ratio, defined as:

$$Compression\ Ratio = (Size\ of\ original\ image)/(Size\ of\ Compressed\ image) \quad (3)$$

The compression ratios reported in this section, as shown in Table II, reflect the pure logic coding technique [15]. Adaptive segmentation technique [32] and cumulative effect of Gray coding [33] adaptive block partitioning, logic minimization, and XOR-based residual processing is compared with standard PVRG-JPEG Lossless [34] compression technique available.

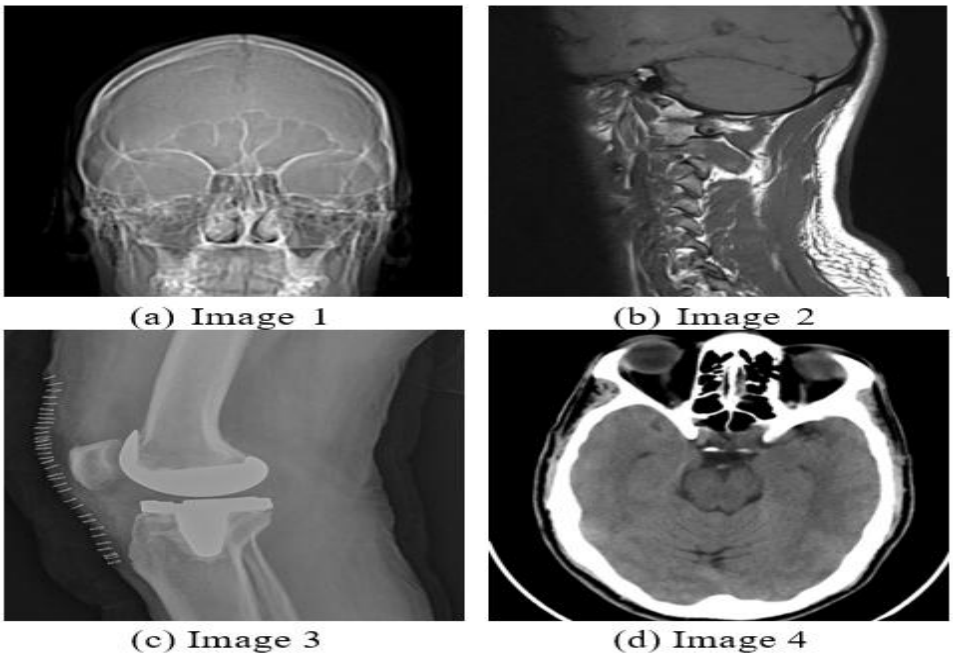


Fig. 9. X-Ray images used for compression.

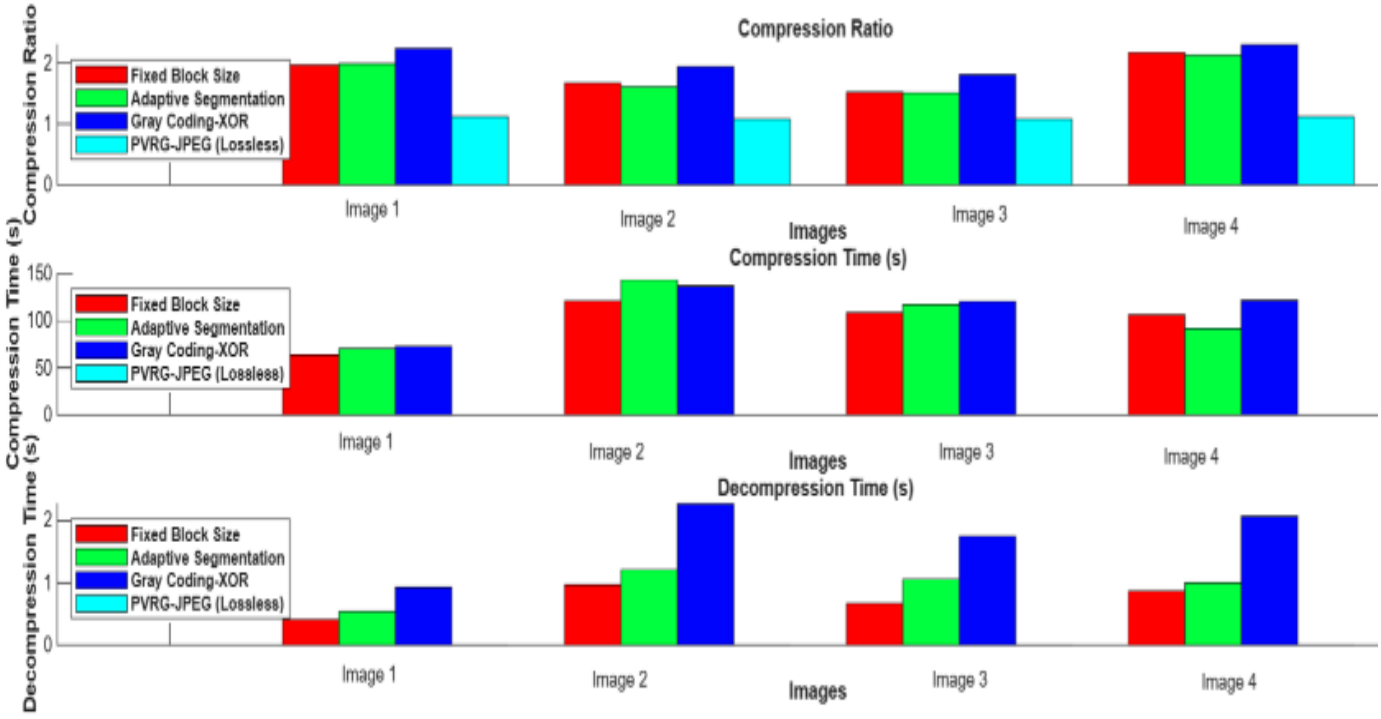


Fig. 10. Comparison of compression efficiency, time, and decompression time for different compression methods.

TABLE. II. SUMMARY OF COMPRESSION RESULTS

Image	Image 1	Image 2	Image 3	Image 4
Logic Coding with Fixed Block Size				
Compression Ratio	1.97	1.67	1.52	2.17
Compression Time (s)	63.8	121.2	108.8	106.5
Decompression Time (s)	0.41	0.97	0.67	0.87
Logic Coding with Adaptive Segmentation				
Compression Ratio	1.99	1.61	1.51	2.13
Compression Time (s)	70.7	143.3	117.3	91.58
Decompression Time (s)	0.54	1.21	1.06	1.00
Gray Coding–XOR Technique				
Compression Ratio	2.24	1.94	1.81	2.30
Compression Time (s)	72.90	137.04	120.60	121.71
Decompression Time (s)	0.93	2.27	1.75	2.07
PVRG-JPEG (Lossless)				
Compression Ratio	1.12	1.08	1.08	1.12
Compression Time (s)	0.02	0.03	0.03	0.03
Decompression Time (s)	0.001	0.001	0.001	0.01

For bit-plane *s*6 of image 1 (as shown in Table III), the Quadtree analysis at the 64 × 64 level shows that 32.50% of blocks are all-black, while no all-white blocks are observed, indicating the presence of large uniform dark regions but absence of bright uniform regions at this scale. The remaining 67.50% of blocks are mixed and therefore require further Quadtree subdivision. At the 32 × 32 level, the percentage of all-black blocks reduces to 18.52%, and all-white blocks increase slightly to 2.56%, while 81.48% of blocks remain mixed, suggesting that significant structural variations still persist. When the block size is reduced to 16 × 16, the proportion of homogeneous blocks in- creases to 16.76% all-black and 2.56% all-white, and the mixed blocks reduce to 80.68%, indicating gradual isolation of uniform regions. At the final 8 × 8 level, 15.67% blocks are all-black and 8.80% are all-white, while a dominant 75.53% blocks become leaf nodes and are forwarded to the logic minimization stage. After applying the ESPRESSO logic minimizer on these 8 × 8 leaf blocks, only 43.63% of blocks are identified as compressible, whereas 56.37% remain incompressible, showing that despite spatial uniformity, logical redundancy is insufficient in many blocks of this higher bit-plane. To further improve compression, left-neighbor predictive XOR coding is applied to these incompressible blocks, followed by a second pass of logic minimization. After XOR, the compressible block percentage becomes 5.53%, while 94.47% blocks still remain incompressible, indicating that although XOR introduces some decorrelation [31] and simplification, the dominant edge and anatomical structures present in the s6 plane limit further Boolean simplification [35]. This behavior is expected for higher bit-planes, where major structural information is preserved, making them less amenable to logic-based compression even after predictive coding. Table III to Table VI shows the statistics of all four images. Fig. 10 compares four approaches to compressing an image: Fixed Block Size, Adaptive Segmentation, Gray Coding-XOR, and PVRG-JPEG (Lossless). The figure is made up of three subplots, which show different performance measurements of each method. The first subplot shows the compression ratio that shows the efficiency of each method in reducing the image size. An increased compression ratio implies increased compression efficiency. The second subplot depicts the compression time which is the duration every method requires to compress the images another important aspect when comparing the speeds of compression techniques. The third subplot revolves around the decompression time as it shows the time needed to restore the images back to their original state. This ratio is most suitable when the application requires details that are real-time. In combination, these subplots present the in-depth evaluation of the compression effectiveness of each approach, processing speed, and decompression effectiveness of each approach, which is of great importance in terms of identifying which one to use in each practice.

TABLE. III. OVERALL STATISTICS FOR IMAGE 1 (PER CENT VALUES)

Block Level	Code Type	s0	s1	s2	s3	s4	s5	s6	s7
	All Black	17.50	17.50	17.50	20.00	25.00	26.25	32.50	37.50
64x64	All White	0.00	0.00	0.00	0.00	0.00	0.00	0.00	5.00
	Quadtree	82.50	82.50	82.50	80.00	75.00	73.75	67.50	57.50

	All Black	13.64	13.64	13.64	12.11	10.00	14.83	18.52	7.07
32x32	All White	0.00	0.00	0.00	0.00	0.00	0.50	2.56	6.52
	Quadtree	86.36	86.36	86.36	87.89	90.00	85.17	81.48	86.41
	All Black	7.13	7.35	7.57	7.33	6.94	14.30	16.76	11.79
16x16	All White	0.00	0.00	0.00	0.00	0.12	0.50	2.56	10.22
	Quadtree	92.87	92.65	92.43	92.67	92.94	85.20	80.68	77.99
	All Black	3.63	4.11	4.33	4.62	8.19	13.83	15.67	14.77
8x8	All White Leaf Node	0.00 96.37	0.00 95.89	0.00 95.67	0.27 95.11	3.21 88.61	3.61 82.55	8.80 75.53	14.82 70.41
	Compressible Blocks	0.77	1.51	7.47	16.74	23.21	35.83	43.63	40.87
	Incompressible Blocks	99.23	98.49	92.53	83.26	76.79	64.17	56.37	59.13
Post-XOR Logic Minimization Applied to the above Incompressible Blocks									
Compressible Blocks after XOR		0.03	0.06	0.20	1.04	1.53	2.28	5.53	3.39
Incompressible Blocks after XOR		99.97	99.94	99.80	98.96	98.47	97.72	94.47	96.61

For the compressible blocks, the distribution of ON-set and OFF-set cubes is computed. Between the two, the smaller set is chosen for encoding to ensure a more compact Boolean representation. This adaptive selection contributes to improved compression efficiency without increasing reconstruction complexity. Although in higher bit planes, pixel values mainly represent sharp edges and major structural boundaries. Applying left-predictive XOR across such edges produces irregular residual patterns, which reduces block uniformity and limits the scope for Boolean minimization. As a result, some MSB planes show a slight negative gain in compression after XOR.

TABLE. IV. QUADTREE STATISTICS FOR IMAGE 2 (PER CENT VALUES)

Block Level	Code Type	s0	s1	s2	s3	s4	s5	s6	s7
	All Black	0.00	0.00	0.00	0.00	0.00	26.52	31.06	51.52
64x64	All White	0.00	0.00	0.00	0.00	1.52	0.00	6.06	0.00
	Quadtree	100.0	100.0	100.0	100.0	98.48	73.48	62.88	48.48
	All Black	0.00	0.00	0.00	4.17	0.00	7.99	8.73	26.56
32x32	All White	0.00	0.00	0.00	0.00	6.35	0.26	9.34	0.00
	Quadtree	100.0	100.0	100.0	95.83	93.65	91.75	81.93	73.44
	All Black	0.00	0.00	0.33	7.61	1.28	12.92	6.43	30.32
16x16	All White	0.05	0.57	0.85	1.14	8.83	1.47	16.64	4.52
	Quadtree	99.95	99.43	98.82	91.25	89.89	85.60	76.93	65.16
	All Black	0.14	0.98	5.43	8.34	5.14	12.65	8.81	32.24
8x8	All White Leaf Node	1.08 98.78	2.31 96.71	2.43 92.14	4.62 87.05	12.11 82.75	8.31 79.04	18.76 72.43	11.28 56.48
	Compressible Blocks	9.26	14.55	20.14	21.33	31.99	34.85	43.15	42.91
	Incompressible Blocks	90.74	85.45	79.86	78.67	68.01	65.15	56.85	57.09
Post-XOR Logic Minimization Applied to the above Incompressible Blocks									
Compressible Blocks after XOR		0.23	0.41	0.63	1.35	1.39	2.25	2.64	3.96
Incompressible Blocks after XOR		99.77	99.59	99.37	98.65	98.61	97.75	97.36	96.04

TABLE. V. QUADTREE STATISTICS FOR IMAGE 3 (PER CENT VALUES)

Block Level	Code Type	s0	s1	s2	s3	s4	s5	s6	s7
	All Black	0.00	0.00	0.00	0.00	0.00	0.93	14.81	38.89
64x64	All White	0.00	0.00	0.00	0.00	0.93	3.70	8.33	0.00
	Quadtree	100.0	100.0	100.0	100.0	99.07	95.37	76.85	61.11
	All Black	0.00	0.00	0.00	0.00	0.00	2.91	16.57	28.79
32x32	All White	0.00	0.00	0.00	0.00	1.87	12.14	20.18	7.20
	Quadtree	100.0	100.0	100.0	100.0	98.13	84.95	63.25	64.02

	All Black	0.00	0.00	0.00	0.29	3.33	6.29	14.88	23.22
16x16	All White	0.00	0.00	0.00	0.00	5.48	12.50	19.52	9.76
	Quadtree	100.0	100.0	100.0	99.71	91.19	81.21	65.60	67.01
	All Black	0.00	0.00	0.06	2.18	6.61	12.95	16.61	25.39
8x8	All White Leaf Node	0.00 100.0	0.00 100.0	0.07 99.87	2.23 95.59	10.51 82.88	11.81 75.24	20.05 63.34	12.36 62.25
	Compressible Blocks	0.06	3.33	20.53	38.20	41.99	42.37	38.42	49.02
	Incompressible Blocks	99.94	96.67	79.47	61.80	58.01	57.63	61.58	50.98
Post-XOR Logic Minimization Applied to the above Incompressible Blocks									
Compressible Blocks after XOR		0.00	0.06	0.08	0.48	0.73	1.13	1.55	3.13
Incompressible Blocks after XOR		100.0	99.94	99.92	99.52	99.27	98.87	98.45	96.87

TABLE. VI. QUADTREE STATISTICS FOR IMAGE 4 (PER CENT VALUES)

Block Level	Code Type	s0	s1	s2	s3	s4	s5	s6	s7
	All Black	11.11	12.05	13.19	12.50	12.50	12.50	13.19	28.47
64x64	All White	0.00	0.00	0.00	0.00	0.00	0.00	4.86	0.00
	Quadtree	88.89	87.50	86.81	87.50	87.50	87.50	81.94	71.53
	All Black	9.18	9.13	9.00	9.72	9.92	10.12	11.23	24.76
32x32	All White	0.00	0.20	0.20	0.40	0.40	1.98	13.35	1.70
	Quadtree	90.82	90.67	90.80	89.88	89.68	87.90	75.42	73.54
	All Black	7.20	7.82	7.71	7.89	8.08	9.09	11.52	25.58
16x16	All White	3.33	5.42	5.78	5.68	6.36	13.71	19.66	14.69
	Quadtree	89.46	86.76	86.51	86.42	85.56	77.20	68.82	59.74
	All Black	5.03	6.15	6.52	6.77	9.63	12.61	15.97	30.39
8x8	All White Leaf Node	4.52 90.44	4.98 88.87	5.19 88.29	5.48 87.76	8.14 82.22	18.51 68.88	20.18 63.85	14.12 55.49
	Compressible Blocks	4.73	4.54	13.30	31.12	36.00	40.57	44.27	50.90
	Incompressible Blocks	95.27	95.46	86.70	68.88	64.00	59.43	55.73	49.10
Post-XOR Logic Minimization Applied to the above Incompressible Blocks									
Compressible Blocks after XOR		0.09	0.06	0.42	0.62	1.31	2.07	3.43	3.42
Incompressible Blocks after XOR		99.91	99.94	99.58	99.38	98.69	97.93	96.57	96.58

Overall, the results demonstrate that progressive spatial refinement down to 8×8 blocks, combined with selective XOR-based preprocessing of incompressible regions, enhances the effectiveness of Boolean function minimization across all bit planes.

V. CONCLUSION AND FUTURE SCOPE

A lossless compression framework for grayscale medical images using logic minimization is proposed. Core of the scheme is treating the bit stream at the blocks of the bit planes of the image as a Boolean function, and representing it in its minimized form. All blocks do not yield a smaller representation this way. When a block does not result in a smaller representation, the bits are stored as such. A switching function of n variables has 2^n minterms which are combined by the logic minimizer ESPRESSO into cubes. Each minterm is a vertex on a binary hypercube and the minimizer try to cover the vertices belonging to ON set into a cube which is a compact representation. ESPRESSO minimizer was developed for VLSI synthesis (of Programmable Logic Arrays) of Boolean functions with many variables. This minimizer is not the optimal one in the compression experiment where the number of variables is 5 or 6 and not 50. This minimizer was used owing to the fact that it is a proven tool and good enough to establish the proof of concept though not optimal in terms of execution time. It is always possible to create an optimal minimizer suitable for functions with a smaller number of variables. So, the time of execution reported for both compression and decompression can easily be improved with an optimized logic minimizer.

The total combination of functions for an n variable Boolean function is $2^{\{2^n\}}$. One of the functions is tautology (always 1) and the other is XOR function. Cube based minimizer do not yield optimal representation for all type of functions. For example, XOR function has a poor representation in the cube form but has an efficient representation if Binary Decision Diagrams (BDDs) are used. It is planned to explore these techniques to target blocks which are incompressible in the current scheme. As there is a general framework, new approaches can be incorporated easily.

The experiment based on the method proposed in this study is an improvement on the coding using fixed block size on the bit planes. With Quadtree, gray coding and prediction, an improvement in compression ratio is achieved though not significant over the previous method based on fixed block size. Incorporation of other techniques to handle incompressible blocks can be tried to push up the compression ratio. By looking at the statistics presented in experimental result, it is evident that some of the incompressible blocks will yield result with another representation of Boolean functions. These are some of the future directions planned for exploration.

REFERENCES

[1] Agnes J, Gunasundari S. Prediction of Lung Disease using Deep Learn- ing Techniques in Chest X-Ray Images: A Systematic Review. In2025 5th International Conference on Soft Computing for Security Applica- tions (ICSCSA) 2025 Aug 4 (pp. 1406-1416). IEEE.

[2] Shatnawi MQ, Abuein Q, Al-Quraan R. Deep learning-based approach to diagnose lung cancer using CT-scan images. Intelligence-Based Medicine. 2025 Jan 1;11:100188.

[3] Bonato B, Nanni L, Bertoldo A. Advancing precision: A comprehensive review of MRI segmentation datasets from brats challenges (2012–2025). Sensors (Basel, Switzerland). 2025 Mar 15;25(6):1838.

[4] Sherly Kanaga Priya P, Helen Sulochana C. Learned Lossless Image Compression Based on Optimal Kernel Transformer Approach. The Eu- ropean Journal on Artificial Intelligence. 2025 May;38(2):198-216.

[5] Rath RK, Hazra A, Nanda R. A Lossless Healthcare Data Compression Approach Using Near-Edge Computing. InIndustry 5.0: Key Tech- nologies and Drivers 2025 Apr 4 (pp. 69-92). Cham: Springer Nature Switzerland.

[6] Mahdi SH, Sahy SA, Hasoon JN. Video Compression Using Quadtree Decomposition and Bitplane Coding. JOIV: International Journal on Informatics Visualization. 2025 Nov 30;9(6).

[7] Dvoinishnikov SV, Zuev VO, Bakakin GV, Pavlov VA. Gray code image processing algorithm for measuring the three-dimensional geometry of objects with complex profiles. Measurement Techniques. 2025 Apr 1:1-8.

[8] Gurunath B. Logic minimization algorithms for VLSI applications (Doc- toral dissertation).

[9] Sato Y, Tezuka H, Kondo R, Yamamoto N. Quantum algorithm for partial differential equations of nonconservative systems with spatially varying parameters. Physical Review Applied. 2025 Jan 1;23(1):014063.

[10] Hanan Samet. The quadtree and related hierarchical data structures. ACM Computing Surveys (CSUR), 16(2):187–260, 1984.

[11] Hanan Samet. An overview of quadtrees, octrees, and related hierarchi- cal data structures. Theoretical Foundations of Computer Graphics and CAD, pages 51–68, 1988.

[12] Prashant Shukla, Abhishek Verma, Abhishek, Shekhar Verma, and Man- ish Kumar. Interpreting svm for medical images using quadtree. Multi- media Tools and Applications, 79(39):29353–29373, 2020.

[13] Swathi Pai M, Shyam P Joy, and Jacob Augustine. Lossless compression of grayscale images via representation of bit planes as minimized boolean functions. International Journal of Electronics and Electrical Engineering (IJEEE), 2025.

[14] Mahdi SH, Sahy SA, Hasoon JN. Video Compression Using Quadtree Decomposition and Bitplane Coding. JOIV: International Journal on Informatics Visualization. 2025 Nov 30;9(6).

[15] Chaudhay AK, Augustine J, Jacob J. Lossless compression of images using logic minimization. InProceedings of 3rd IEEE International Con- ference on Image Processing 1996 Sep 19 (Vol. 1, pp. 77-80). IEEE.

[16] Augustine J, Feng W, Jacob J. Logic minimization based approach for compressing image data. InProceedings of the 8th International Confer- ence on VLSI Design 1995 Jan 4 (pp. 225-228). IEEE.

[17] Choi J, Lee S, Kang K, Suh H. Lightweight Machine Learning Method for Real-Time Espresso Analysis. Electronics. 2024 Feb 19;13(4):800.

[18] Butler JT, Sasao T. Boolean functions for cryptography. InProgress in Applications of Boolean Functions 2022 Mar 17 (pp. 33-53). Cham: Springer International Publishing.

[19] PejaŚ J, Cierocki Ł. Reversible data hiding scheme for images using gray code pixel value optimization. Procedia Computer Science. 2021 Jan 1;192:328-37.

[20] Nazemi M, Kanakia H, Pedram M. Heuristics for million-scale two-level logic minimization. In2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD) 2021 Nov 1 (pp. 1-7). IEEE.

[21] Wang C, Tao Y. Karnaugh maps of logical systems and applications in digital circuit design. Circuits, Systems, and Signal Processing. 2020 May;39(5):2245-71.

[22] Van Aerschot W, Jansen M, Bultheel A. Normal offsets for digital image compression. InWavelet Applications in Industrial Processing III 2005 Nov 7 (Vol. 6001, pp. 148-159). SPIE.

[23] Liu X, Abd-Elmoniem KZ, Stone M, Murano EZ, Zhuo J, Gullapalli RP, Prince JL. Incompressible deformation estimation algorithm (IDEA) from tagged MR images. IEEE transactions on medical imaging. 2011 Sep 19;31(2):326-40.

[24] Swathi PM, Nandan RP, Augustine J. Logic minimisation and compres- sion—A lossless approach for medical imaging. InEmerging Technologies in AI, Computation, Communication, and Cybersecurity (pp. 355-361). CRC Press.

[25] Zhou RG, Sun YJ, Fan P. Quantum image Gray-code and bit-plane scrambling. Quantum Information Processing. 2015 May;14(5):1717-34.

[26] He D, Yang Z, Peng W, Ma R, Qin H, Wang Y. Elic: Efficient learned image compression with unevenly grouped space-channel contextual adap- tive coding. InProceedings of the IEEE/CVF conference on computer vision and pattern recognition 2022 (pp. 5718-5727).

[27] Faroß N, Schwarz S. Gröbner bases for Boolean function minimization. Mathematics in Computer Science. 2025 Dec;19(1):7.

[28] O'Donnell R. Analysis of boolean functions. Cambridge University Press; 2014 Jun 5.

[29] Spratling MW. A review of predictive coding algorithms. Brain and cognition. 2017 Mar 1;112:92-7.

[30] Eldawy A, Alarabi L, Mokbel MF. Spatial partitioning techniques in SpatialHadoop. Proceedings of the VLDB Endowment. 2015 Aug 1;8(12):1602-5.

[31] Descloux A, Grußmayer KS, Radenovic A. Parameter-free image resolu- tion estimation based on decorrelation analysis. Nature methods. 2019 Sep;16(9):918-24.

[32] Anitha V, Murugavalli SJ. Brain tumour classification using two-tier classifier with adaptive segmentation technique. IET computer vision. 2016 Feb;10(1):9-17.

[33] Dvoinishnikov SV, Zuev VO, Bakakin GV, Pavlov VA. Gray code image processing algorithm for measuring the three-dimensional geometry of objects with complex profiles. Measurement Techniques. 2025 Apr 1:1-8.

[34] Clunie DA. Lossless compression of grayscale medical images: effectiveness of traditional and state-of-the-art approaches. Medical Imaging 2000: PACS Design and Evaluation: Engineering and Clinical Issues. 2000 May 18;3980:74-84.

[35] Falkowski BJ. Lossless binary image compression using logic functions and spectra. Computers and Electrical Engineering. 2004 Jan 1;30(1):17- 43.

An Empirical Evaluation of Multivariate Temporal Convolutional Networks with Global Market Indicators for Forecasting the Indonesian LQ45 Index

Yohanes Marakub, Muhammad Zarlis
Department of Information Systems Management, Bina Nusantara University, Jakarta, Indonesia

***Abstract*—This study develops a multivariate Temporal Convolutional Network (TCN) framework to forecast the LQ45 stock index using daily time-series data from January 2015 to January 2025. The objective is to examine whether incorporating global market indicators, namely the Volatility Index (VIX), Brent crude oil price, and the Effective Federal Funds Rate (EFFR), alongside lagged LQ45 values, improves forecasting performance in Indonesia's equity market. Two comparison models are considered: an Autoregressive Integrated Moving Average with Exogenous Variables (ARIMAX) model as a statistical baseline and a univariate TCN as a deep learning benchmark. Data preprocessing includes normalization and a seven-day sliding-window framing. Forecasting accuracy is evaluated using Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), complemented by validation and interpretability analyses. The results show that while the multivariate TCN captures interactions among multiple temporal features, it does not provide measurable performance advantages over ARIMAX or the univariate TCN. The lagged LQ45 series exhibits the strongest predictive contribution, followed by EFFR with a stable secondary effect, whereas Brent oil prices and VIX display weak and unstable influences. These findings suggest that in short-horizon forecasting of relatively stable emerging markets, integrating exogenous variables showed no performance improvement and was associated with higher forecast error when their predictive structure was unstable, highlighting the trade-off between feature complexity and model robustness.**

***Keywords*—Deep learning; financial forecasting; multivariate forecasting; TCN; time series analysis**

I. Introduction

A stock represents a unit of ownership in a company and is one of the most traded financial instruments in the capital market. In Indonesia, the performance of listed companies is measured through market indices that summarize the movement of selected securities. Among these indices, the LQ45 Index, which was introduced by the Indonesia Stock Exchange (IDX) in 1997, consists of forty-five companies, selected based on liquidity and market capitalization, representing fundamentally strong and actively traded equities [1]. The index serves as a primary benchmark for evaluating market performance and has become a key indicator for investors to assess the stability and attractiveness of the domestic capital market [2]. The value of stock indices, including the LQ45, is influenced by both domestic factors and global market indicators that shape investor sentiment and cross-border capital flows in emerging markets [3]. Therefore, forecasting the movement of the LQ45 Index is essential to support investment decision-making and risk management in Indonesia's financial market.

The movement of the LQ45 Index is also affected by global financial conditions and external market indicators that transmit volatility and capital flow shocks into Indonesia's capital market. One of the key indicators, the Volatility Index (VIX), reflects investor sentiment and global risk perception, which can drive fluctuations in emerging market equities [4]. Brent oil prices, on the other hand, reflect both global energy demand and inflationary expectations that significantly affect stock market dynamics in Asian economies.

Studies in China and ASEAN countries confirm that oil price shocks have a strong influence on equity volatility, sectoral performance, and risk spillovers [5, 6]. Moreover, the Effective Federal Funds Rate (EFFR), reflecting the operational stance of U.S. monetary policy, serves as a transmission channel through which global liquidity conditions affect equity markets across Asia, as documented by evidence on U.S. policy spillovers to emerging economies [7, 8]. These interconnected factors highlight the importance of a multivariate predictive framework that can incorporate global indicators to improve the accuracy of LQ45 Index forecasts.

Previous studies have employed both classical statistical methods, such as Autoregressive Integrated Moving Average (ARIMA) and modern deep learning architectures including Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) to forecast financial time series and other real-world temporal data [3,9]. Although these models have achieved satisfactory forecasting performance, most prior studies relied on domestic or univariate data without incorporating external market indicators, despite evidence that global financial dynamics significantly influence Indonesia's stock market performance [10].

Moreover, while Temporal Convolutional Networks (TCN) have been increasingly applied in financial time series modeling, their application in forecasting the LQ45 Index has not yet been widely examined. This is important because recent studies in both financial and non-financial domains have shown that TCN can achieve higher stability and accuracy than recurrent architectures such as LSTM and RNN when modeling long-term temporal dependencies [11, 12]. Therefore, this study applies a multivariate TCN framework and systematically compares it with a univariate TCN and a classical Autoregressive Integrated Moving Average with Exogenous Variables (ARIMAX)

baseline to evaluate whether the integration of the Volatility Index (VIX), Brent crude oil price, and the Effective Federal Funds Rate (EFFR) provides measurable forecasting benefits for the LQ45 Index. This research contributes empirical evidence on the practical value and limitations of incorporating global market indicators into deep learning-based forecasting models for Indonesia's equity market.

The remainder of this study presents the related literature, research methodology, experimental results and discussion, followed by the conclusion.

II. Literature Review

A. Classical and Statistical Forecasting Models

Classical statistical models, such as the Autoregressive Integrated Moving Average (ARIMA) and its multivariate extension, ARIMAX, have long been established as fundamental tools for time series forecasting due to their interpretability and low computational cost. In the Indonesian stock market, traditional time series and econometric approaches remain widely used when analyzing or modeling the behavior of the LQ45 Index, and linear frameworks continue to dominate empirical studies linking macroeconomic variables with LQ45 movements [2]. Globally, ARIMA-based models are still frequently compared with modern machine learning approaches, as they provide stable baselines for short-term and linear data series [13].

However, the rigid linear structure of ARIMAX limits its capacity to learn contextual temporal dependencies or feature representations when data exhibit volatility and cross-variable interaction [14, 15]. This structural limitation has encouraged the use of deep learning architectures that rely on hierarchical convolutional or recurrent mechanisms to capture dynamic and long-term temporal patterns more effectively. Even when configured with linear activation functions to maintain continuous forecasting behavior, models such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Temporal Convolutional Networks (TCN) remain capable of learning multi-scale dependencies across time [3].

B. Deep Learning Approaches in Financial Time Series

Deep learning techniques have become increasingly popular in financial forecasting due to their capability to model nonlinear patterns and long-term temporal dependencies that conventional statistical models cannot capture. Recurrent neural architectures such as Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) have demonstrated strong performance in sequential modeling, including LQ45 stock index prediction [3], yet they are limited by sequential computation and gradient instability.

Temporal Convolutional Networks (TCN) were introduced as an alternative architecture that uses causal and dilated convolutions to support stable gradient propagation and parallel processing, enabling more reliable, robust, and efficient learning under volatile market conditions [16]. Recent reviews in financial time series forecasting further indicate that convolution-based architectures, including TCN, effectively extract hierarchical local features and provide stronger flexibility for modeling nonlinear market patterns compared with classical approaches [17, 18].

C. The Role of Global Market Indicators

While deep learning architectures such as the Temporal Convolutional Network (TCN) have demonstrated strong capability in capturing temporal structures and long-range dependencies, previous studies indicate that deep learning models achieve improved predictive performance when incorporating diverse and high-quality external market indicators [19, 20]. In financial forecasting, such external factors, commonly referred to as global market indicators, serve as exogenous variables that capture cross border interactions affecting stock index movements.

Among the most influential are the Volatility Index (VIX), Brent crude oil price, and the Effective Federal Funds Rate (EFFR). The VIX reflects global investor sentiment and risk perception, and its spikes often lead to increased uncertainty and capital outflows from emerging markets [4]. Brent crude oil prices represent global energy demand and inflationary expectations, significantly influencing stock market volatility and sectoral performance in Asian economies [5, 6]. Meanwhile, the EFFR represents the stance of United States monetary policy, shaping international capital flows and liquidity across Asian emerging markets [7, 8].

Collectively, these indicators represent volatility, commodity, and monetary dimensions of global finance, all of which jointly influence the performance of regional indices such as LQ45. Integrating these global dynamics into forecasting frameworks therefore enables a more comprehensive representation of market behavior and improves model robustness in capturing cross market interdependencies.

D. Research Gap and Novelty

Previous studies have applied various methods to forecast the LQ45 Index, including classical statistical models such as ARIMA and deep learning architectures like Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) [1, 3]. Although these models achieved reasonable accuracy, most relied on domestic or univariate data without integrating global indicators, even though prior evidence shows that international financial dynamics can affect equity performance in Asian emerging markets, including Indonesia [2, 10]. Traditional models such as ARIMA are constrained by linear assumptions and limited ability to represent complex temporal structures [9, 13]. Recurrent architectures like LSTM capture short-term dependencies but remain restricted by sequential computation and overfitting risk when processing long sequences [21].

In contrast, Temporal Convolutional Networks (TCN) provide longer effective memory, parallel training efficiency, and stable gradients, consistently outperforming recurrent models in sequence forecasting [22]. However, limited evidence exists regarding how exogenous feature integration interacts with advanced convolutional architectures such as TCN in short-horizon LQ45 forecasting, particularly in evaluating the predictive role of global indicators including VIX, Brent crude oil, and EFFR. Therefore, this study develops a multivariate TCN framework integrating these indicators and compares it with the ARIMAX baseline to examine the implications of

feature stability and model complexity in short-horizon financial forecasting.

III. METHODOLOGY

A. Research Framework and System Overview

The proposed research framework aims to evaluate the forecasting performance of Temporal Convolutional Network (TCN) and ARIMAX models for predicting the LQ45 stock index based on global indicators: VIX, Brent Oil, and the Effective Federal Funds Rate (EFFR). The framework consists of five stages: data acquisition, preprocessing, hyperparameter optimization, model development, and performance evaluation with subsequent analysis, as shown in Fig. 1. This structured pipeline ensures methodological consistency across experimental settings and facilitates a systematic comparison between statistical and deep learning approaches.

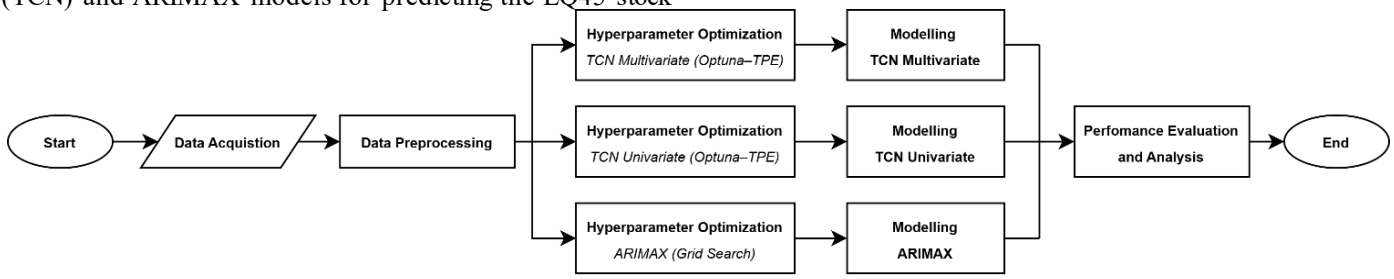


Fig. 1. Research framework for LQ45 forecasting using TCN and ARIMAX models.

The data flow begins with the collection of daily financial data, followed by preprocessing steps to ensure consistency and readiness for model input. Model training and tuning are conducted independently for both TCN variants using Optuna TPE, and for ARIMAX using grid search. The final stage focuses on performance evaluation and analysis using both statistical and interpretability metrics. Detailed descriptions of each stage are provided in Section III-B to Section III-F.

B. Data Description and Collection

The dataset used in this study consists of daily financial data covering the period from January 2015 to January 2025, selected to capture multiple market phases including pre-pandemic stability, the COVID-19 shock, and recovery dynamics. The dependent variable is the LQ45 stock index, representing the performance of the 45 most liquid and fundamentally strong stocks listed on the Indonesia Stock Exchange (IDX). Historical lag values of LQ45 are included to capture short-term temporal dependencies relevant for forecasting. Three global indicators are incorporated as exogenous variables: the Volatility Index (VIX) to reflect global risk sentiment, Brent Crude Oil price to represent commodity-related macroeconomic exposure relevant to Indonesia, and the Effective Federal Funds Rate (EFFR) to capture global monetary policy influence. LQ45, VIX, and Brent Oil data were obtained from Yahoo Finance, while the EFFR was collected from the Federal Reserve Economic Data (FRED). All variables were synchronized by date to ensure temporal alignment and consistency, producing 2,591 daily observations for each variable.

C. Data Preprocessing

The preprocessing phase ensures that all variables are properly structured and ready for supervised learning. The temporal alignment of exogenous indicators was refined based on Granger causality screening (lags 1–10), where statistically significant relationships ($p < 0.05$) were detected across multiple lag intervals, although the strength and stability of significance varied across VIX, Brent, and EFFR. This procedure follows the principle that causal influence in time-series systems may emerge at different lag intervals across variables [23]. Missing values were handled using linear interpolation, a simple yet effective approach for short gaps in continuous time-series data [24]. Outlier detection was conducted using the interquartile range (IQR) method; however, extreme values were retained since financial returns exhibit heavy-tailed behavior, and such extreme fluctuations contain valuable information about market dynamics [25]. A sliding-window framing with a seven-day lookback was then applied to generate input-output sequences for one-step-ahead forecasting. This window length was selected based on empirical findings demonstrating that a seven-day window achieves the best predictive accuracy in financial time-series forecasting before performance declines with longer input spans [26]. Finally, the dataset was chronologically divided into 80% training and 20% testing subsets, with 10% of the training portion allocated for validation, and all features were normalized using Min-Max scaling to ensure consistent numerical ranges across the dataset.

D. Model Architecture and Implementation

This study employs two forecasting models, namely the Autoregressive Integrated Moving Average with Exogenous Variables (ARIMAX) as the statistical baseline and the Temporal Convolutional Network (TCN) as the deep learning model. Both are designed to perform one-step-ahead prediction of the LQ45 Index using seven-day sliding-window sequences as inputs.

1) ARIMAX: ARIMAX is an extension of the ARIMA framework that incorporates exogenous variables to improve forecasting accuracy when external drivers influence the target series. The model combines autoregressive, differencing, and moving-average components with lagged external predictors, enabling it to represent both internal temporal dynamics and the impact of relevant external signals. Recent studies have shown that ARIMAX remains a widely used statistical approach for time series forecasting, particularly in applications where external environmental or climatic variables inform system behavior. Its formulation relies on stationarity assessment, order identification through ACF and PACF patterns, and parameter selection based on information criteria, ensuring that the resulting model remains parsimonious and interpretable.

Although ARIMAX may struggle in highly nonlinear settings, it provides a stable and computationally efficient baseline for forecasting tasks involving exogenous influences [27]. As a linear parametric framework, ARIMAX assumes additive relationships and stationarity of the underlying series, which may constrain its flexibility in capturing nonlinear or regime-dependent financial dynamics. In this study, model order selection was conducted through grid search over p, d, q combinations, allowing the differencing parameters to adapt to potential non-stationary behavior. However, formal unit-root tests such as ADF or KPSS were not explicitly reported and are acknowledged as a methodological limitation.

Recent empirical research has further validated the robustness of ARIMAX as a baseline model in financial forecasting. Previous studies have applied the ARIMAX model to the Vietnam stock index (VNINDEX) by incorporating dual market indicators as exogenous variables and demonstrated that ARIMAX outperformed non-linear benchmarks such as LSTM, GA-LSTM, XGBoost, and Prophet [14]. Their results confirmed that ARIMAX provides a reliable and interpretable framework for capturing the influence of external factors on financial time-series movements while maintaining computational efficiency and model stability.

In this study, ARIMAX serves as the statistical baseline for benchmarking the proposed deep learning approach. Empirical findings indicate that ARIMAX performs consistently well in financial forecasting when relevant exogenous variables are included and can outperform nonlinear alternatives when external signals exhibit strong explanatory strength [14]. In multivariate settings, improvements in predictive accuracy are further enhanced when exogenous drivers are properly specified, reinforcing ARIMAX as a stable and interpretable benchmark for comparison with deep learning models [27]. These insights collectively reinforce the role of ARIMAX as a stable and interpretable statistical benchmark, making it suitable for comparative evaluation against deep learning models.

2) Temporal Convolutional Network (TCN): Temporal Convolutional Network (TCN) is a convolution-based deep learning architecture designed for sequential data modelling. Unlike recurrent neural networks that process information step by step, TCN applies one-dimensional causal convolutions so that each output at time t depends only on the current and past inputs, thereby maintaining temporal causality. The model extends its receptive field through dilated convolutions, where gaps are inserted between filter elements to capture long-term dependencies without increasing the number of parameters. Each convolutional layer operates in parallel across time steps, which allows faster computation and more stable gradient propagation compared with recurrent structures.

The standard TCN architecture consists of stacked residual blocks that include two layers of dilated causal convolution, weight normalization, ReLU activation, and dropout, followed by an optional 1×1 convolution to align feature dimensions. These components collectively expand the receptive field, regulate parameter magnitudes, mitigate overfitting, and maintain consistent feature dimensionality across layers. Combination of dilation, residual pathways, and normalization mechanisms enables more stable gradient propagation and supports efficient optimization, allowing the architecture to represent both short-range and moderately long-range temporal dependencies [22].

Recent reviews on financial forecasting highlight growing interest in convolution-based sequence models, including architectures such as TCN, due to their ability to capture hierarchical local patterns and nonlinear temporal dynamics that classical statistical models struggle to represent [17]. These convolutional architectures learn temporal structure through stacked filters that efficiently extract multi-scale dependencies, reinforcing the suitability of TCN for modelling nonlinear financial time series [17]. In this study, the TCN framework was implemented using a seven-day sliding-window input, where dilations (1, 2) were applied to expand the effective receptive field beyond the input length, allowing the model to capture both short-term variations and broader temporal patterns while maintaining computational efficiency.

E. Experimental Setup and Tuning

The experimental setup was designed to provide a consistent and reproducible environment for model training and evaluation. All experiments were executed locally using GPU acceleration, as summarized in Table I. The software stack consisted of Python 3.10 and TensorFlow 2.15, with Optuna employed for automated hyperparameter optimization. Fixed random seeds were applied to ensure reproducibility across trials and to minimize stochastic variation during model training. This configuration ensured a stable computational environment throughout the experimental process.

TABLE I. EXPERIMENTAL ENVIRONMENT AND SYSTEM CONFIGURATION

Component	Specification
Device	Laptop with AMD Ryzen 7 5800H CPU (8 cores, 16 threads) and RTX 3060 GPU (6 GB)
Operating System	Windows 11 Home 64-bit
RAM	16 GB DDR4
Programming Language	Python 3.10
Deep Learning Framework	TensorFlow 2.15, Keras (via tf.keras)
Supporting Libraries	NumPy, Pandas, scikit-learn, Optuna
ARIMAX Optimization	Grid search over (p,d,q) combinations based on MAE
TCN Optimization	Optuna TPE Sampler (150 trials)
Reproducibility Settings	random.seed(42), numpy.random.seed(42), tf.random.set_seed(42)
Execution Environment	Local GPU (Laptop, non-cloud setup)

To ensure a fair and objective comparison between the forecasting models, a systematic hyperparameter tuning procedure was conducted for both ARIMAX and TCN. Each model was tuned independently using optimization strategies appropriate to its underlying structure to identify configurations that yield optimal predictive performance while maintaining model stability and reproducibility. In addition, all hyperparameter tuning experiments were conducted under identical data splits and evaluation protocols to ensure comparability across models. This design choice minimizes potential bias arising from data leakage or inconsistent validation settings during the optimization process.

ARIMAX was optimized using grid search over selected autoregressive, differencing, and moving-average orders. The TCN models were tuned automatically using Optuna with a Tree-structured Parzen Estimator (TPE) sampler, which efficiently explored parameters such as filters, dense units, learning rate, batch size, and dropout rate. This Bayesian optimization approach adaptively selected promising configurations and has been shown to achieve higher tuning efficiency compared with grid or random search strategies [28, 29]. Table II summarizes the defined search space and the optimal hyperparameters identified through the Optuna process.

TABLE II. HYPERPARAMETER SEARCH SPACE AND OPTIMAL CONFIGURATIONS FOR ARIMAX AND TCN MODELS

Model	Parameters	Search Space	Optimal Value
ARIMAX	Order (p, d, q)	$p \in \{1, 2\}$, $d \in \{0, 1\}$, $q \in \{0, 1\}$	(2, 0, 1)
TCN Univariate	Dilations	Fixed = [1,2]	[1,2]
	Kernel Size	Fixed = 2	2
	Filters	[16, 32, 64]	16
	Dense Units	[8,16,32]	32
	Learning Rate	[1e−4, 1e−3] (log-uniform)	2.90×10^{-4}
	Batch Size	[8, 16, 32]	32
	Dropout Rate	[0.0, 0.2]	0.08
TCN Multivariate	Dilations	Fixed = [1,2]	[1,2]
	Kernel Size	Fixed = 2	2
	Filters	[16, 32, 64]	32
	Dense Units	[8,16,32]	8
	Learning Rate	[1e−4, 1e−3] (log-uniform)	4.44×10^{-4}
	Batch Size	[8, 16, 32]	32
	Dropout Rate	[0.0, 0.2]	0.20

Each model was trained for a maximum of 100 epochs with early stopping (patience = 30) to prevent overfitting. The Univariate TCN converged at epoch 65, while the Multivariate TCN reached optimal validation accuracy at epoch 86. For ARIMAX, the search space was restricted to low-order combinations ($p \in \{1,2\}$, $d \in \{0,1\}$, $q \in \{0,1\}$) as part of this study's methodological design. This follows the parsimony principle, where parsimonious ARIMA-type models are preferred over over-parameterized alternatives for stability and computational efficiency [30].

The dilation pattern (1, 2) was selected to capture short-to-mid-term dependencies within the seven-day input window while preserving causal ordering [22]. This configuration expands the receptive field exponentially and supports multi-scale temporal aggregation without increasing model depth [11]. Prior financial forecasting studies have shown that compact receptive fields improve short-horizon accuracy by emphasizing recent dynamics [31]. The convolutional block employed a kernel size of 2 to extract short-term variations while retaining continuity across adjacent time steps [11]. Empirical evidence indicates that compact kernels enhance training stability and limit unnecessary parameter growth [32].

Building on prior TCN implementations, a filter configuration of {16, 32, 64} was adopted to balance expressive capacity and computational efficiency, a setting previously shown to yield smooth convergence behavior in TCN-based forecasting tasks [33]. A dense readout layer was appended after the TCN block to produce regression outputs, while the multivariate configuration incorporated fully connected layers with 8, 16, and 32 ReLU units to strengthen nonlinear interactions among the exogenous indicators [33]. The univariate configuration used a linear activation at the output stage, which is standard in continuous-valued forecasting where temporal dependencies are already encoded by the convolutional layers [11, 34].

Regularization and optimization parameters were tuned to support stable convergence. Dropout values between 0.0 and 0.2 were tested based on evidence that moderate dropout improves generalization in Temporal Convolutional Networks [33, 35]. The Adam optimizer used a learning-rate range of 1×10^{-4} to 1×10^{-3} to ensure smooth gradient updates, consistent with recent TCN frameworks [36, 37]. Batch sizes {8, 16, 32} were adopted to balance computational efficiency and gradient-update stability in multivariate forecasting tasks [35].

F. Evaluation Metrics and Validation

The evaluation stage aimed to assess model performance in terms of predictive accuracy, generalization, and computational efficiency using the reserved test dataset to ensure unbiased comparison. Model accuracy was quantified through Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), which represent complementary perspectives of error magnitude and proportional deviation. RMSE emphasizes large deviations and penalizes extreme forecast errors, while MAE reflects average

prediction accuracy in the same scale as the original data. MAPE provides an interpretable percentage-based measure that facilitates cross-model comparison. Together, these three metrics ensure a balanced evaluation between precision and robustness across all model configurations [38].

To ensure fairness and reliability of the comparison, the Diebold-Mariano (DM) test was employed to evaluate the statistical significance of forecast differences between ARIMAX and TCN models [38]. Feature attribution was further investigated using Integrated Gradients (IG) to visualize temporal attributions along historical input windows [39], and Permutation Feature Importance (PFI) to estimate the relative contribution of each exogenous factor, including LQ45 historic data, VIX, Brent Oil, and EFFR [40]. These analysis revealed the extent to which multivariate TCN utilized external indicators to enhance predictive power.

Additionally, the total runtime and hyperparameter search duration were recorded to measure computational efficiency and scalability. This comprehensive validation framework ensured that the comparative analysis reflected not only predictive accuracy but also interpretability, robustness, and efficiency across models.

IV. Results and Discussion

This section presents the empirical results of the comparative forecasting experiments, followed by a structured discussion to interpret model performance, statistical significance, feature relevance, and computational implications.

A. Model Performance Comparison

The comparative evaluation aims to assess the forecasting accuracy and robustness of three primary models, namely the ARIMAX (2,0,1) baseline representing classical statistical methods, the Univariate Temporal Convolutional Network (TCN) as a deep learning approach relying solely on historical LQ45 patterns, and the Multivariate TCN that integrates global exogenous indicators (VIX, Brent Oil, and EFFR). The evaluation metrics used are Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). The summarized results on the testing dataset are presented in Table III.

TABLE III. Model Performance Comparison on the Test Dataset

Model	RMSE	MAE	MAPE (%)
ARIMAX	7.47	5.72	0.62
TCN Univariate	10.57	8.34	0.90
TCN Multivariate	13.02	9.96	1.08

The comparative performance trends presented in Table III are further illustrated in Fig. 2, which visually depicts the RMSE and MAE values across the three evaluated models. This graphical presentation reinforces the observed error hierarchy and facilitates a clearer interpretation of relative model robustness on the testing dataset. Such visualization also enhances interpretability by providing an intuitive comparison of performance differences across modeling approaches.

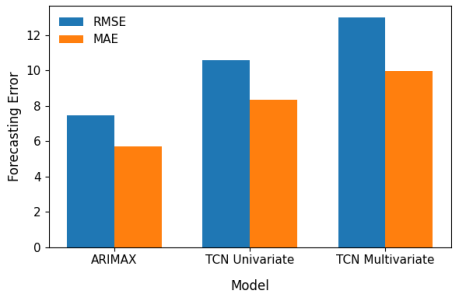


Fig. 2. RMSE and MAE comparison across forecasting models.

To further verify the adequacy of the ARIMAX specification, residual diagnostics were conducted on the test forecast errors. The autocorrelation function (ACF) plot indicates that residual correlations remain within the 95% confidence bounds across lags. In addition, the Ljung–Box test produced p-values greater than 0.05 at lag 10 and 20, suggesting that the null hypothesis of no serial correlation cannot be rejected. These results suggest that the ARIMAX residuals approximate white noise behavior, indicating that the linear structure adequately captures the underlying temporal dependence of the LQ45 series (see Fig. 3).

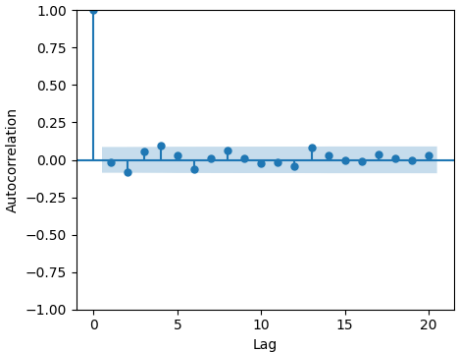


Fig. 3. Autocorrelation function of ARIMAX forecast errors.

The univariate TCN achieved an RMSE of 10.57 and MAPE of 0.90 per cent, indicating moderate forecasting capability. Although less accurate than ARIMAX, it demonstrated the ability to capture intrinsic temporal dependencies through causal and dilated convolutions [42], allowing the model to learn both short- and long-term patterns in the LQ45 index, consistent with prior studies demonstrating TCN's efficiency in modeling sequential dependencies, including financial time series [43].

Conversely, the multivariate TCN recorded the highest errors, suggesting that integrating exogenous variables such as VIX, Brent Oil, and EFFR may have introduced additional noise and reduced model stability. This outcome aligns with the concept of forecastability variance, which states that the predictive contribution of exogenous features fluctuates depending on their causal strength and signal stability [44]. When such auxiliary variables exhibit stochastic volatility or weak causality with the target series, the resulting instability can diminish model generalization and elevate forecast uncertainty [44]. This behavior has also been observed in recent multivariate TCN studies, where noisy exogenous inputs distorted temporal representations and propagated volatility across convolutional layers [33].

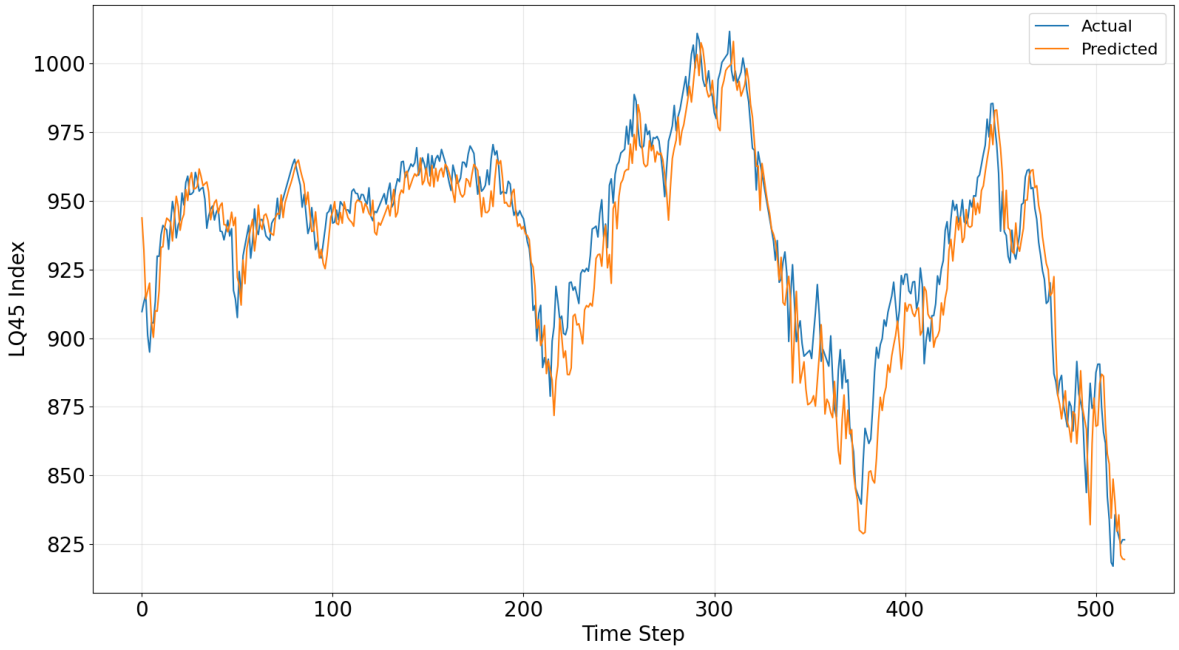


Fig. 4. Actual and predicted LQ45 index using the multivariate TCN model.

Although the univariate TCN leverages deep hierarchical structures, its predictive accuracy remained slightly below that of the ARIMAX model. This result may be explained by the dominance of short-term linear dependencies within the LQ45 index, where autoregressive mechanisms efficiently capture recurring temporal patterns [38, 41]. Meanwhile, deep learning architectures such as TCN are generally more advantageous when nonlinear or multi-scale interactions dominate the series, which appears less pronounced in the LQ45 dataset [43]. Hence, when temporal dynamics are relatively smooth and stationary, classical statistical models can still achieve competitive performance. Taken together, these quantitative and visual analyses indicate that the inclusion of exogenous features did not yield measurable predictive improvement in the evaluated setting, suggesting a stronger influence of intrinsic temporal dynamics in the LQ45 index. This finding further suggests that model complexity should be carefully aligned with underlying data characteristics to avoid unnecessary overparameterization (see Fig. 4).

Overall, the comparative findings highlight that ARIMAX remains an efficient and accurate baseline, while the univariate TCN offers a flexible deep-learning alternative with comparable stability. Although the multivariate TCN captured broader temporal structures, its error rates increased due to noisy exogenous inputs [33, 43]. These results suggest that for short-horizon forecasting in emerging markets such as Indonesia, simpler univariate frameworks may yield more consistent and interpretable outcomes.

B. *Statistical Significance Analysis using Diebold-Mariano Test (DM Test)*

The Diebold-Mariano (DM) test was applied to examine whether the observed forecasting differences among models were statistically significant. This procedure compares the expected loss differential between two competing models and determines if one achieves superior predictive accuracy [38]. A significant level of 5 per cent was used, reflecting standard practice in recent financial forecasting research [38]. The test was conducted pairwise among ARIMAX, Univariate TCN, and Multivariate TCN models using three evaluation metrics: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) (see Table IV).

All models were aligned on identical time spans to ensure comparability and to avoid sampling bias during evaluation. This pairwise testing approach ensures that the statistical comparison isolates model-specific differences rather than artifacts arising from sampling variability, thereby providing a more reliable basis for determining whether the observed accuracy gaps reflect genuine structural distinctions between the forecasting methods.

TABLE IV. DIEBOLD-MARIANO TEST RESULTS ACROSS ALL EVALUATION METRICS

Pair	MAE			RMSE			MAPE		
	Delta (Δ)	DM stat	p-value	Delta (Δ)	DM stat	p-value	Delta (Δ)	DM stat	p-value
TCN Multivariate vs. TCN Univariate	1.62	5.932	5.50e-09	57.65	6.26	7.98e-10	0.0018	5.83	9.89e-09
TCN Multivariate vs. ARIMAX	4.25	11.2	<1e-12	113.72	9.01	<1e-12	0.0046	11.04	<1e-12
TCN Univariate vs. ARIMAX	2.63	9.31	<1e-12	56.07	8.33	<1e-12	0.0028	9.23	<1e-12

Across all metrics, p-values were far below 0.05, confirming that the observed performance gaps among the models were statistically significant. Moreover, the consistently positive loss differentials (Δ) and large DM statistics across all model pairs indicate that the accuracy gaps are not only statistically significant but also quantitatively substantial, reinforcing the robustness of the comparative ranking. The positive Δ values indicate that the first model in each pair produced higher forecast errors, establishing a consistent performance hierarchy, where ARIMAX outperformed the Univariate TCN, which in turn outperformed the Multivariate TCN. These findings align with prior evidence that increasing model complexity does not necessarily yield higher predictive accuracy [38]. When exogenous variables contribute unstable or weakly correlated information, simpler statistical frameworks often provide more reliable results [38]. In summary, the Diebold-Mariano test confirms the statistically significant superiority of ARIMAX over both TCN variants across all error metrics.

C. Feature Contribution Analysis using Ablation Test

The ablation test was conducted to evaluate the structural contribution of each input feature to the predictive performance of the multivariate Temporal Convolutional Network (TCN) model. In this procedure, one feature was removed at a time while the model was retrained using identical hyperparameter configurations. The resulting performance was then compared with that of the full baseline model to quantify the model's dependency on each feature. This approach has been widely applied to verify the effectiveness of architectural components and feature dependencies in deep learning forecasting models [45].

In this study, four primary input features, namely lagged LQ45 data (t-1), VIX, Brent Oil, and EFFR, were subjected to ablation using a multi-seed configuration with five random seeds. The RMSE, MAE, and MAPE were computed on the real scale and averaged for each ablation condition. The deviation in RMSE relative to the baseline multivariate TCN (13.02) was used as the key performance indicator (ΔRMSE vs. Baseline). This design ensured consistent experimental conditions, controlled stochastic variation across runs, and provided a fair comparison framework to quantify the individual influence of each input feature on predictive accuracy in the proposed framework.

TABLE V. ABLATION TEST RESULTS ACROSS ALL EVALUATION METRICS

Dropped Feature	RMSE (mean ± std)	MAE (mean ± std)	MAPE (mean ± std, %)	ΔRMSE vs Baseline
LQ45 (t-1)	53.78 ± 14.61	43.71 ± 13.20	4.65 ± 1.34	+40.76
EFFR	13.30 ± 1.73	10.27 ± 1.71	1.12 ± 0.18	+0.28
VIX	11.77 ± 1.10	9.30 ± 1.12	1.00 ± 0.12	−1.25
Brent Oil	10.88 ± 0.36	8.36 ± 0.22	0.90 ± 0.02	−2.14

The results in Table V reveal that removing the lagged LQ45 feature caused the highest increase in RMSE (+40.76) relative to the baseline, confirming that the temporal information of the LQ45 index serves as the dominant autoregressive driver in the multivariate structure. Meanwhile, the slight improvement observed after removing VIX and Brent Oil suggests that these global indicators introduce weak or noisy signals that interfere with the temporal consistency of the target variable. The small RMSE increase observed when removing EFFR reflects a minor stabilizing contribution, indicating a secondary but complementary role in capturing macro-financial dynamics. These findings demonstrate that endogenous autoregressive patterns remain the most influential predictors in the TCN framework, while exogenous features contribute primarily as contextual modifiers rather than principal drivers [46].

Given the negative ΔRMSE observed for VIX and Brent Oil, Pearson correlation analysis was conducted to examine their statistical association with LQ45. Fig. 5 illustrates the correlation structure among LQ45 and the external indicators.

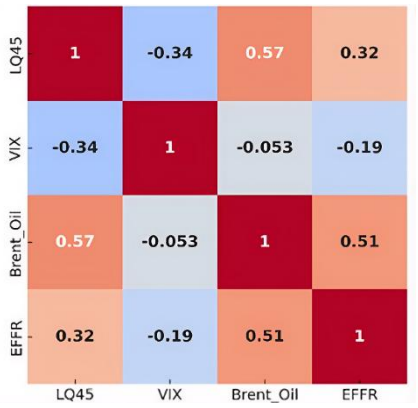


Fig. 5. Pearson correlation among LQ45 and exogenous market indicators.

Although moderate contemporaneous association is observed between LQ45 and certain exogenous indicators, static correlation does not directly imply incremental forecasting contribution, as reflected in the ΔRMSE results in Table V.

D. Temporal Attribution Analysis using Integrated Gradients

The Integrated Gradients (IG) method is a gradient-based attribution approach that quantifies feature importance by integrating gradients along a straight-line path between a baseline and the input. Because it does not modify a model's architecture, IG offers a consistent and model-agnostic mechanism for interpreting deep learning systems in sequential prediction tasks. Recent studies have demonstrated that IG yields stable and meaningful attributions in time-series forecasting, effectively highlighting the contribution of input variables and their temporal positions. These findings support the methodological suitability of IG for deep learning forecasting models, including convolution-based architectures that encode short-term and long-term dependencies through hierarchical feature representations [47].

Empirical studies have shown that Integrated Gradients provide systematic and reliable attributions for time series prediction models. The method identifies influential input variables and their associated temporal positions, enabling a transparent explanation of how variations in sequential inputs contribute to the model's output while preserving the original predictive structure [39].

In this study, each test sample was compared against five reference baselines representing the distribution of the training data. Linear integration was performed with 64 steps to derive average gradient attributions, and the results were aggregated to compute global feature importance. The averaged values were then visualized as bar plots and heatmaps to illustrate the relative contribution of each input feature and its sensitivity across temporal lags.

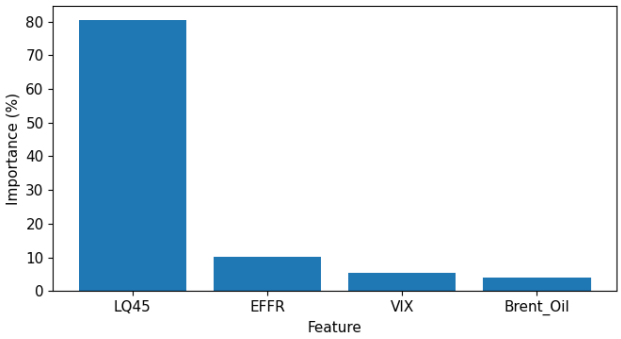


Fig. 6. Feature attribution results using integrated gradients.

As shown in Fig. 6, the LQ45 feature accounted for the largest share of total contribution, with an average importance of 80.50 per cent. This confirms that the model relies predominantly on the autoregressive dynamics of the target index. A clear gap is observed between LQ45 and the remaining features EFFR ranked as the second most influential feature at 10.15 per cent, while VIX and Brent Oil contributed 5.45 per cent and 3.89 per cent, respectively. These values indicate that macro-financial indicators provide supplementary information but play a secondary role compared with the dominant predictive influence of the LQ45 series.

TABLE VI. INTEGRATED GRADIENTS ATTRIBUTION ACROSS TEMPORAL LAGS

Lag (t-n)	LQ45 (t-n)	VIX (t-n)	Brent Oil (t-n)	EFFR (t-n)
t−1	0.192686	0.006070	0.004754	0.010884
t−2	0.001162	0.002830	0.002569	0.003966
t−3	0.002760	0.003115	0.005543	0.005910
t−4	0.000132	0.000078	0.000073	0.000366
t−5	0.001888	0.000766	0.000996	0.002343
t−6	0.000052	0.000474	0.000560	0.000198
t−7	0.000243	0.000142	0.000111	0.001427

The temporal attribution values in Table VI indicate a consistent pattern across all input variables, where the highest attribution scores are observed at lag t−1, corresponding to the most recent observation before prediction. This pattern suggests that the model consistently prioritizes short-term temporal information, regardless of whether the inputs are endogenous or exogenous. As the lag increases, the attribution scores for all variables decrease substantially, indicating that the influence of older information (t−6 to t−7) becomes notably weaker. These results confirm that the receptive-field configuration with dilations (1, 2) effectively captures short-term dependencies while reducing the risk of overfitting to distant noise [48].

E. Feature Sensitivity Analysis using Permutation Feature Importance

The feature sensitivity analysis aimed to quantify the dependency of model performance on each input variable. The Permutation Feature Importance (PFI) approach was applied as a model-agnostic interpretability method that measures changes in prediction accuracy when a single feature's temporal sequence is randomly permuted while others remain in their original order. This method has been recognized as a reliable tool for assessing the relevance of features in complex deep learning forecasting models [40]. Conceptually, PFI evaluates model sensitivity by computing the increase in prediction error after the temporal order of one feature is disrupted. A higher error increase implies a stronger dependence of the model on that feature, providing a global estimate of feature contribution across the entire dataset [48].

This approach allows a transparent evaluation of model robustness and reduces bias caused by individual feature correlations. Multiple random initializations were applied to ensure robust and reproducible sensitivity estimation, and the averaged RMSE, MAE, and MAPE were computed to obtain stable results. The deviation in RMSE relative to the baseline model (ΔRMSE) served as the main sensitivity indicator, where larger ΔRMSE values denote higher feature influence.

TABLE VII. COMPARISON OF MODEL PERFORMANCE IN FEATURE PERMUTATION TEST

Dropped Feature	RMSE (mean ± std)	MAE (mean ± std)	MAPE (mean ± std, %)	ΔRMSE vs. Baseline
LQ45 (t-1)	45.96 ± 2.59	35.75 ± 2.23	3.88 ± 0.23	+32.94
EFFR	19.04 ± 2.63	14.61 ± 1.36	1.59 ± 0.16	+6.02
VIX	17.95 ± 1.59	14.03 ± 1.05	1.52 ± 0.11	+4.93
Brent Oil	18.65 ± 2.40	14.34 ± 1.47	1.56 ± 0.16	+5.63

As presented in Table VII, randomizing the LQ45 feature resulted in the largest performance degradation (ΔRMSE + 32.94), indicating that the model relies strongly on the autoregressive structure of the LQ45 index as the main temporal driver. This effect is substantially larger than that of other features and highlights the central role of the target series in shaping predictive performance. In contrast, randomizing EFFR, VIX, and Brent Oil led to increases in moderate errors, suggesting that these global indicators contribute auxiliary information rather than being primary determinants. This pattern aligns with prior studies applying PFI to multivariate TCN frameworks for economic forecasting, confirming that the TCN model effectively captures dominant temporal dependencies while treating exogenous factors as complementary stabilizing inputs [33].

F. Computational Efficiency

This section evaluates the computational efficiency of all forecasting models by comparing the total duration of hyperparameter optimization and final training. These measurements reflect the computational effort and practical feasibility of each model in financial forecasting implementation. In addition, the comparison provides insights into the trade-off between model complexity and computational cost, which is an important consideration for real-world deployment in resource-constrained environments. Such considerations are particularly relevant for financial applications where timely model updates and operational efficiency are critical.

TABLE VIII. COMPUTATIONAL EFFICIENCY COMPARISON BASED ON TOTAL TRAINING AND HYPERPARAMETER OPTIMIZATION TIME

Model	Hyperparameter Search (HH:MM:SS.ss)	Final Training (HH:MM:SS.ss)	Total Runtime (HH:MM:SS.ss)
ARIMAX	00:00:05.51	00:00:00.21	00:00:05.72
TCN Univariate	01:16:09.32	00:00:56.17	01:17:05.49
TCN Multivariate	01:30:40.94	00:01:07.14	01:31:48.08

As presented in Table VIII, the ARIMAX model achieved the fastest overall runtime, completing both stages in approximately 5.7 seconds. The univariate TCN required around one hour and seventeen minutes, while the multivariate TCN needed about one hour and thirty-one minutes. The results indicate that the computational time consistently increases as the model architecture becomes deeper and the number of parameters grows. This pattern reflects the higher computational load of convolutional models due to iterative gradient updates and the complexity of multi-layer feature extraction. This difference also illustrates how convolutional architectures scale with input dimensionality, where each additional feature increases the computational cost of filter operations and gradient updates.

Consequently, even moderate expansions in model complexity can lead to noticeable growth in total training time. From a computational perspective, deep learning architecture generally requires longer training duration and greater training complexity than linear models [49]. The computational time of deep neural networks tends to increase rapidly as model depth and dataset size expand, often demanding extended training despite computational acceleration [49]. In contrast, the ARIMAX model in this study achieved superior predictive accuracy together with substantially faster computation. Its linear parameter estimation allows efficient convergence and minimal computational effort, making it more suitable for moderately complex financial time series that exhibit seasonal and autoregressive characteristics.

Overall, these findings confirm that statistical models such as ARIMAX remain more computationally efficient and practically reliable compared with deep learning alternatives for financial forecasting scenarios where computational efficiency and model interpretability are prioritized. The contrast in computational demands also highlights the importance of selecting modelling approaches that align with available resources and operational constraints. In practical applications, organizations may favor models that provide stable accuracy with minimal computational overhead, especially when rapid retraining or frequent model updates are required.

V. CONCLUSION

This comparative evaluation examines the forecasting accuracy and robustness of three predictive models applied to the LQ45 index. The study evaluates a multivariate Temporal Convolutional Network (TCN) incorporating global market indicators and compares its performance against ARIMAX and a univariate TCN as benchmark models. The results show that the multivariate TCN yields the highest prediction error (RMSE 13.02), indicating inferior performance relative to the benchmark models. ARIMAX achieves the lowest error (RMSE 7.47), followed by the univariate TCN (RMSE 10.57). These findings indicate that the inclusion of global exogenous indicators increased model complexity while resulting in higher forecast errors and reduced stability under the evaluated configuration.

The Diebold–Mariano test confirmed that these performance differences were statistically significant, indicating that the weaker outcomes of the multivariate TCN reflect structural limitations rather than random variation. Feature-level analyses provided consistent insights. The ablation test and Integrated Gradients showed that LQ45 dominates predictive performance, with EFFR contributing modest stabilizing effects and VIX and Brent Oil acting as weak or noisy signals. This pattern was further reinforced by the Permutation Feature Importance results, where randomizing LQ45 produced the largest degradation.

Overall, the findings indicate that short-horizon forecasting of the LQ45 index is driven primarily by its own autoregressive structure, while global indicators provided limited incremental accuracy improvement and were associated with higher error variance. Although the multivariate TCN captures hierarchical temporal dependencies, it does not surpass the ARIMAX baseline or its univariate counterpart in terms of predictive accuracy. In addition, the convolutional architectures require substantially higher computational effort due to extended training and hyperparameter optimization, which further limits their practical applicability. From both predictive and computational perspectives, ARIMAX remains a more efficient and reliable choice for short-horizon financial forecasting of the LQ45 index.

These findings are limited to the evaluated short-horizon configuration and may vary under different market regimes, including periods of structural disruption. Although the dataset spans pre-pandemic, pandemic, and recovery phases, no explicit structural-break analysis was conducted, which may influence parameter stability across macroeconomic periods. Future work may consider adaptive mechanisms such as attention-based weighting or dynamic feature selection to better manage noisy exogenous inputs under volatile market conditions.

ACKNOWLEDGMENT

The author gratefully acknowledges the thesis supervisor for the guidance, critical insights, and continuous support that contributed significantly to the completion of this research. The constructive feedback and academic discussions throughout the research process were instrumental in refining both the methodological framework and analytical interpretation presented in this study. The author also extends sincere appreciation to Bina Nusantara University for providing a supportive academic environment and institutional resources that facilitated the development and completion of this work.

DECLARATION OF GENERATIVE AI AND AI-ASSISTED TECHNOLOGIES IN THE WRITING PROCESS

During the preparation of this manuscript, the authors used ChatGPT solely as a writing assistance tool to support language refinement. All research ideas, methodological design, data analysis, and scientific conclusions were entirely developed by the authors. The authors carefully reviewed and edited the manuscript and take full responsibility for the accuracy, originality, and integrity of the published work.

REFERENCES

[1] A. Syukur and D. Istiawan, "Prediction of LQ45 Index in Indonesia Stock Exchange: A Comparative Study of Machine Learning Techniques," International Journal of Intelligent Engineering and Systems, vol. 14, no. 1, pp. 453–463, 2021, doi: 10.22266/IJIES2021.0228.42.

[2] S. H. Utomo, D. Wulandari, B. S. Narmaditya, P. Handayati, and S. Ishak, "Macroeconomic factors and LQ45 stock price index: Evidence from Indonesia," 2019, LLC CPC Business Perspectives. doi: 10.21511/imfi.16(3).2019.23.

[3] S. Hansun and J. C. Young, "Predicting LQ45 financial sector indices using RNN-LSTM," J Big Data, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00495-x.

[4] G. Sarwar and W. Khan, "Interrelations of U.S. market fears and emerging markets returns: Global evidence," International Journal of Finance and Economics, vol. 24, no. 1, pp. 527–539, Jan. 2019, doi: 10.1002/ijfe.1677.

[5] M. F. Bashir, "Oil price shocks, stock market returns, and volatility spillovers: a bibliometric analysis and its implications," Apr. 01, 2022, Springer Science and Business Media Deutschland GmbH. doi: 10.1007/s11356-021-18314-4.

[6] S. M. Hashmi, F. Ahmed, Z. Alhayki, and A. A. Syed, "The impact of crude oil prices on Chinese stock markets and selected sectors: evidence from the VAR-DCC-GARCH model," Environmental Science and Pollution Research, vol. 29, no. 35, pp. 52560–52573, Jul. 2022, doi: 10.1007/s11356-022-19573-5.

[7] T. B. N. Tran and H. C. H. Pham, "The Spillover Effects of the US Unconventional Monetary Policy: New Evidence from Asian Developing Countries," Journal of Risk and Financial Management, vol. 13, no. 8, 2020, doi: 10.3390/jrfm13080165.

[8] R. Arshad, H. Zada, K. Sohag, W. K. Wong, E. Ullah, and H. Raza, "Does US monetary policy uncertainty affect returns of Asian Developed, emerging, and frontier equity markets? Empirical evidence by using the quantile-on-quantile approach," Heliyon, vol. 10, no. 12, Jun. 2024, doi: 10.1016/j.heliyon.2024.e32962.

[9] E. D. Spyrou, I. Tsoulos, and C. Stylios, "Applying and Comparing LSTM and ARIMA to Predict CO Levels for a Time-Series Measurements in a Port Area," Signals, vol. 3, no. 2, pp. 235–248, Jun. 2022, doi: 10.3390/signals3020015.

[10] S. M. Hashmi, M. A. Gilal, and W. K. Wong, "Sustainability of global economic policy and stock market returns in Indonesia," Sustainability (Switzerland), vol. 13, no. 10, May 2021, doi: 10.3390/su13105422.

[11] P. Lara-Benítez, M. Carranza-García, J. M. Luna-Romera, and J. C. Riquelme, "Temporal convolutional networks applied to energy-related time series forecasting," Applied Sciences (Switzerland), vol. 10, no. 7, Apr. 2020, doi: 10.3390/app10072322.

[12] Y. Yang, "TCN-QV: an attention-based deep learning method for long sequence time-series forecasting of gold prices," PLoS One, vol. 20, no. 5 May, May 2025, doi: 10.1371/journal.pone.0319776.

[13] V. I. Kontopoulou, A. D. Panagopoulos, I. Kakkos, and G. K. Matsopoulos, "A Review of ARIMA vs. Machine Learning Approaches for Time Series Forecasting in Data Driven Networks," Aug. 01, 2023, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/fi15080255.

[14] P. C. Wang and T. T. H. Vo, "Stock price prediction based on dual important indicators using ARIMAX: A case study in Vietnam," Journal of Intelligent Systems, vol. 34, no. 1, Jan. 2025, doi: 10.1515/jisys-2024-0101.

[15] H. Song and H. Choi, "Forecasting Stock Market Indices Using the Recurrent Neural Network Based Hybrid Models: CNN-LSTM, GRU-CNN, and Ensemble Models," Applied Sciences (Switzerland), vol. 13, no. 7, Apr. 2023, doi: 10.3390/app13074644.

[16] W. Dai, Y. An, and W. Long, "Price change prediction of Ultra high frequency financial data based on temporal convolutional network," in Procedia Computer Science, Elsevier B.V., 2021, pp. 1177–1183. doi: 10.1016/j.procs.2022.01.149.

[17] C. Zhang, N. N. A. Sjarif, and R. Ibrahim, "Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022," Jan. 01, 2024, John Wiley and Sons Inc. doi: 10.1002/widm.1519.

[18] W. Ge, P. Lalbakhsh, L. Isai, A. Lenskiy, and H. Suominen, "Neural Network-Based Financial Volatility Forecasting: A Systematic Review," Sep. 30, 2022, Association for Computing Machinery. doi: 10.1145/3483596.

[19] E. Mienye, N. Jere, G. Obaido, I. D. Mienye, and K. Aruleba, "Deep Learning in Finance: A Survey of Applications and Techniques," Dec. 01, 2024, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/ai5040101.

[20] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, "Financial Time Series Forecasting with Deep Learning : A Systematic Literature Review: 2005-2019," Appl Soft Comput, vol. 90, Nov. 2019, doi: 10.1016/j.asoc.2020.106181.

[21] S. Giantsidi and C. Tarantola, "Deep learning for financial forecasting: A review of recent trends," International Review of Economics & Finance, vol. 104, p. 104719, Dec. 2025, doi: 10.1016/j.iref.2025.104719.

[22] X. Lv, X. Xiong, and B. Geng, "Increasing the prediction performance of temporal convolution network using multimodal combination input: Evidence from the study on exchange rates," Front Phys, vol. 10, Jan. 2023, doi: 10.3389/fphy.2022.1008445.

[23] A. Shojaie and E. B. Fox, "Granger Causality: A Review and Recent Advances," Mar. 01, 2022, Annual Reviews Inc. doi: 10.1146/annurev-statistics-040120-010930.

[24] A. Flores, H. Tito-Chura, O. Cuentas-Toledo, V. Yana-Mamani, and D. Centty-Villafuerte, "PM2.5 Time Series Imputation with Moving Averages, Smoothing, and Linear Interpolation," Computers, vol. 13, no. 12, Dec. 2024, doi: 10.3390/computers13120312.

[25] A. A. Shehadeh, S. M. Alwadi, and M. I. Almaharmeh, "Detecting and Analysing Possible Outliers in Global Stock Market Returns," Cogent Economics and Finance, vol. 10, no. 1, 2022, doi: 10.1080/23322039.2022.2066762.

[26] Z. Hu, Z. Wang, S. B. Ho, and A. H. Tan, "Stock market trend forecasting based on multiple textual features: A deep learning method," in Proc. 33rd IEEE Int. Conf. Tools with Artificial Intelligence (ICTAI), 2021, pp. 1–9, doi: 10.1109/ICTAI52525.2021.00160.

[27] D. Perazzolo, G. Lazzaro, A. Fiume, P. Fanton, and E. Grisan, "Streamflow Forecasting: A Comparative Analysis of ARIMAX, Rolling Forecasting LSTM Neural Network and Physically Based Models in a Pristine Catchment," Water (Switzerland), vol. 17, no. 15, Aug. 2025, doi: 10.3390/w17152341.

[28] S. Hanifi, A. Cammarono, and H. Zare-Behtash, "Advanced hyperparameter optimization of deep learning models for wind power prediction," Renew Energy, vol. 221, Feb. 2024, doi: 10.1016/j.renene.2023.119700.

[29] B. Bischl et al., "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges," Mar. 01, 2023, John Wiley and Sons Inc. doi: 10.1002/widm.1484.

[30] A. S. Al Shammre and B. Chidmi, "Oil Price Forecasting Using FRED Data: A Comparison between Some Alternative Models," Energies (Basel), vol. 16, no. 11, Jun. 2023, doi: 10.3390/en16114451.

[31] Z. Zhan and S. K. Kim, "Versatile time-window sliding machine learning techniques for stock market forecasting," Artif Intell Rev, vol. 57, no. 8, Aug. 2024, doi: 10.1007/s10462-024-10851-x.

[32] P. Bhatt, Y. Kumar, and A. Soulaïmani, "Deep convolutional architectures for extrapolative forecasts in time-dependent flow problems," Adv Model Simul Eng Sci, vol. 10, no. 1, Dec. 2023, doi: 10.1186/s40323-023-00254-y.

[33] J. Chen, Y. Cui, X. Zhang, J. Yang, and M. Zhou, "Temporal Convolutional Network for Carbon Tax Projection: A Data-Driven Approach," Applied Sciences (Switzerland), vol. 14, no. 20, Oct. 2024, doi: 10.3390/app14209213.

[34] H. K. M. Nguyen, Q. D. Phan, Y. K. Wu, and Q. T. Phan, "Multi-Step Wind Power Forecasting with Stacked Temporal Convolutional Network (S-TCN)," Energies (Basel), vol. 16, no. 9, May 2023, doi: 10.3390/en16093792.

[35] N. Chen, C. Su, S. Wu, and Y. Wang, "El Niño Index Prediction Based on Deep Learning with STL Decomposition," J Mar Sci Eng, vol. 11, no. 8, Aug. 2023, doi: 10.3390/jmse11081529.

[36] R. Wan, C. Tian, W. Zhang, W. Deng, and F. Yang, "A Multivariate Temporal Convolutional Attention Network for Time-Series Forecasting," Electronics (Switzerland), vol. 11, no. 10, May 2022, doi: 10.3390/electronics11101516.

[37] Z. Dou and D. Guo, "DPSTCN: Dynamic Pattern-Aware Spatio-Temporal Convolutional Networks for Traffic Flow Forecasting," ISPRS Int J Geoinf, vol. 14, no. 1, Jan. 2025, doi: 10.3390/ijgi14010010.

[38] J. Shang and S. Hamori, "Do Large Datasets or Hybrid Integrated Models Outperform Simple Ones in Predicting Commodity Prices and Foreign Exchange Rates?," Journal of Risk and Financial Management, vol. 16, no. 6, Jun. 2023, doi: 10.3390/jrfm16060298.

[39] H. Choi, C. Jung, T. Kang, H. J. Kim, and I. Y. Kwak, "Explainable Time-Series Prediction Using a Residual Network and Gradient-Based Methods," IEEE Access, vol. 10, pp. 108469–108482, 2022, doi: 10.1109/ACCESS.2022.3213926.

[40] F. Fumagalli, M. Muschalik, E. Hüllermeier, and B. Hammer, "Incremental permutation feature importance (iPFI): towards online explanations on data streams," Mach Learn, vol. 112, no. 12, pp. 4863–4903, Dec. 2023, doi: 10.1007/s10994-023-06385-y.

[41] E. Chodakowska, J. Nazarko, and Ł. Nazarko, "Arima models in electrical load forecasting and their robustness to noise," Energies (Basel), vol. 14, no. 23, Dec. 2021, doi: 10.3390/en14237952.

[42] Q. Huang and T. Hain, "Improving audio anomalies recognition using temporal convolutional attention networks," in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 6473–6477. doi: 10.1109/ICASSP39728.2021.9414611.

[43] X. Xiang, T. Yuan, G. Cao, and Y. Zheng, "Short-Term Electric Load Forecasting Based on Signal Decomposition and Improved TCN Algorithm," Energies (Basel), vol. 17, no. 8, Apr. 2024, doi: 10.3390/en17081815.

[44] S. M. A. Karim, B. Zarrin, and N. B. Lassen, "Leveraging Exogenous Regressors in Demand Forecasting," MDPI AG, Aug. 2025, p. 15. doi: 10.3390/cmsf2025011015.

[45] B. Liu, Z. Li, Z. Li, and C. Chen, "CL-Informer: Long time series prediction model based on continuous wavelet transform," PLoS One, vol. 19, no. 9 September, Sep. 2024, doi: 10.1371/journal.pone.0303990.

[46] X. Ma and H. Zhang, "Time Series Forecasting Method Based on Multi-Scale Feature Fusion and Autoformer," Applied Sciences (Switzerland), vol. 15, no. 7, Apr. 2025, doi: 10.3390/app15073768.

[47] M. Fan, D. Lu, and S. Gangrade, "Enhancing Multi-Step Reservoir Inflow Forecasting: A Time-Variant Encoder–Decoder Approach," Geosciences (Switzerland), vol. 15, no. 8, Aug. 2025, doi: 10.3390/geosciences15080279.

[48] W. Cheon et al., "Feature Importance Analysis of a Deep Learning Model for Predicting Late Bladder Toxicity Occurrence in Uterine Cervical Cancer Patients," Cancers (Basel), vol. 15, no. 13, Jul. 2023, doi: 10.3390/cancers15133463.

[49] L. Alzubaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," J Big Data, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.

Integrating ABM and GIS for Flood Evacuation Planning: A Systematic Review and Future Direction

Kabir Musa Ibrahim[1], Abubakar Ahmad[2], Noor Akma Abu Bakar[3], Mazlina Abdul Majid[4], Azamuddin Rahman[5*]
Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdallah, Pekan, Malaysia[1, 2]
College of Computing and Informatics, Universiti Tenaga Nasional (UNITEN), Putrajaya Campus, Kajang, Selangor, Malaysia[3]
The Institute of Informatics and Computing in Energy (IICE), Universiti Tenaga Nasional (UNITEN), Malaysia[3]
Faculty of Computing-Centre for Artificial Intelligence & Data Science, Universiti Malaysia Al-sultan, Abdullah, Malaysia[4]
Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdallah, Pekan, Malaysia[5]

***Abstract*—This systematic review examines the integration of agent-based modeling (ABM) and Geographic Information Systems (GIS) in flood evacuation planning from 2015 through early 2025. This review aims to systematically evaluate how ABM and GIS have been integrated in flood evacuation research, identify methodological gaps, and propose a structured framework to guide future model development. Using PRISMA guidelines, 67 studies were selected and analyzed to uncover methodological trends, empirical gaps, and policy relevance in this growing research domain. Using the PRISMA 2020 framework, the analysis reveals a dominant reliance on mesoscopic modeling (43%), limited real-time data integration (17.9%), weak empirical validation practices (16.4%), and minimal machine learning adoption (4.5%). To structure the evolving landscape, a conceptual integration framework is proposed to classify studies by modeling scale, data fidelity, and validation strategy. This framework highlights a gradual shift toward behaviorally realistic, spatially precise, and policy-relevant evacuation models. Persistent challenges include limited validation practices, weak real-time responsiveness, and insufficient policy integration. Conclusions were drawn by identifying five research priorities: AI integration, real-time enhancement, multi-hazard modeling, empirical grounding, and participatory policy co-design. This review offers actionable insights for advancing robust, scalable, and operational ABM-GIS systems in disaster risk reduction.**

Keywords*—Agent-based modeling; GIS integration; flood simulation; spatial modeling; evacuation dynamics; multi-agent systems

I. Introduction

Floods are among the most frequent and devastating natural disasters globally, causing extensive loss of life, displacement, and damage to infrastructure. Rapid urbanization, climate change, and land-use transformation have intensified both the frequency and severity of flood events in recent years [1] [2]. Notable examples include the 2018 Kerala floods in India [3], 2017 Hurricane Harvey in Houston [4], and the 2020 inundation of Wuhan, China [5]. These disasters underscore the urgent need for robust evacuation planning tools that can adapt to dynamic, uncertain conditions.

Traditional evacuation models often rely on aggregate or static assumptions, limiting their ability to reflect real-time decision-making and heterogeneous human behavior [6]. In contrast, agent-based modeling (ABM) offers a bottom-up approach, simulating individual or group behavior in complex systems [7],[8]. When coupled with GIS, which enables spatial analysis and environmental representation, ABM becomes a powerful tool for flood evacuation planning, integrating human dynamics with terrain, infrastructure, and hazard data [9], [10].

Although the use of ABM-GIS integration in flood modeling has expanded over the past decade, existing reviews have primarily focused on general disaster modeling [11] or lack systematic synthesis across methodological, spatial, and policy dimensions [12],[13],[14]. Moreover, many studies remain invalidated, use outdated data sources, or fail to connect modeling outputs to operational policy frameworks [15][16].

This study addresses these gaps by presenting a systematic review of 67 studies published between 2015 and early 2025 that combine ABM and GIS in flood evacuation contexts. PRISMA [17] review protocol is applied to ensure transparency in study selection and synthesis of findings through a newly proposed Conceptual Integration Framework. This framework categorizes studies by modeling scale, data fidelity, and validation strategy, enabling structured comparison across cases.

The goal of this review is twofold: 1) to map the current landscape of ABM-GIS flood evacuation research and 2) to identify methodological gaps and strategic priorities for future work. The main goal is to guide researchers, planners, and decision-makers in designing more adaptive, empirically grounded, and policy-relevant flood evacuation systems.

II. Literature Review

Flood evacuation modeling has evolved significantly with the integration of agent-based modeling (ABM) and Geographic Information Systems (GIS), enabling the simulation of complex, spatially grounded, and behaviorally diverse human responses to hydrological hazards. Over the past two decades, ABM-GIS frameworks have been widely adopted in urban risk assessment, emergency management, and real-time evacuation decision support [18][19].

A. Progress in ABM-GIS Integration

Recent studies demonstrate increasing sophistication in coupling GIS-based spatial layers with ABM decision logic. Hydrodynamic models (e.g., SWMM, HEC-RAS) are often used to simulate flood propagation, while ABMs simulate agent movement and decision-making under dynamic hazard conditions [20][21]. GIS platforms have enhanced spatial

*Corresponding author.

realism, allowing building-level, road-level, and infrastructure-specific modeling of evacuation behaviors.

Furthermore, models have become more inclusive of multi-agent interactions, shelter dynamics, and transportation network constraints [22] [23]. This integration enables more granular analysis of evacuation bottlenecks and policy interventions, such as staggered alerts or shelter assignments.

B. Gaps in Behavioral and Empirical Rigor

Despite technical advances, many ABM-GIS studies rely on deterministic or static behavior rules. Psychological realism, such as risk perception, delayed response, or social influence, is often underrepresented [24]. Few models incorporate agent heterogeneity (e.g., age, mobility, risk attitude), and even fewer validate outputs against real-world evacuation data.

Moreover, real-time data streams, such as IoT sensors or mobile tracking, remain rare in ABM-GIS platforms due to integration complexity and data availability constraints. Consequently, many models are scenario-driven rather than adaptive or predictive, limiting their utility in time-critical decisions.

C. Limitations of Existing Reviews

While several reviews have assessed ABM or GIS in isolation or within broader disaster contexts (e.g., pandemics, wildfires), few focus specifically on their combined application in flood evacuation modeling. For example, [25] Lakmali et al., 2024 and [26] Noor et al. (2025) discussed ABM applications in disaster simulation but offered a limited spatial methodological synthesis. More recent works (e.g., Wikstrom et al., 2022) [27] reviewed ABM-GIS tools in infrastructure planning, not evacuation behavior.

Importantly, past reviews have not systematically analyzed modeling scales, data fidelity, and validation strategies critical for translating simulations into operational emergency tools.

D. Need for the Current Review

Given the increasing severity and frequency of flood events and the rising importance of anticipatory evacuation planning, there is an urgent need to map how ABM-GIS tools are being used, validated, and operationalized in flood contexts. This review addresses that need by:

- Synthesizing 67 peer-reviewed studies with a clear inclusion and exclusion protocol.
- Classifying studies across three core dimensions: modeling scale, data fidelity, and validation strategy.
- Visualizing trends, identifying research gaps, and proposing actionable future directions.

By doing so, this review not only summarizes the current state of ABM-GIS flood evacuation modeling, but also offers a diagnostic framework for improving the realism, responsiveness, and policy alignment of future simulations. Table I shows how this review differs from other existing reviews on the area.

TABLE I. COMPARATIVE OVERVIEW OF EXISTING ABM-GIS REVIEWS AND THE PRESENT STUDY

Study	Hazard Type	ABM-GIS Focus	Scope of Review	Gap Addressed by Current Study
[20]	Floods propagation	GIS and hydrodynamic models	Flood spread, hazard mapping	Does not focus on adaptive ABM mechanisms
[23]	Urban disasters	ABM	Transportation bottlenecks, network constraints	No GIS integration for flood-specific evacuation
[25]	General disaster	GIS	ABM applications in disaster simulation	No methodological synthesis of ABM and GIS flood evacuation modeling
[24]	General disaster behaviour	ABM (Behaviour focused)	Human behavior, risk perception, response delay	Limited GIS integration; lacks spatial realism and hydrodynamic context
[27]	Infrastructure	ABM-GIS tools	ABM-GIS in infrastructure planning	No focus on evacuation behavior/modeling
This study (2025)	**Flood evacuation**	**ABM-GIS (integrated)**	**modeling scale, data fidelity, and validation strategy**	**Focus on flood-specific ABM-GIS models with Conceptual Integration Framework**

III. METHODOLOGY

This review adopts the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) 2020 guidelines [17] to ensure transparency and methodological rigor in the identification, screening, and inclusion of studies. The review protocol was not pre-registered, but all search strategies and inclusion criteria are documented in this section for full reproducibility. Two reviewers independently screened titles, abstracts, and full texts. Discrepancies were resolved through discussion. Inter-rater agreement during the full-text screening phase was 91%, indicating high screening consistency.

A. Data Sources and Search Strategy

Literature was retrieved from four major academic databases: Scopus, Web of Science, IEEE, and Science Direct. The search spanned publications from January 2015 to May 2025, focusing on studies that integrate agent-based modeling and Geographic Information Systems in flood evacuation planning. Boolean search strings combined keywords related to agent-based models, GIS, and flooding scenarios. Table II summarizes the database-specific queries. Only peer-reviewed journal articles, review papers, and selected book chapters written in English were considered.

TABLE II. DATABASES SEARCH STRATEGIES USED FOR IDENTIFYING RELEVANT ABM-GIS FLOOD EVACUATION STUDIES.

Data Source	Search Strategies
Web of Science	("Flood Evacuation Planning" OR "Flood Evacuation Modeling") AND ("Agent-Based Modeling" OR "Agent-Based Model" OR "ABS") AND ("Geographic Information System" OR "GIS")
IEEE Xplore	("Flood Evacuation" OR "Flood Rescue") AND ("Agent-Based Modeling" OR "Agent-Based Modeling") AND ("GIS" OR "Geographic Information System")
Scopus	("Agent-Based Model" OR ABM) AND ("Geographic Information System" OR GIS) AND ("Flood evacuation")
Science Direct	("Flood Evacuation Planning with Agent-Based Modeling and GIS") OR ("Systematic Review" AND "Flood Evacuation" AND "Agent-Based Modeling")

B. Inclusion and Exclusion Criteria

Studies were included in this review if they focused specifically on flood evacuation or flood response modeling, combined ABM with GIS, and produced spatially explicit modeling outputs. Conversely, studies were excluded if they addressed non-flood hazards such as wildfires or pandemics, lacked a simulation or modeling component, or were published as editorials, conference abstracts, or in languages other than English. A detailed summary of the inclusion and exclusion criteria is provided in Table III.

TABLE III. INCLUSION AND EXCLUSION CRITERIA FOR STUDY SELECTION.

Criterion Type	Inclusion	Exclusion
Publication type	Peer-reviewed journal articles, conference papers, or book chapters in English	Non-peer-reviewed, theses, reports, or non-English texts
Disaster type	Flood-specific evacuation or risk-reduction contexts	Other hazards (earthquakes, wildfires, tsunamis)
Methodology	Explicit integration of ABM and GIS, with empirical, simulated, or hybrid modeling outcomes	Purely equation-based, system-dynamics, or statistical models without ABM and GIS linkage
Data output	Studies providing measurable or model-based evacuation results	Conceptual papers without implementation or evaluation

C. Screening and Selection

The initial search yielded 215 records, which were imported into EndNote for de-duplication. After removing 32 duplicates, 183 articles were screened by title and abstract. A total of 83 full-text articles were assessed for eligibility, leading to 67 studies being included in the final synthesis. Fig. 1 visualizes the identification, screening, and inclusion stages used to select studies for this systematic review. Out of 215 records retrieved from Scopus (n = 123), IEEE Xplore (n = 27), Science Direct (n = 30) and Web of Science (n = 35), 67 studies met the inclusion criteria. Articles were excluded due to incomplete methods or data use (n = 97) and the absence of both GIS and ABM simulation components (n = 16).

D. Data Extraction and Synthesis

For each of the 67 included studies, key metadata were extracted, including the year of publication, geographic focus, modeling scale (micro, meso, or macro), level of data fidelity (e.g., static, dynamic, or real-time), validation strategy (empirical, scenario-based, or none), integration of artificial intelligence or machine learning, and the degree of policy relevance or stakeholder engagement. The extracted information was tabulated and thematically coded using an inductive approach to allow for the identification of patterns across the dataset. Summary statistics were then generated to quantify these patterns, and a Conceptual Integration Framework was developed to classify studies along the dimensions of modeling scale, data fidelity, and validation approach.

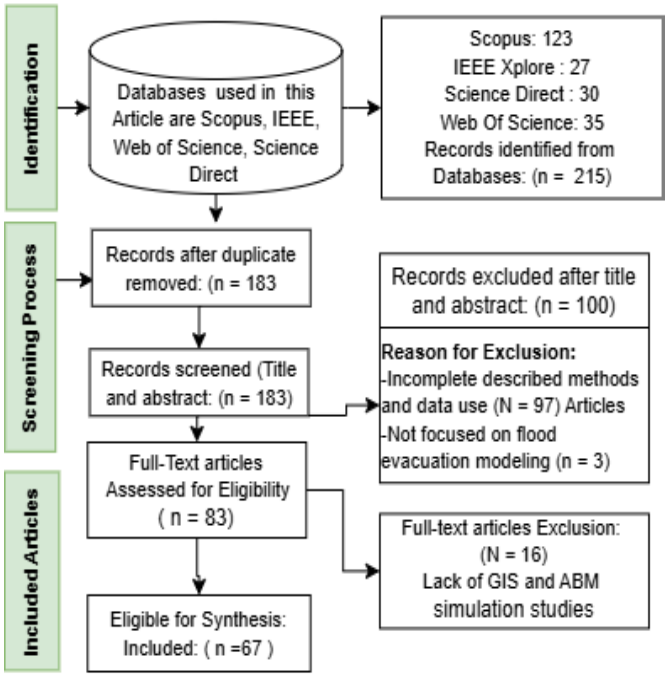


Fig. 1. Illustrates the systematic process used to identify, screen, assess, and include studies in this review of ABM-GIS integration for flood evacuation planning. Out of 215 initial records, 67 studies met the inclusion criteria based on methodological relevance, disaster type, and data integration requirements.

Due to the methodological diversity of the reviewed literature, no formal quality assessment tool (e.g., MMAT or GRADE) was applied. Instead, studies were qualitatively grouped based on the depth of empirical engagement and the transparency of their modeling processes. Table IV summarizes the data extracted from the selection process.

TABLE IV. SUMMARY OF STUDY SELECTION FROM DATABASE SEARCH RESULTS.

Database source	Initial Results	Screened Articles	Full-Text Reviewed	Included Studies
Web of science	35	30	16	14
IEEE Xplore	27	20	5	7
Scopus	123	113	45	33
Science Direct	30	20	17	13
Total	215	183	83	67

IV. RESULTS

This section presents the descriptive and thematic findings from 67 studies integrating agent-based modeling (ABM) and Geographic Information Systems (GIS) for flood evacuation planning, published between 2015 and early 2025. The analysis is structured around publication trends, geographic distribution, methodological characteristics, validation strategies, and integration capabilities.

A. Publication Trends

The publication trajectory of ABM-GIS flood evacuation research over the past decade reflects both steady methodological maturation and a sharp recent increase in scholarly interest. As shown in Fig. 2, the number of publications remained relatively modest between 2015 and 2018, averaging four to five studies per year. A notable uptick began in 2019. The annual publication count peaks in 2025 (n = 15), more than triple the annual output of most prior years. This upward trend suggests growing recognition of ABM-GIS integration as a critical tool in disaster risk reduction, particularly in the face of escalating flood-related hazards driven by climate change and urban exposure. Beyond raw publication counts, thematic synthesis of the 67 reviewed studies reveals four dominant research orientations:

- Methodological integration of ABM and GIS frameworks;
- Geographic and contextual diversity in case study settings;
- Applications in flood policy, planning, and climate adaptation; and
- Limitations related to technical scalability, data availability, and behavioral realism.

Notably, all reviewed studies (100%) integrated both ABM and GIS components in their modeling workflows, underscoring the conceptual synergy between spatial data infrastructure and agent-based decision modeling. However, methodological depth and practical integration remain uneven across the literature. Only 17.9% of studies incorporated real-time data assimilation, 16.4% conducted empirical validation against observed evacuation behavior or outcomes, and a mere 4.5% integrated machine learning or artificial intelligence components to support decision-making under uncertainty. This sharp disparity between theoretical model development and operational implementation signals a critical research gap, particularly in translating simulation outputs into actionable, real-time policy tools.

B. Geographic Distribution

Fig. 3 illustrates the geographic distribution of case studies across 30 countries, revealing clear regional concentrations and gaps. China leads with the highest number of studies (n = 8), followed by the United Kingdom and the United States (n = 6 each), and India (n = 5). Several countries, including Canada, South Korea, Thailand, the Netherlands, and Japan, each contributed three studies. A diverse but sparse representation is observed across other regions, with two studies originating from countries such as Iran, Malaysia, Germany, Nepal, and Poland. Meanwhile, a long tail of countries, including Australia, Indonesia, Greece, Portugal, and the Russian Federation, appear only once in the dataset.

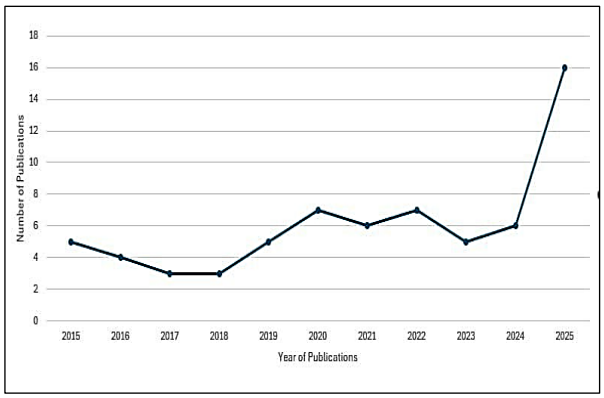


Fig. 2. Annual trends in ABM-GIS Flood Evacuation Publications (2015–2025) depict the yearly distribution of the 67 studies included in this review.

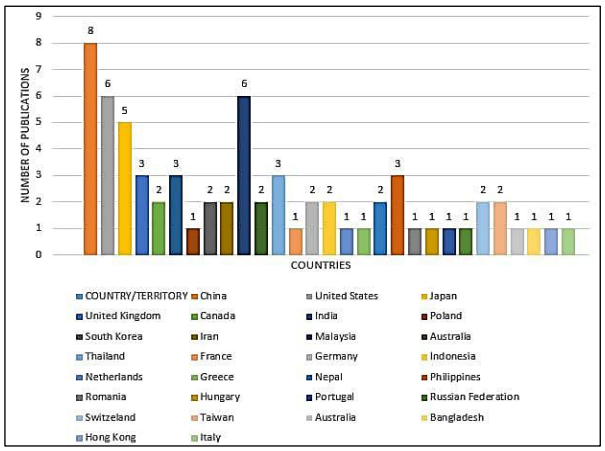


Fig. 3. Geographic distribution of ABM-GIS flood evacuation studies (2015–2025).

The data indicate that ABM-GIS flood evacuation remains heavily concentrated in high-income and upper-middle-income countries, particularly in East Asia, Western Europe, and North America. In contrast, low- and middle-income countries, despite being disproportionately affected by climate-induced flooding, remain underrepresented. This regional imbalance highlights a critical need for broader geographic engagement and increased research capacity in vulnerable regions such as Sub-Saharan Africa, Southeast Asia, and parts of Latin America.

C. Modeling Scale and Scope

The reviewed studies demonstrate a diverse application of modeling scales, reflecting the complexity and adaptability of ABM-GIS frameworks for flood evacuation scenarios. Modeling scale in this context refers to the level of agent aggregation and spatial resolution used to simulate evacuation dynamics, typically categorized as microscopic, mesoscopic, or macroscopic.

Among the 67 studies analyzed, mesoscopic models were the most prevalent, used in approximately 43% of cases. These

models typically simulate groups of agents such as households or neighborhoods within bounded geographic areas. Their popularity stems from a pragmatic balance between behavioral realism and computational efficiency, making them particularly suitable for urban-scale evacuations.

Microscopic models, accounting for 34% of studies, simulate individual agents with high granularity, often including variables such as age, mobility constraints, or decision-making heuristics. While offering the highest level of behavioral fidelity, such models are typically computationally intensive and often limited to smaller spatial domains or hypothetical case studies. Nevertheless, they provide critical insights into individual response variability, agent interaction, and localized congestion patterns.

Macroscopic models used in 23% of the studies represent populations in aggregate form, often using flow-based or zone-level simulation. These are typically coupled with hydrodynamic flood models and employed in large-scale risk assessments or regional planning exercises. While less behaviorally nuanced, macroscopic models are valuable for policy analysis and scenario testing over wide geographic extents.

The distribution of modeling scales reveals a methodological trade-off between computational scalability and behavioral detail. Few studies demonstrated multiscale integration (e.g., linking macro-level planning with micro-level agent behaviors), which represents an important direction for future research. Furthermore, only a small subset of models incorporated dynamic scaling, adjusting resolution based on the phase of the flood or real-time data availability, suggesting further potential for methodological innovation.

D. Data Fidelity and Integration

The quality, resolution, and dynamism of data inputs play a central role in the effectiveness and realism of ABM-GIS flood evacuation models. Across the 67 reviewed studies, there was substantial variation in how spatial, behavioral, and environmental data were sourced, integrated, and updated.

Approximately 61% of the studies relied on static or semi-static data inputs. These typically included pre-existing land use maps, census demographics, transportation networks, and historical flood records. While such inputs are useful for scenario testing or long-term planning, they often fail to capture the evolving dynamics of real-time flood events or adaptive human behavior under stress. Only 17.9% of the reviewed studies integrated real-time or near-real-time data such as live rainfall feeds, remote sensing imagery, traffic congestion updates, or sensor-based river level monitoring. These studies demonstrated enhanced situational responsiveness but often faced challenges related to data availability, system interoperability, and model latency.

In terms of data integration architecture, most models embedded GIS data as static spatial layers within ABM environments, primarily using raster-based flood extents or vector-based infrastructure maps. A smaller subset employed bidirectional coupling mechanisms where ABM and GIS platforms dynamically exchanged inputs and outputs during simulation runs, allowing for continuous updates to hazard extents or agent decisions in response to environmental change.

Notably, only a few models demonstrated multi-source data fusion, combining satellite imagery, crowd-sourced reports, mobile GPS data, and hydrological sensors. These approaches, while technically demanding, offer a pathway to more adaptive and realistic evacuation simulations. The uneven adoption of real-time and high-fidelity data reflects a persistent gap between methodological potential and technical feasibility. Bridging this gap will require not only improved access to interoperable data streams but also a greater emphasis on data assimilation techniques, modular system architectures, and validation of input reliability. To illustrate how data source types influence modeling approaches, Table V summarizes a subset of representative studies categorized by their dominant data inputs, modeling strategy, and associated references. The table highlights how data availability, ranging from static GIS layers to real-time sensor streams, influences the design and focus of ABM-GIS flood evacuation models.

TABLE V. REPRESENTATIVE STUDIES CATEGORIZED BY DATA SOURCES AND MODELING APPROACH.

No. of Studies	Common Data Sources	Dominant Modeling Approach	Representative References
12	DEM, LiDAR, hydrodynamic models	GIS and ABM hybrid flood simulation	[57], [58], [59], [60]
6	OpenStreetMap, Sentinel-2 imagery	GIS-based multi-criteria + agent routing	[30], [61], [62]
7	Remote sensing and survey data	Policy-oriented ABM and GIS adaptation models	[63], [64], [64]
4	Open data sets, DEM	Community-scale pedestrian ABM	[65], [66], [6]
3	Sensor data, urban hydrology	Traffic-focused ABM	[67], [68]
3	Mixed databases	Comparative framework reviews	[69], [70]

E. Validation Strategies

Validation remains a critical yet underdeveloped component in ABM-GIS flood evacuation modeling. Of the 67 studies reviewed, only 16.4% employed empirical validation techniques comparing model outputs to observed evacuation behavior, historical flood outcomes, or survey-based data. These studies typically used post-disaster reports, GPS traces, traffic sensor data, or retrospective interviews to calibrate agent behavior and verify model outputs.

The majority of studies (over 50%) relied on scenario-based or face validation, in which model behavior is assessed against expert expectations, hypothetical benchmarks, or visual plausibility. While valuable for internal consistency checks and stakeholder communication, such methods lack the rigor and reproducibility needed for high-confidence decision support.

Cross-dimensional analysis shows that empirical validation was more common in microscopic models (34%) than in mesoscopic (43%) and macroscopic (23%) models. High-fidelity data integration was also more frequently associated with scenario-based rather than empirical validation, indicating clustering of methodological sophistication rather than balanced integration.

A small number of studies (approximately 10%) applied cross-model validation, comparing their ABM-GIS results to outputs from other simulation platforms (e.g., hydrodynamic models or system dynamics). These efforts improved model credibility but often lacked formal metrics or sensitivity analyses. Several barriers to robust validation were noted across the literature:

- Limited availability of real-time or post-event behavioral data.
- Ethical and logistical challenges in collecting ground-truth evacuation information.
- Lack of standard performance metrics, especially for complex human-environment interactions.

Only a minority of studies reported using statistical goodness-of-fit tests, confusion matrices, or spatial accuracy metrics. In many cases, the validation process was either omitted entirely or vaguely described, undermining transparency and reproducibility.

The limited emphasis on empirical validation reflects a broader issue in disaster modeling: a methodological bias toward model development over model testing. Without rigorous validation, even sophisticated simulations risk becoming academic exercises rather than operational tools. Future research must prioritize data-grounded calibration, quantitative error reporting, and transparent validation protocols, particularly if ABM-GIS tools are to inform real-world policy and emergency response.

F. ABM-GIS Methodological Capabilities

The methodological capabilities observed across the 67 reviewed studies highlight both the strengths and limitations of current ABM-GIS integration practices in flood evacuation modeling. Fig. 4 synthesizes the core features evaluated, including spatial resolution, behavioral realism, data integration, validation, computational performance, and policy relevance.

Most studies (83.6%) employed high-resolution GIS data such as road networks, building footprints, land use, and elevation, enabling detailed analysis of agent movement, bottlenecks, and evacuation routes. Behavioral realism varied widely: 74.6% included basic movement or crowd dynamics, but only 38.8% modeled adaptive or heterogeneous decision-making based on factors like risk perception, social influence, or individual constraints. Few studies incorporated empirically derived behavioral parameters or hazard-dependent scenario switching.

Regarding data integration, 65.7% of studies relied on static GIS layers, while only 17.9% integrated real-time or dynamic data sources. This gap continues to constrain the operational applicability of many models, especially in fast-evolving flood contexts. Model validation, as discussed in Section IV E, was another area of concern. While 43.3% of models conducted internal validation (e.g., sensitivity or scenario testing), only 16.4% employed empirical validation against observed data. This underscores the need for greater emphasis on robust, data-informed calibration and performance assessment.

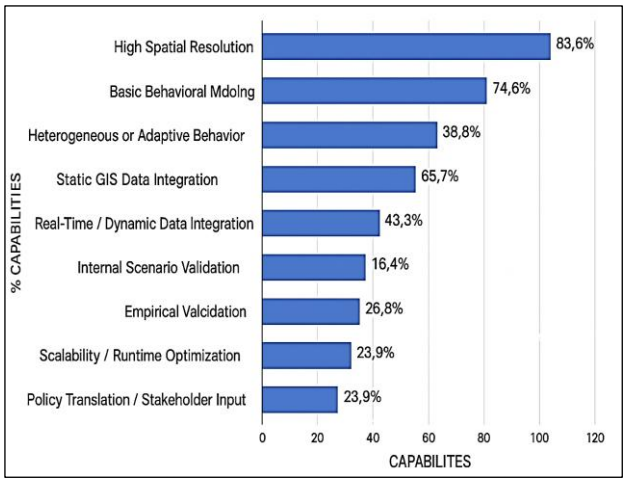


Fig. 4. Prevalence of key methodological capabilities across reviewed ABM-GIS Flood Evacuation Studies illustrates the percentage of studies (n = 67) exhibiting specific modeling capabilities, highlighting strengths in spatial resolution and basic behavior modeling, and gaps in real-time data integration and empirical validation.

In terms of computational performance, only 26.8% of studies addressed scalability or runtime efficiency, indicating that most models remain conceptual rather than real-time deployable. Similarly, policy translation and stakeholder engagement were limited (23.9%), raising concerns about practical relevance and adoption.

Overall, the analysis of methodological capabilities reveals a field that is advancing in spatial and behavioral complexity but still constrained by gaps in real-time responsiveness, empirical grounding, and stakeholder co-design. Addressing these limitations is essential for transitioning ABM-GIS systems from academic prototypes to operational decision-support tools.

To illustrate the diversity and depth of ABM-GIS modeling approaches in real-world contexts, Table VI summarizes selected case studies, outlining their integration frameworks, modeling focus, key contributions, and limitations, highlighting both methodological diversity and persistent gaps in realism, validation, and scalability.

G. Framework for Integrating Scale, Data, and Validation in ABM-GIS Models

To consolidate the methodological diversity observed across the 67 studies, a Conceptual Integration Framework was developed. This framework synthesizes findings across three core analytical dimensions modeling scale, data fidelity, and validation strategy and situates each study within a structured landscape of ABM-GIS flood evacuation modeling practices.

The modeling scale dimension captures the granularity of agent representation, distinguishing between microscopic models (individual-level), mesoscopic models (group-based), and macroscopic models (population aggregates). As described in Section IV C, mesoscopic approaches dominated the literature (43%), offering a balance between realism and computational tractability.

TABLE VI. COMPARATIVE SUMMARY OF SELECTED ABM-GIS FLOOD EVACUATION STUDIES

Author (Year)	Study Area	Model / Framework	Integration Focus	Key Findings / Contributions	Identified Limitations
[28]	Thailand	TU-FS Simulator	GIS-based flood mapping with ABM crowd simulation	Modeled large-scale pedestrian movement to optimize routes and shelter allocation during floods.	Limited to urban settings; did not consider behavioral diversity or multi-hazard context.
[29]	Nanning, China	SWMM-LISFLOOD + ABM	Hydrodynamic and GIS data coupled with ABM	Assessed pedestrian vulnerability and identified high-risk zones under dynamic flooding.	High computational cost; calibration challenges with real-time data.
[7]	Wuhan, China	Coupled Agent-Based Multi-Model Framework (CABMF)	Traffic and shelter data integration via GIS	Tested alternative evacuation strategies using spatially explicit ABM modeling.	Heavy computational demand; limited behavioral validation.
[6]	Kibera, Kenya	OpenStreetMap + DEM + ABM	Community-scale pedestrian simulation	Identified congestion points and suboptimal shelters in informal settlements.	Low-resolution open-source data; constrained by local data quality.
[30]	Nepal	GIS-based Analytic Hierarchy Process (AHP) + ABM	Flood-susceptibility mapping integrated with agent decisions	Prioritized evacuation corridors for high-risk areas; applicable with limited data.	Low-resolution DEM reduced route accuracy; static behavioral rules.
[31]	Coastal China	GIS + ABM for tsunami evacuation	Multi-hazard coupling (flood + landslide + liquefaction)	Demonstrated cross-hazard modeling capability within ABM and GIS framework.	Focused on specific hazard; limited to simulation, not validation.
[32]	China (Urban Metro)	GIS and ABM Transit Model	Infrastructure resilience analysis	Evaluated flood impacts on metro networks and rerouting efficiency.	Did not model passenger behavior diversity.
[33]	China (National Review)	Review of ABM and GIS frameworks	Policy-oriented, climate-adaptation focus	Highlighted potential for long-term risk and insurance modeling.	Conceptual; lacked empirical validation.
[34]	Hong Kong, China	ABM using AnyLogic platform integrated with GIS	Simulates self-evacuation under infrastructure failure	Developed an ABM incorporating household behavior diversity and infrastructure disruption.	Limited empirical validation
[35]	La Ciotat, France	SiFlo Model	GIS and ABM impact modeling	Quantified how awareness and maintenance reduce casualties (~30%).	Regional focus; simplified social behavior.
[36]	United Kingdom	Life Safety Model (LSM)	Dam-break simulation with GIS-ABM coupling	Measured effects of evacuation delay and congestion on mortality.	Requires high-quality DEM; computationally intensive.
[37]	Iran	Hybrid BIM and GIS and ABM	Behavioral and psychological modeling	Incorporated stress and social dynamics in evacuation after earthquakes.	Applied to earthquake context; limited flood transferability.

The data fidelity dimension assesses the type and dynamism of input data used. Models were classified as using low, medium, or high-fidelity data based on their use of static versus real-time information, sensor integration, and spatial resolution. Only 17.9% of studies achieved high data fidelity through real-time or dynamic input streams.

The third axis, validation strategy, reflects the extent and rigor of model verification. It distinguishes between empirical validation (against real-world observations), scenario-based validation, and no formal validation. Most studies (over 50%) relied on scenario-based validation, while only 16.4% employed empirical benchmarks.

By plotting studies across these three axes, the framework reveals distinct clusters and gaps. A central trend is the convergence toward spatially detailed, behaviorally rich, yet weakly validated models indicating methodological innovation without sufficient empirical grounding. Very few studies simultaneously demonstrated high fidelity, individual-level modeling, and rigorous validation.

The framework also exposes a structural limitation: few studies adopted hybrid approaches that span multiple scales, integrate dynamic data, and validate across both technical and behavioral dimensions. This lack of integration suggests that while the field is advancing in specific areas, systemic operational maturity remains limited. Ultimately, the Conceptual Integration Framework, as illustrated in Fig. 5, offers a diagnostic tool for positioning current research, identifying best practices, and directing future efforts toward comprehensive, validated, and policy-relevant ABM-GIS flood evacuation systems.

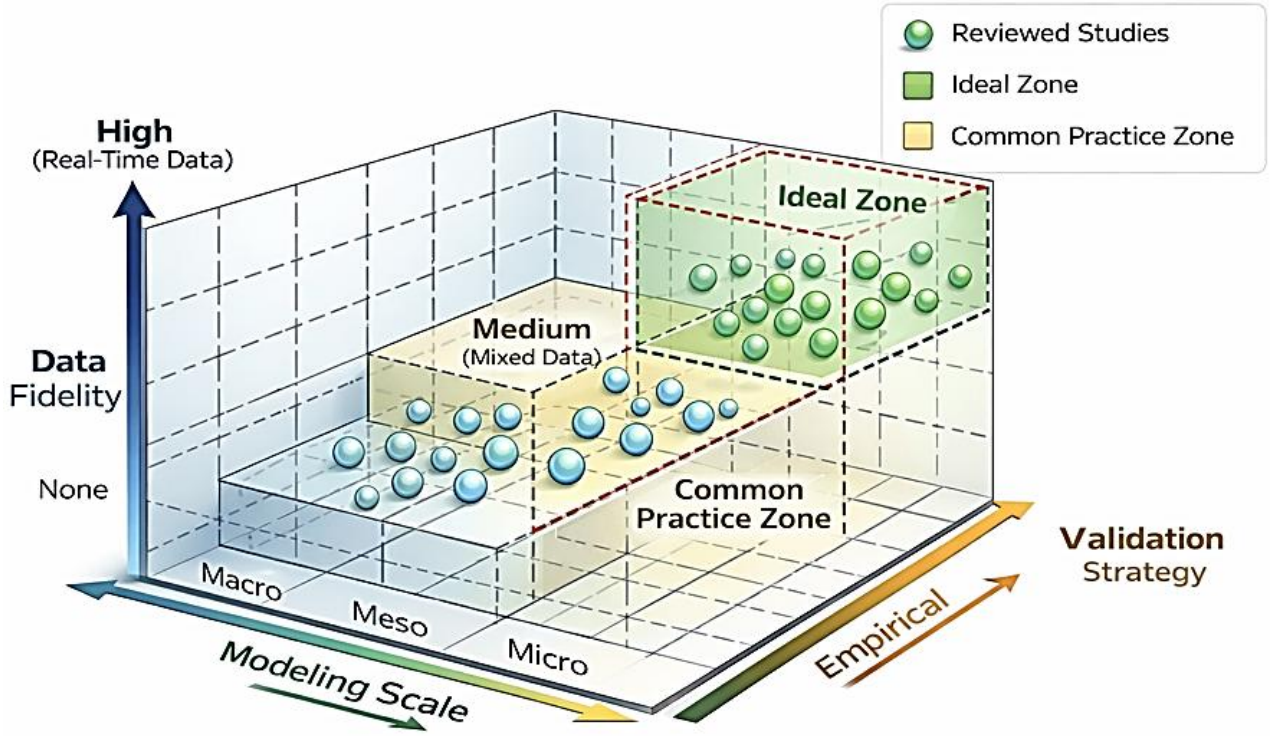


Fig. 5. Conceptual Integration Framework for ABM-GIS Flood Evacuation Models: A three-dimensional framework categorizing the 67 reviewed studies across modeling scale (macroscopic to microscopic), data fidelity (static to real-time), and validation strategy (none to empirical). The framework highlights dominant methodological clusters and identifies underdeveloped areas requiring future research.

H. Summary of Key Findings

The results of this review reveal a rapidly evolving field characterized by methodological diversity and growing sophistication in the integration of agent-based modeling and GIS for flood evacuation planning. While notable progress has been made in spatial resolution, behavioral modeling, and scenario-based applications, significant limitations persist. These include limited real-time data integration, inconsistent validation practices, and a lack of stakeholder engagement in model development. The Conceptual Integration Framework highlights clear trade-offs between fidelity, scale, and validation rigor emphasizing the need for more holistic, empirically grounded approaches. These findings collectively underscore a critical transition point for the field: from exploratory modeling toward operational decision-support systems that are robust, scalable, and grounded in real-world dynamics.

V. Discussion

The systematic review of 67 ABM-GIS flood evacuation studies demonstrates growing methodological innovation, particularly in spatial modeling and agent-based behavior representation. However, significant gaps persist in empirical grounding, real-time responsiveness, and operational integration. This discussion contextualizes those findings within current research trajectories and highlights the critical steps needed to evolve ABM-GIS models from academic prototypes into policy-relevant, adaptive systems.

A. Integration Advancements and Spatial Maturity

The field has matured notably in spatial fidelity and model sophistication. Over 80% of reviewed studies employed high-resolution GIS inputs such as DEMs, land use classifications, or detailed infrastructure networks. This reflects the increasing availability and accessibility of geospatial data and hydrodynamic models such as HEC-RAS or SWMM, which enhance terrain realism and flood dynamics. In many cases, spatial resolution enabled evacuation models to operate at building-level or street-level granularity, essential for identifying bottlenecks, exit points, and vulnerable zones in urban areas.

Concurrently, integration of GIS within ABM has improved in terms of layering, routing, and environment-agent interactions. However, in most studies, GIS is still used as a static input environment rather than a dynamically updating system. Real-time integration remains rare, and the use of GIS in feedback loops, such as adjusting flood depth or road blockage during simulation, is still largely absent.

B. Behavioral Realism and Modeling Scale Limitations

In terms of agent behavior, progress is mixed. While approximately 75% of studies modeled individual or group-level movement decisions, only a minority (less than 40%) introduced behavioral heterogeneity or psychological dynamics. Most models used deterministic rules (e.g., shortest-path algorithms or fixed compliance rates), overlooking decision uncertainty, social influence, or delayed response.

This undermines realism, as empirical studies show that evacuation behavior is often influenced by cascading cues, misinformation, or peer actions. There is a critical opportunity here to align ABM-GIS models with behavioral science, including risk perception theories and social contagion frameworks, to better emulate real-world complexities.

Additionally, the dominance of mesoscopic models, while efficient, may obscure fine-grained dynamics in dense urban environments or neglect macro-level impacts such as inter-jurisdictional flows. Multiscale or hybrid modeling approaches blending microscopic agent actions with regional-scale scenarios remain underutilized.

C. Validation and Data Assimilation Remain Critical Bottlenecks

Validation remains one of the weakest links in current ABM-GIS modeling. Despite methodological rigor in model design, only 16.4% of studies employed empirical validation strategies. The rest relied on scenario testing or visual plausibility. This validation gap weakens the credibility of simulation outputs and limits their utility in high-stakes planning contexts.

The main barriers include a lack of accessible post-event behavioral datasets, ethical challenges in observing real-time evacuations, and difficulty in matching dynamic model outputs with real-world baselines. Although some studies used field surveys or traffic counts, few integrated sensor-based evacuation data or mobile trace analytics.

Similarly, only a small subset of studies (17.9%) integrated real-time data sources (e.g., IoT sensors, remote sensing imagery, live weather feeds), despite the growing availability of such streams. Without real-time data assimilation, models remain static representations rather than responsive decision-support systems.

D. Limitations of the Review Methodology

Although this review followed PRISMA guidelines to ensure transparency in study identification and selection, a formal quality or risk-of-bias assessment tool (e.g., MMAT or GRADE) was not applied. This decision was primarily due to the methodological heterogeneity of the included studies, which span simulation-based modeling, hydrodynamic coupling, conceptual frameworks, and computational experiments that do not align neatly with conventional appraisal instruments designed for clinical or empirical research.

Nevertheless, this absence may influence the interpretation of research gaps, particularly regarding validation rigor. The reported validation prevalence reflects reporting patterns rather than a weighted assessment of study robustness. Future reviews would benefit from appraisal criteria specifically designed for ABM-GIS simulation research.

E. Policy Translation and Stakeholder Integration are Underdeveloped

Despite the theoretical potential of ABM-GIS models, only 23.9% of studies involved stakeholders or addressed implementation pathways. Even fewer aligned simulations with policy frameworks, emergency management protocols, or operational workflows. This represents a disconnect between scientific modeling and practical deployment.

Effective policy integration requires not just accurate models, but interpretable outputs, user-friendly interfaces, and stakeholder trust. Most reviewed studies stopped at visualizations or dashboards, without clear evidence of their use in drills, response plans, or municipal decision-making. Co-design processes, scenario walkthroughs, and participatory simulation remain the exception rather than the norm.

F. Research Priorities and Strategic Directions

To advance ABM-GIS flood evacuation modeling, five strategic research priorities emerge from the reviewed literature and are illustrated as follows:

1) Real-time IoT integration: Few models dynamically respond to live flood or mobility data. Integrating IoT sensors, satellite feeds, and traffic APIs would enable real-time updates to road conditions and hazard zones, improving the responsiveness of simulations.

2) Behavioral model enrichment: Agent behaviors are often simplistic, ignoring stress, uncertainty, or social influence. Future models should incorporate cognitive and psychological dimensions to better reflect real-world evacuation patterns.

3) Machine learning integration: ML methods can enhance model calibration, pattern recognition, and adaptive agent behavior. Reinforcement learning, in particular, can help agents learn optimal escape strategies over time, though interpretability must be preserved.

4) Multi-hazard modeling: Most simulations focus on single hazards. Incorporating cascading risks (e.g., power outages, pandemics) would allow for more realistic crisis planning and assessment of compound disaster scenarios.

5) Cloud computing and digital twins: Scalability remains a bottleneck. Cloud platforms and GPU-accelerated simulations can enable high-resolution, real-time modeling. Digital twins could support continuous monitoring and simulation of evolving emergencies.

This concise roadmap aligns emerging technologies with current modeling limitations, offering a practical foundation for building more adaptive, validated, and operationally integrated ABM-GIS systems

The reviewed studies and frameworks reveal both progress and ongoing gaps in ABM-GIS flood evacuation modeling. To summarize these findings, Fig. 6 presents key modeling components, typical methods, and recurring limitations offering a final lens to identify areas of innovation and those needing further development.

Table VII presents a strategic mapping of core challenges in ABM-GIS flood evacuation modeling alongside recommended solutions and anticipated outcomes. This synthesis aims to guide future work in developing empirically grounded, computationally efficient, and socially relevant simulation systems.

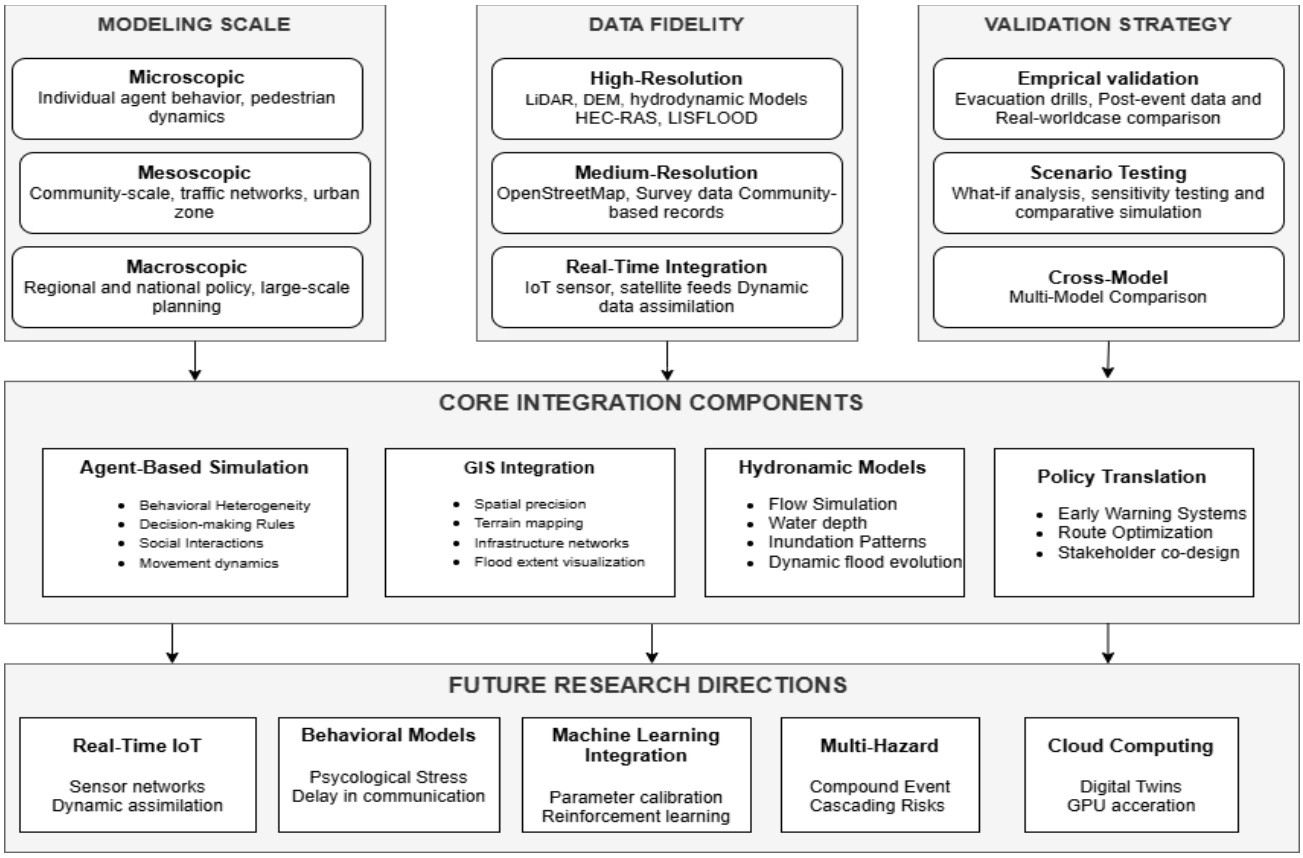


Fig. 6. Synthesis of ABM-GIS Modeling Practices, Core Integration Components, and Future Research Directions summarizes key dimensions of modeling scale, data fidelity, and validation strategy in ABM-GIS flood evacuation studies. The middle layer identifies essential integration components: agent simulation, GIS coupling, hydrodynamic modeling, and policy interfaces, while the bottom layer highlights six research priorities critical for advancing the field.

TABLE VII. STRATEGIC CHALLENGES, RECOMMENDED SOLUTIONS, AND EXPECTED OUTCOMES FOR ADVANCING ABM-GIS FLOOD EVACUATION MODELING

Research Focus	Key Challenge	Recommended Strategy	Expected Outcome	Ref.
Real-Time Data Integration	Static datasets limit responsiveness	Adopt IoT, UAV imagery, and satellite streams	Dynamic updates to ABM and GIS layers	[38], [39], [40]
Machine Learning Integration	Manual parameter tuning	Use reinforcement learning for behavioral adaptation	Self-calibrating models with faster simulation speeds	[41], [42], [43]
Multi-Hazard Modeling	Single-event focus	Link hydrologic, meteorologic, and infrastructure datasets	Cascading risk assessment	[44], [45], [46]
Behavioral Realism	Simplified decision rules	Integrate social-network and psychological factors	Culturally sensitive evacuation policies	[47], [48], [49]
Computational Scalability	High processing time	Parallel computing + cloud deployment	Real-time simulation capability	[50], [51], [52]
Validation Standards	Lack of benchmarks	Develop shared post-event datasets	Cross-model comparability	[53]
Policy Co-Design	Weak research	Participatory scenario planning with local agencies	Higher adoption of ABM and GIS outputs	[54], [55], [56]

VI. CONCLUSION

This review has critically examined 67 peer-reviewed studies that integrate ABM and GIS in the context of flood evacuation. The findings reveal clear progress in spatial resolution, agent-based behavior modeling, and hydrodynamic coupling. However, several persistent limitations hinder the operational utility of these models, including weak empirical validation, limited behavioral diversity, insufficient real-time data integration, and low policy engagement.

Through structured thematic synthesis and conceptual frameworks, the review identifies three core modeling dimensions: scale, data fidelity, and validation strategy that define the methodological landscape. Most studies emphasize spatial detail and heuristic agent behavior, yet few achieve comprehensive integration across all dimensions. Most remain static, proof-of-concept systems rather than fully adaptive, operational decision-support tools.

To address these challenges, the study outlines five strategic research priorities: real-time IoT integration, behavioral model enrichment, machine learning, multi-hazard coupling, and scalable cloud-based deployment. These priorities are not merely technical enhancements; they

represent critical shifts toward empirical rigor, stakeholder relevance, and cross-disciplinary innovation.

As climate-driven flood risks intensify globally, the need for robust, adaptive, and context-sensitive evacuation models has never been more urgent. Future ABM-GIS systems must move beyond academic silos to become embedded components of anticipatory planning, real-time response, and long-term resilience strategies. This review provides a roadmap for that transition, bridging science, computation, and policy to strengthen disaster preparedness in the face of escalating uncertainty.

ACKNOWLEDGMENT

This research is supported and funded by the University Malaysia Pahang Al-Sultan Abdullah, Malaysia, under the Postgraduate Research Grant Scheme.

REFERENCES

[1] M. Lippe et al., "Using agent-based modelling to simulate social-ecological systems across scales," Geoinformatica, vol. 23, no. 2, pp. 269–298, Apr. 2019, doi: 10.1007/s10707-018-00337-8.

[2] J. Wang and N. Wang, "Forecasting road network functionality states during extreme rainfall events to facilitate real-time emergency response planning," Reliab Eng Syst Saf, vol. 252, Dec. 2024, doi: 10.1016/j.ress.2024.110452.

[3] C. Yaibok et al., "Synergizing flood risk and road network dynamics for optimized evacuation strategies," Asian Transport Studies, vol. 10, Jan. 2024, doi: 10.1016/j.eastsj.2024.100137.

[4] S. Z. Golazad, G. Heravi, A. AminShokravi, and A. Mohammadi, "Integrating GIS, agent-based, and discrete event simulation to evaluate patient distribution policies for enhancing urban healthcare access network resilience," Sustain Cities Soc, vol. 111, Sep. 2024, doi: 10.1016/j.scs.2024.105559.

[5] A. Couasnon, A. Sebastian, and O. Morales-Nápoles, "A Copula-Based Bayesian Network for Modeling Compound Flood Hazard from Riverine and Coastal Interactions at the Catchment Scale: An Application to the Houston Ship Channel, Texas," Water (Basel), vol. 10, no. 9, p. 1190, Sep. 2018, doi: 10.3390/w10091190.

[6] O. Butters and R. J. Dawson, "Flood Evacuation in Informal Settlements: Application of an Agent-Based Model to Kibera Using Open Data," Urban Science, vol. 9, no. 1, Jan. 2025, doi: 10.3390/urbansci9010012.

[7] W. Lv, F. Deng, J. Wang, Y. Han, and S. Yang, "Large-scale real-time evacuation modeling during urban floods: A coupled agent-based multi-model framework," Simul Model Pract Theory, vol. 140, Apr. 2025, doi: 10.1016/j.simpat.2025.103075.

[8] M. A. Majid and U. Aickelin, "Comparing Simulation Output Accuracy of Discrete Event and Agent Based Models: A Quantitative Approach."

[9] B. Kleinmeier, G. Köster, and J. Drury, "Agent-based simulation of collective cooperation: From experiment to model: Agent-based simulation of collective cooperation: From experiment to model," J R Soc Interface, vol. 17, no. 171, Oct. 2020, doi: 10.1098/rsif.2020.0396.

[10] S. Khan and Z. Deng, "Agent-based crowd simulation: an in-depth survey of determining factors for heterogeneous behavior," Visual Computer, vol. 40, no. 7, pp. 4993–5004, Jul. 2024, doi: 10.1007/s00371-024-03503-2.

[11] O. Urbina, H. S. Sousa, A. Fekete, J. C. Matos, and E. Teixeira, "Spatial Vulnerability Assessment of Critical Infrastructure Based on Fire Risk through GIS Systems—Case Study: Historic City Center of Guimarães, Portugal," Applied Sciences (Switzerland), vol. 13, no. 15, Aug. 2023, doi: 10.3390/app13158881.

[12] X. Chen, "Agent-based modeling of emergency evacuation in a spatially-aware and time-aware environment," in Proc. 25th Int. Cartographic Conf. (ICC), Jul. 2011, pp. 3–8.

[13] A. Nuhu. F. M. Raffei, M. F. Ab Razak, and A. Ahmad, "Distributed denial of service attack detection in IoT networks using deep learning and feature fusion: A review," Mesopotamian Journal of CyberSecurity, vol. 4, no. 1, pp. 47–70, 2024.

[14] Q. N. Hong, M. Bangpan, C. Stansfield, D. Kneale, A. O'Mara‐Eves, L. van Grootel, and J. Thomas, "Using systems perspectives in evidence synthesis: A methodological mapping review," Research Synthesis Methods, vol. 13, no. 6, pp. 667‐680, 2022

[15] N. Hamza, M. A. Majid, and F. Hujainah, "SIM-PFED: A Simulation-Based Decision Making Model of Patient Flow for Improving Patient Throughput Time in Emergency Department," IEEE Access, vol. 9, pp. 103419–103439, 2021, doi: 10.1109/ACCESS.2021.3098625.

[16] N. Akma, A. Bakar, and M. A. Majid, "Modelling Human Evacuation Egress using Simulation Techniques: A Review," Proceedings of Malaysian Technical Universities Conference on Engineering and Technology (MUCET) 2021

[17] D. V. Parums, "Editorial: Review articles, systematic reviews, meta-analysis, and the updated preferred reporting items for systematic reviews and meta-analyses (PRISMA) 2020 Guidelines," Aug. 24, 2021, International Scientific Information, Inc. doi: 10.12659/MSM.934475.

[18] D. Carramiñana, A. M. Bernardos, J. A. Besada, and J. R. Casar, "Towards resilient cities: A hybrid simulation framework for risk mitigation through data-driven decision making," Simul Model Pract Theory, vol. 133, May 2024, doi: 10.1016/j.simpat.2024.102924.

[19] G. P. D. P. Senanayake, M. Kieu, Y. Zou, and K. Dirks, "Agent-based simulation for pedestrian evacuation: A systematic literature review," Sep. 01, 2024, Elsevier Ltd. doi: 10.1016/j.ijdrr.2024.104705.

[20] E. Avila-Aceves, W. Plata-Rocha, S. A. Monjardin-Armenta, and J. G. Rangel-Peraza, "Geospatial modelling of floods: a literature review," Nov. 01, 2023, Springer Science and Business Media Deutschland GmbH. doi: 10.1007/s00477-023-02505-1.

[21] A. Anshuka, F. F. van Ogtrop, D. Sanderson, and S. Z. Leao, "A systematic review of agent-based model for flood risk management and assessment using the ODD protocol," Natural Hazards, vol. 112, no. 3, pp. 2739–2771, Jul. 2022, doi: 10.1007/s11069-022-05286-y.

[22] F. Ahmad and L. Al-Fagih, "Game Theory Applications in Micro and Macroscopic Simulations in Transportation Networks: A Comprehensive Review," 2023, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ACCESS.2023.3308048.

[23] N. Kaur and H. Kaur, "A Multi-Agent Based Evacuation Planning for Disaster Management: A Narrative Review," 2022, Springer Science and Business Media B.V. doi: 10.1007/s11831-022-09729-4.

[24] M. Canales, J. Castilla-Rho, R. Rojas, S. Vicuña, and J. Ball, "Agent-Based Models of groundwater systems - Database," Feb. 21, 2024, HydroShare. doi: 10.4211/hs.5e6cc59ec95b4aa2bcc8f0be8f8832fb.

[25] R. G. N. Lakmali, P. V. Genovese, and A. A. B. D. P. Abewardhana, "Evaluating the Efficacy of Agent-Based Modeling in Analyzing Pedestrian Dynamics within the Built Environment: A Comprehensive Systematic Literature Review," Jul. 01, 2024, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/buildings14071945.

[26] G. Business and M. Research, "Enhancing Disaster Management for Human Emergency Evacuation using an Integration of the Social Force Model and Agent-Based Simulations," 2025.

[27] K. Wikstrom and H. T. Nelson, "Spatial Validation of Agent-Based Models," Sustainability (Switzerland), vol. 14, no. 24, Dec. 2022, doi: 10.3390/su142416623.

[28] ICAICTA 2015: the 2015 International Conference on Advanced Informatics: Concepts, Theory and Application: August 19-22, 2015, the Tide Resort, Bang Saen Beach, Chonburi, Thailand. IEEE, 2015.

[29] H. Lan et al., "Dynamic disaster risk assessment of urban waterlogging on pedestrian flow by intelligent simulation of hydrodynamics coupled with agent-based models in Chao-yang river basin of Nanning, China," Stochastic Environmental Research and Risk Assessment, Oct. 2024, doi: 10.1007/s00477-024-02782-4.

[30] G. Parajuli, S. Neupane, S. Kunwar, R. Adhikari, and T. D. Acharya, "A GIS-Based Evacuation Route Planning in Flood-Susceptible Area of Siraha Municipality, Nepal," ISPRS Int J Geoinf, vol. 12, no. 7, Jul. 2023, doi: 10.3390/ijgi12070286.

[31] C. Chen, C. Koll, H. Wang, and M. K. Lindell, "An interdisciplinary agent-based evacuation model: integrating the natural environment, built environment, and social system for community preparedness and resilience," Natural Hazards and Earth System Sciences, vol. 23, no. 2, pp. 733–749, Feb. 2023, doi: 10.5194/nhess-23-733-2023.

[32] B. Zhao, Y. Tang, C. Wang, S. Zhang, and K. Soga, "Evaluating the flooding level impacts on urban metro networks and travel demand: behavioral analyses, agent-based simulation, and large-scale case study," Resilient Cities and Structures, vol. 1, no. 3, pp. 12–23, Sep. 2022, doi: 10.1016/j.rcns.2022.10.004.

[33] L. Zhuo and D. Han, "Agent-based modelling and flood risk management: A compendious literature review," Dec. 01, 2020, Elsevier B.V. doi: 10.1016/j.jhydrol.2020.125600.

[34] A. M. Edjossan-Sossou, M. Vuillet, R. Mehdizadeh, and O. Deck, "Agent-based model for simulating households' self-evacuation decision in high-rise buildings under critical infrastructure failures induced by a slow-onset flood conditions – A case study in Paris," IDRiM Journal, vol. 10, no. 2, Nov. 2020, doi: 10.5595/001c.18160.

[35] F. Taillandier, P. Di Maiolo, P. Taillandier, C. Jacquenod, L. Rauscher-Lauranceau, and R. Mehdizadeh, "An agent-based model to simulate inhabitants' behavior during a flood event," International Journal of Disaster Risk Reduction, vol. 64, Oct. 2021, doi: 10.1016/j.ijdrr.2021.102503.

[36] D. Lumbroso, M. Davison, and M. Wetton, "Development of an agent-based model to improve emergency planning for floods and dam failures," Journal of Hydroinformatics, vol. 25, no. 5, pp. 1610–1628, Sep. 2023, doi: 10.2166/hydro.2023.194.

[37] S. Hassanpour, V. Gonzalez, J. Liu, Y. Zou, and G. Cabrera-Guerrero, "A hybrid hierarchical agent-based simulation approach for buildings indoor layout evaluation based on the post-earthquake evacuation," Advanced Engineering Informatics, vol. 51, Jan. 2022, doi: 10.1016/j.aei.2022.101531.

[38] Z. Wang, "Technological Integration in Urban Emergency Management: Challenges, Innovations, and Pathways to Resilient Systems," 2025. [Online]. Available: https://cpcig-conferences.com/index.php/setp

[39] N. Tyagi, J. Singh, S. Singh, and S. S. Sehra, "A 3D Model-Based Framework for Real-Time Emergency Evacuation Using GIS and IoT Devices," ISPRS Int J Geoinf, vol. 13, no. 12, Dec. 2024, doi: 10.3390/ijgi13120445.

[40] M. Priyaa and D. Sridhar, "Real-Time Flood Mapping for Disaster Management Decision Support in Chennai "ARCHIVES," 2017.

[41] J. Nousiainen et al., "Toward on-sky adaptive optics control using reinforcement learning: Model-based policy optimization for adaptive optics," Astron Astrophys, vol. 664, Aug. 2022, doi: 10.1051/0004-6361/202243311.

[42] J. Nousiainen, C. Rajani, M. Kasper, and T. Helin, "Adaptive optics control using model-based reinforcement learning," Opt Express, vol. 29, no. 10, p. 15327, May 2021, doi: 10.1364/oe.420270.

[43] A. Xavier Fidêncio, C. Klaes, and I. Iossifidis, "Error-Related Potentials in Reinforcement Learning-Based Brain-Machine Interfaces," Jun. 24, 2022, Frontiers Media S.A. doi: 10.3389/fnhum.2022.806517.

[44] C. Huang, S. Nie, Y. R. Fan, L. R. Liu, T. N. Song, and W. Huang, "An Inexact Copula-Based Stochastic Fractional Programming Model for Planning Emergency Evacuation at Nuclear Power Plant Sites," Journal of Environmental Informatics, vol. 46, no. 1, 2025, doi: 10.3808/jei.202500541.

[45] "Integrated Multi-Hazard and Vulnerability Modelling for Flood Risk Assessment in the US Gulf Coast."

[46] S. Scholarship@western, S. Bakhtiari, S.: Najafi, and M. Reza, "Multi-Hazard Risk Assessment of the Interconnected Multi-Hazard Risk Assessment of the Interconnected Infrastructure Systems Infrastructure Systems," 2024. [Online]. Available: https://ir.lib.uwo.ca/etdhttps://ir.lib.uwo.ca/etd/10494

[47] T. Haer, W. J. W. Botzen, and J. C. J. H. Aerts, "The effectiveness of flood risk communication strategies and the influence of social networks-Insights from an agent-based model," Environ Sci Policy, vol. 60, pp. 44–52, Jun. 2016, doi: 10.1016/j.envsci.2016.03.006.

[48] C. Bulumulla, "Integrating Social Network Diffusion Into BDI-based Simulations: Application Focus on Large-scale Evacuations," 2022.

[49] H. Moradi et al., "Enhancing Disaster Evacuation Planning with Cognitive Agent-Based Models and Co-Creation," European Journal of Sustainable Development, vol. 14, pp. 153–167, 2025, doi: 10.14207/ejsd. 2025.v14n3p153.

[50] Q. Liu, Y. Qin, and G. Li, "Fast simulation of large-scale floods based on GPU parallel computing," Water (Switzerland), vol. 10, no. 5, May 2018, doi: 10.3390/w10050589.

[51] J. Wei et al., "Enable high-resolution, real-time ensemble simulation and data assimilation of flood inundation using distributed GPU parallelization," J Hydrol (Amst), vol. 619, Apr. 2023, doi: 10.1016/j.jhydrol.2023.129277.

[52] M. M. Morsy et al., "A cloud-based flood warning system for forecasting impacts to transportation infrastructure systems," Environmental Modelling and Software, vol. 107, pp. 231–244, Sep. 2018, doi: 10.1016/j.envsoft.2018.05.007.

[53] C. Lowrie, A. Kruczkiewicz, S. N. McClain, M. Nielsen, and S. J. Mason, "Evaluating the usefulness of VGI from Waze for the reporting of flash floods," Sci Rep, vol. 12, no. 1, Dec. 2022, doi: 10.1038/s41598-022-08751-7.

[54] A. Bustillos Ardaya, M. Evers, and L. Ribbe, "Participatory approaches for disaster risk governance? Exploring participatory mechanisms and mapping to close the communication gap between population living in flood risk areas and authorities in Nova Friburgo Municipality, RJ, Brazil," Land use policy, vol. 88, Nov. 2019, doi: 10.1016/j.landusepol.2019.104103.

[55] M. Pawlik et al., "Agent-based modelling for flood evacuation as inclusive disaster risk reduction: Pilot participatory action research with 11- and 12-year-old children from a Japanese school," International Journal of Disaster Risk Reduction, vol. 117, Feb. 2025, doi: 10.1016/j.ijdrr.2025.105180.

[56] A. M. F. Lagmay et al., "An Impact-Based Flood Forecasting System for Citizen Empowerment," Asian Journal of Agriculture and Development, vol. 21, no. SAI, pp. 129–148, Oct. 2024, doi: 10.37801/ajad2024.21.20AI.8.

[57] J. Teng, J. Vaze, D. Dutta, and S. Marvanek, "Rapid Inundation Modelling in Large Floodplains Using LiDAR DEM," Water Resources Management, vol. 29, no. 8, pp. 2619–2636, Jun. 2015, doi: 10.1007/s11269-015-0960-8.

[58] C. Frizzle, M. Trudel, S. Daniel, A. Pruneau, and J. Noman, "LiDAR topo-bathymetry for riverbed elevation assessment: A review of approaches and performance for hydrodynamic modelling of flood plains," Earth Surf Process Landf, vol. 49, no. 9, pp. 2585–2600, Jul. 2024, doi: 10.1002/esp.5808.

[59] E. M. Prior, N. Michaelson, J. A. Czuba, T. J. Pingel, V. A. Thomas, and W. C. Hession, "Lidar DEM and Computational Mesh Grid Resolutions Modify Roughness in 2D Hydrodynamic Models," Water Resour Res, vol. 60, no. 7, Jul. 2024, doi: 10.1029/2024WR037165.

[60] N. A. Muhadi, A. F. Abdullah, S. K. Bejo, M. R. Mahadi, and A. Mijic, "The use of LiDAR-derived DEM in flood applications: A review," Jul. 01, 2020, MDPI AG. doi: 10.3390/rs12142308.

[61] S. AL SALAH, "A GIS-MCDA-Based Analysis for Spatial Ecotourism Suitability Assessment in Saudi Arabia's Hail Province," Gazi University Journal of Science Part A: Engineering and Innovation, vol. 10, no. 3, pp. 361–377, Sep. 2023, doi: 10.54287/gujsa.1328036.

[62] H. Li et al., "Automatic mapping of national surface water with OpenStreetMap and Sentinel-2 MSI data using deep learning," International Journal of Applied Earth Observation and Geoinformation, vol. 104, Dec. 2021, doi: 10.1016/j.jag.2021.102571.

[63] Md. S. Sheikh, Md. A. H. Babu, S. S. Nath, M. J. Uddin, and M. N. A. Ansari, "Mapping the Research Landscape of Land Use and Land Cover Change: A Bibliometric Analysis," MBSTU Journal of Science and Technology, vol. 11, no. 1, pp. 18–30, May 2025, doi: 10.69728/jst. v11.61.

[64] S. Puttinaovarat and P. Horkaew, "Internetworking flood disaster mitigation system based on remote sensing and mobile GIS," Geomatics, Natural Hazards and Risk, vol. 11, no. 1, pp. 1886–1911, Jan. 2020, doi: 10.1080/19475705.2020.1815869.

[65] F. I. W. Rohmat, Z. Sa'adi, I. Stamataki, A. A. Kuntoro, M. Farid, and R. Suwarman, "Flood modeling and baseline study in urban and high

population environment: A case study of Majalaya, Indonesia," Urban Clim, vol. 46, Dec. 2022, doi: 10.1016/j.uclim.2022.101332.

[66] R. Bin Mostafiz et al., "Comparison of Neighborhood-Scale, Residential Property Flood-Loss Assessment Methodologies," Front Environ Sci, vol. 9, Oct. 2021, doi: 10.3389/fenvs.2021.734294.

[67] L. Marçal Russo et al., "Do urban digital twins need agents?" Environ Plan B Urban Anal City Sci, 2025, doi: 10.1177/23998083251317666.

[68] G. Mylonas, A. Kalogeras, G. Kalogeras, C. Anagnostopoulos, C. Alexakos, and L. Munoz, "Digital Twins from Smart Manufacturing to Smart Cities: A Survey," 2021, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ACCESS.2021.3120843.

[69] M. Marzouk and B. Mohamed, "Integrated agent-based simulation and multi-criteria decision making approach for buildings evacuation evaluation," Saf Sci, vol. 112, pp. 57–65, Feb. 2019, doi: 10.1016/j.ssci.2018.10.010.

[70] Z. Wang, J. Huang, H. Wang, J. Kang, and W. Cao, "Analysis of flood evacuation process in vulnerable community with mutual aid mechanism: An agent-based simulation framework," Int J Environ Res Public Health, vol. 17, no. 2, Jan. 2020, doi: 10.3390/ijerph17020560.

A Novel Lightweight Explainable Multilayer Adaptive RNN-Based Intrusion Detection Framework

Nidhi Srivastav[1], Rajiv Singh[2]
Department of Computer Science, Banasthali Vidyapith, Rajasthan, India [1, 2]
Centre for Artificial Intelligence, Banasthali Vidyapith, Rajasthan, India[2]

***Abstract*—A rapid increase in the instances of cyberattacks has been observed with the expanding digitization. This leads to an urgent and critical need for developing robust intrusion detection systems (IDS) which can identify the occurrences of malicious activities within the network traffic. The present work proposes a novel, explainable, multilayer, lightweight adaptive IDS based on a Recurrent Neural Network (RNN). The purpose of this proposed IDS is to improve threat detection capabilities, especially low frequency high severe attack. The performance of the proposed IDS is evaluated using the UNR-IDD dataset. The network traffic is classified into normal and attack categories to assess the effectiveness of the proposed IDS. Two separate IDS models are developed. Model A is used to detect attacks on the basis of the frequency of the attacks, and Model B detects threats based on the severity of the attacks. Through the layered approach, the overall detection accuracy of 95.7% is achieved in Model A, and 97.5% is obtained in Model B. The present work highlights that the proposed IDS shows a remarkable improvement in the detection of less frequent severe attacks in comparison to existing IDS. The comparative result analysis of RNN-based IDS with Machine Learning models such as LR, Naïve Bayes, Cat Boost, Random Forest and Multilayer Perceptron models shows RNN-based IDS has outperformed the Machine Learning models. Explainable AI (XAI), the SHAP method is used for better interpretation of the proposed decisions. XAI helps to identify the network traffic that can influence predictions and detect potential biases. It also helps researchers and practitioners to validate model behaviour and establish trust in the system's outputs.**

***Keywords*—IDS; network security; adaptive techniques; RNN; cybersecurity; explainable AI; UNR-IDD**

I. Introduction

With a rapid increase in the number of networks and internet usage, there has been a substantial rise in cybersecurity issues. Fig. 1 depicts the organization global weekly cyber-attack data. It shows that during the first quarter of 2025, cyberattacks increased noticeably. Each business saw 1,925 attacks on average per week, which is 47% increase over the same time period in 2024. Government, telecommunications, and education have become the most targeted industries as cybercriminals continue to improve their techniques [1]. Ransomware attacks rose by 126%, with North America accounting for 62% of global incidents, and Consumer Goods & Services being the most targeted sector [1]. The complexity and variety of modern cyberattacks pose a considerable challenge to traditional IDS methods. Traditional methods, such as anomaly-based detection, which identifies deviations from typical behavior, and signature-based detection, which depends on predefined attack signatures, frequently have drawbacks like high false-positive rates, a slow rate of adaptation to new attack types, and an inability to identify zero-day attacks [2].

With the increasing complexity of cyberattacks, it becomes difficult to detect low-occurrence and zero-day attacks. Recent years have seen several rare but high-impact cyberattacks that demonstrate the growing sophistication of threat actors. XZ Utils backdoor did widespread Linux server compromise [3]. ICS "blackhole" is an attack that disrupted control traffic without malware [4]. Real-world incidents include the FrostyGoop [5]. Malicious PyPI packages attempt credential theft in the supply-chain attack [6]. Another example of such an attack is TetraSoft's industrial ecosystem attacking the major energy sector [7]. Kyivstar telecom attack [8], which disrupted nationwide communications and alerts. All the above examples show that these events occur rarely but have large-scale operational and national consequences.

Generally, low-occurrence attacks are rare, but their per-incident impact is often experienced as disproportionately high. As in [9], the 2021 Colonial Pipeline ransomware attack affected the entire fuel supply chain, causing billions in economic damage. The North American Electric Reliability Corporation (NERC) identified events of coordinated cyberattacks on critical infrastructure, leaving severe consequences, despite their rarity [10]. According to IBM's Cost of a Data Breach report, low-frequency, high-impact attacks, like advanced persistent threats (APTs), usually target sensitive infrastructure like power grids or financial systems and worsen the damages [11].

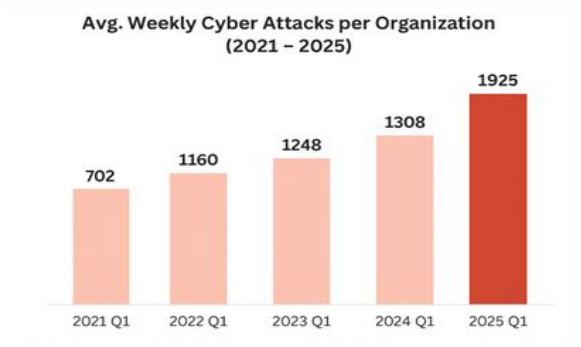


Fig. 1. Global weekly cyber attacks per organisation.

The rest of the study is structured as follows: Section II addresses relevant work about intrusion detection systems, with a particular emphasis on the use of deep learning and machine learning methods. Section III describes the methodology that introduces the proposed layered IDS framework. Section IV throws light on the architecture, steps of data preprocessing and

RNN-based IDS. Section V discusses experimental requirements, configuration of models, datasets, and metrics used for performance assessment and then analyses the results. Section VI summarizes the findings of this research.

II. Literature Review

Since the last decade, there has been significant research on Intrusion Detection Systems (IDS) based on Machine Learning (ML) algorithms. Researchers extensively used ML algorithms such as Decision Trees, Support Vector Machines (SVM), Random Forest, K-Nearest Neighbor (KNN) and Naïve Bayes for developing IDSs for decreasing false alarms [12], [13], [14]. Presently, deep learning techniques like Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Units (GRU) are used particularly for increasing anomaly detection rates due to their ability to identify sequential patterns [15] [16], [17]. With the increasing complexity, the traditional ML IDS often fail to detect complex attack patterns. Therefore, deep learning models are preferred over traditional ML models, due to Deep Learning classifiers and ensemble classifiers' ability to process an enormous amount of data and capture temporal dependencies [18], [19], [20]. Therefore, deep learning classifiers and ensemble classifiers are used to detect intrusions. However, several challenges remain, such as overfitting, high computational costs, and dataset dependency. Each examined study attempts to enhance the effectiveness and application of IDS by uniquely addressing these issues. Despite advancements in ML classifiers, IDS still come across challenges such as overfitting, high computational costs, and dataset dependency. Researchers have addressed these issues in their work to increase the effectiveness of IDS [21].

Prior literature review over the past few years, Gautam [22] introduced a composite approach combining LSTM and GRU architectures for improving IDS performance. The combination of LSTM and GRU enables the model to retain long-term dependencies and handle short-term events effectively. This hybrid approach helped to extract complex patterns in large-scale traffic data. The model was trained on CICIDS-2017, which comprised current attack types like DDoS, port scans and web attacks. In this model, a very high accuracy of 99.3% was reported. Ibrahim, 2023[23] modelled IDS with Long Short-Term Memory and Simple RNN. Ibrahim's study devised an IDS using a combination of simple RNN and LSTM, aiming to compare the performance of basic RNN with its enhanced LSTM counterpart. The primary emphasis was on evaluating the trade-offs between Simple RNN computational efficiency and LSTM accuracy. Ibrahim [23] used the KDD Cup dataset and achieved a 90.6% accuracy, which was lower than other deep learning approaches. The limitation of the study was to use an older dataset and a simpler RNN architecture. The focus was on evaluating the trade-offs between computational efficiency (Simple RNN) and accuracy (LSTM). Ibrahim uses KDD Cup dataset, and older but still used datasets for IDS research were used. The IDS model achieved a 90.6% accuracy, which is lower than other deep learning approaches. The limitation in accuracy was attributed to the older dataset and simpler RNN architecture. Fog-Cloud-Based IDS Using Bi-Directional LSTM and Simple RNN by Syed, 2023[24].

This study proposes fog-cloud computing-based IDS, comprising Simple RNN and Bi-Directional LSTM. The fog-cloud architecture is used to decentralize IDS by reducing the latency and improving real-time detection. Bi-Directional LSTM is found capable of processing data in both directions. It is used for capturing complex traffic patterns and was tested on the BoT-IoT dataset. A hybrid NID-Recurrent Neural Network (RNN) with Long Short-Term Memory (LSTM) was developed by Amutha et al., 2022 [25] to enhance detection capabilities. The study used the UNSW-NB18 dataset and found 8% increase in accuracy in comparison to the simple RNN-based models. Recently, Azarudeen, 2024 [26], developed an innovative model integrating LSTM-RNN with a hybrid sparse autoencoder and Deep Neural Network (DNN). The performance of IDS was found to be better than earlier methods in terms of accuracy, detection rate, and low false alarm rates. Adamax optimizer was used on the NSL-KDD dataset for multi-attack categorization. Network IDS developed through Attention Mechanisms and RNNs by [27] Djaidja, 2024. IDS developed by combining RNNs, LSTM and Gated Recurrent Units (GRU) for detecting network intrusions at an earlier. stage. During testing on the CICIDS-2017 and 5G-NIDD datasets, the model performed early-stage detection before the flow termination. Parveen, 2024 [28] presented a real-time intrusion detection system (IDS) on combining Convolutional Neural Networks (CNNs) and RNNs. Using the UNR-IDD dataset, the hybrid CNN-RNN model achieved a 96.2% accuracy with a low false alarm rate. Samriya, 2024 [29] presented a novel NIDS which comprises Support Vector Machines (SVM) and Extreme Gradient Boosting (XGBoost). The developed NIDS performance was improved through hyperparameter optimization using the Crow Search Algorithm. Rathee [30] examined a number of deep learning models, such as CNNs, shallow neural networks, and deep neural networks (DNN), with attention mechanisms. NIDS developed through Deep Learning techniques tested on NSL-KDD, Kyoto, and UNSW-NB15 datasets. NIDS achieved robust detection results across varying architectures. Mighan, 2021 [31] developed IDS from a hybrid approach consisting of stacked autoencoders for feature extraction, followed by traditional classifiers like SVM, random forests, and decision trees. The IDS was tested on the UNB ISCX 2012 dataset. The system showed scalability and high performance in real-time scenarios.

The study explored various feature selection approaches to enhance the effectiveness of ML-based IDS. [32] focused on optimizing deep learning models for attack classification through hyperparameter optimization methods such as grid search and random search. The accuracy of the model was 83.33% on the NSL-KDD dataset and 95.79% on the CSECIC-ID2018 dataset. Kasongo [33] proposed a layered framework for IDS, where each layer contributed to the detection of specific types of network intrusions. The novelty of this approach lies in the hierarchical structure of deep learning layers that help improve feature extraction. The framework combined multiple deep learning techniques (e.g., CNN, LSTM) in a pipeline for multi-stage detection. The datasets used are UNSW-NB15, a dataset containing contemporary attack scenarios. [34] presented a novel NIDS using a stacked non-symmetric deep autoencoder combined with an SVM classifier, the system achieved 99.65% by using KDD Cup'99 dataset. [35] proposed

a stack ensemble of tree-based boosting classifiers-XGBoost, LightGBM, and Cat-Boost-for classification.[36] presents a privacy-preserving hybrid machine learning model, AOPRF-XGBoost, designed to enhance intrusion detection in IoT systems. Prominent existing work and their findings are summarized in Table I.

TABLE I. EXISTING IDS TECHNIQUES

Author	Year	Methodology	Findings
Gautam et al. [22]	2022	Used LSTM and GRU with RNN.	Worked on CICIDS-2017, achieved 99.3% accuracy.
Ibrahim and Elhafiz [23]	2023	Use Long Short-Term Memory (LSTM) with Recurrent Neural Network (RNN) architecture.	Worked on KDD Dataset and achieved only 90.6% accuracy on testing dataset.
Amutha et al. [25]	2022	integrating NID-RNN with Long short-term memory (LSTM)	Used UNSW-NB18 data set. Accuracy increases by 8% as compared to simple RNN.
Parveen et al. [28]	2024	This model used CNNs,RNNs, and synthetic Neural Networks (ANNs),	Worked on UNR-IDD dataset achieve an accuracy of 96.2% with very less false alarm rate.
Samriya et al. [29]	2024	Use Support Vector Machine (SVM) and eXtreme Gradient Boosting (XGBoost) techniques.	Worked on NSL-KDD and UNR-IDD dataset.
Qazi et al. [32]	2022	utilized a stacked Non-symmetric deep auto encoder & support vector machine classifier	Accuracy of 99.65% is achieved at terms of increasing complexity
Kasongo et al. [33]	2022	Developed a framework where each layer contributes to data preprocessing, sequential pattern analysis, and a hierarchical model structure.	Used dataset UNSW-NB15, that improved both the detection accuracy and the interpretability of results.
P. R Buvaneswari, et al. [35]	2024	Uses stacked ensemble model	Accuracy is 98% but complexity of model increases

Although extensive research has been carried out in the field of intrusion detection using Artificial Intelligence and Machine Learning techniques, the literature indicates that only a limited number of studies have specifically focused on the detection of low-frequency yet high-impact (severe) attacks. Moreover, many of the existing systems that achieve high overall detection accuracy are often complex in nature, leading to increased computational cost and reduced real-time applicability. The literature reviewed shares several common challenges, including:

- Dataset limitations: The reliance on older or static datasets like NSL-KDD and KDD CUP'99 raises concerns about the ability of these models to generalize to modern network environments.
- Model complexity: Deep learning models are generally accurate, but often require significant computational resources, as in a stacked ensemble model, which is complex in terms of system design and management, and therefore, they are challenging to deploy in real-time, resource-constrained situations like cloud systems or the Internet of Things.
- Attack diversity: Many models are trained on specific datasets with a limited variety of attack types, which affects their ability to detect novel, zero-day, or obscure attacks.
- Detection of less frequent attacks: Although many systems achieve high overall accuracy, often reaching 96% to 97% but, they struggle to detect less frequent attack types despite achieving a good accuracy. This limitation arises because these rare attacks are underrepresented in the training data, leading to insufficient learning and poor detection performance for such cases.
- Detection of highly severe attacks: Many systems achieve good accuracy in detecting attacks, but the time required to detect all attacks is so high that sometimes it reduces the practical usefulness of attack detection.

The objectives of this research are as follows.

- To develop an IDS that uses a simple Recurrent Neural Network (RNN) within a layered architecture to enhance detection accuracy in less time and minimize false positives.
- To introduce a novel multi-layered IDS framework that combines data preprocessing, feature extraction, and RNN-based sequence modeling corresponding to specific attack type which helps to effectively detect low-frequency, zero–day–like behavior patterns by learning generalized representations of anomalous traffic rather than relying solely on predefined attack signatures.
- To highlight the advantages of the layered design, such as improved modularity, enhanced detection capabilities, and reduced computational overhead and detection time.
- To integrate Explainable Artificial Intelligence (XAI) techniques with black-box machine learning models to enhance model transparency by not only generating predictions but also providing interpretable explanations.

The proposed model focuses on creating more adaptable models that can handle large-scale and complex data using some simple, less complex ML/DL models, which are able to detect severe low-occurrence attacks with less computational overhead and are fast in terms of inference time, in spite of having few samples in the dataset.

III. PROPOSED METHODOLOGY

The suggested approach uses machine learning and deep learning classifiers in a multi-layer classification architecture to identify and stop network attacks. The approach is designed to optimize detection accuracy by categorizing threats based on their frequency of occurrence. The architecture is structured into two distinct layers, each tailored to a specific category of attacks: Layer 1 focuses on the detection of frequent

attacks/severe attacks, whereas Layer 2 is dedicated to identifying less frequent/ less severe attack types. This approach not only allows for a more systematic detection process but also helps in addressing challenges like high false-positive rates, real-time detection, and resource management.

The proposed model employs the UNR-IDD dataset for the detection of attacks. Authors in [37] suggested that UNR-IDD captures modern network behaviour and includes realistic network traffic, which is more representative of contemporary network environments. It is clearly seen from Table II that various attack categories (such as TCP-SYN, PortScan, Diversion and Blackhole) represented with enough samples, but instances of Overflow were very few in UNR-IDD, contributing only 2% of total instances and thus considered as a less frequent attack. It is very difficult to detect such attacks, as the models were not perfectly trained.

Data gathered from various sources [38], [39], [40], [41] has been summarized in Table III, which shows that Overflow and Blackhole attacks are among the most severe, although these attacks occur less frequently. As a result, there are fewer samples of these attacks in datasets, which makes it challenging to effectively train models to detect them. The limited number of samples hampers the model's ability to learn and generalize types of attacks. One potential solution is to synthetically generate samples to increase their representation in the dataset. However, this approach often leads to a decrease in the overall accuracy of the model.

To address this challenge, the proposed methodology employs a simple, low-complexity layered adaptive framework specifically designed to detect less frequent, severe attacks, as shown in Fig. 4. This method enhances the detection accuracy for Overflow attacks without compromising the overall performance of the model.

TABLE II. ATTACK TYPES AND POPULATION SIZE IN UNR-IDD

Attack Types	Population Size	% of attack
Blackhole	8420	22.5%
Diversion	5615	15.00%
Normal	3773	10.08%
Overflow	1022	2.7%
PortScan	9500	25.39%
TCPSYN	9081	24.27%

TABLE III. ATTACK SEVERITY AND FREQUENCY OF ATTACKS AVAILABLE IN UNR-IDD DATASET [38], [39], [40], [41]

Attack Types	Description	Severity	Frequency	Impact
TCP-SYN Flood	DoS attack that exploits the TCP handshake to flood a server	Moderate to High	high	Causes server overload, resource exhaustion, and denial of service
PortScan	Reconnaissance attack identifies open ports and services on a target system.	Moderate	Very High	provides information for further exploitation
Overflow	sends more data than a buffer can manage	Critical	low	System crash, remote code execution system compromise.
Blackhole	drop packets or reroute traffic, disrupting data flow.	Critical	moderate	Complete data loss disruption of comm. in networks.
Diversion	Traffic is misdirected leading to data theft or service disruption.	Moderate to High	high	Sensitive data exposure, potential service interruption,

IV. ARCHITECTURE OF ADAPTIVE PROPOSED FRAMEWORK

The proposed system is trained in a hierarchical manner similar to its detection workflow. In the offline training phase, as shown in Fig. 2, the dataset is first preprocessed and then divided into frequent/ severe and less-frequent/less-severe attack subsets. Separate feature extractors and classifiers are trained for each category.

During the detection process, incoming traffic is first evaluated using the frequent-attack classifier. After a frequent attack is detected, the traffic is further analyzed using the less-frequent attack classifier, as shown in the pseudocode of Fig. 3 and the architecture of the proposed layered framework in Fig. 4. This hierarchical training and detection strategy improves detection accuracy for rare and severe attacks while maintaining efficiency for common attack patterns.

The proposed adaptive framework works for two models:

Model A: Responsible for frequency-based attack detection.

Model B: Responsible for severity-based attack detection.

Both Models are based on a layered/hierarchical approach, where Layer 1 is responsible for detecting frequent/severe attacks, and Layer 2 is responsible for detecting less frequent / less severe attacks. Layer is designed specifically to detect less frequent and severe attacks. Despite the limited number of samples, it operates efficiently by focusing solely on these less common but critical threats. Records which remain unidentified after Layer 2 are considered as Normal.

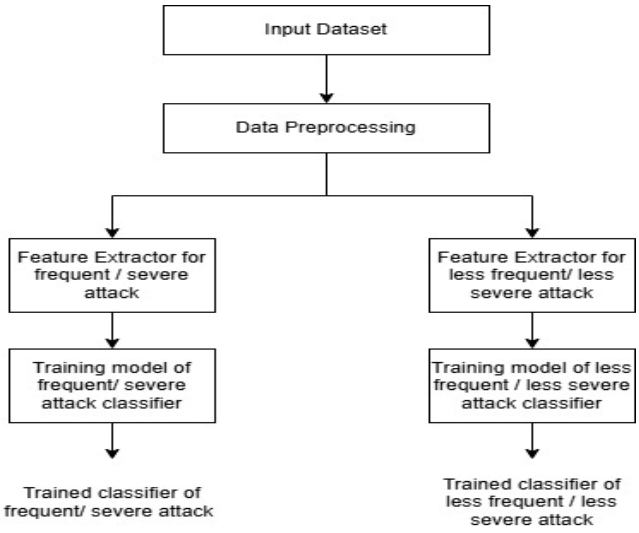


Fig. 2. Offline training process.

```
Algorithm Detect_Attack
    Input: dataset ← load_dataset("all_features")
    Output: attack_label ∈ {"Frequent Attack", "Less Frequent Attack", "Normal Traffic"}
    Begin
    // ---------------------------- Layer 1 ----------------------------
    // Step 1: Preprocess dataset for frequent attacks
    preprocessed_frequent ← data_preprocessor(dataset)
    // Step 2: Extract features relevant to frequent attacks
    features_frequent ← extract_features(preprocessed_frequent, type = "frequent")
    // Step 3: Classify using frequent attack model
    result_frequent ← classify (features_frequent, model = "frequent_attack_classifier")
    IF result_frequent == "Frequent Attack" THEN
    block traffic ()
    RETURN "Frequent Attack"
    ELSE
    Proceed to Layer 2
    END IF
    // ---------------------------- Layer 2 ----------------------------
    // Step 4: Preprocess dataset for less frequent attacks
    preprocessed_less_frequent ← data_preprocessor(dataset)
    // Step 5: Extract features for detecting rare or stealthy attacks
    features_less_frequent ← extract_features(preprocessed_less_frequent, type = "less_frequent")
    // Step 6: Classify using less frequent attack model
    result_less_frequent ← classify(features_less_frequent, model = "less_frequent_attack_classifier")
    IF result_less_frequent == "Less Frequent Attack" THEN
    block_traffic()
    RETURN "Less Frequent Attack"
    ELSE
    // ---------------------- Normal Traffic ----------------------
    // If not classified as either frequent or rare attack
    mark_as_normal()
    RETURN "Normal Traffic"
    END IF
    End
```

Fig. 3. Proposed methodology for detection of attacks.

The proposed framework consists of the following components, as shown in Fig. 4:

1) Data preprocessor: It is used to preprocess the data and can be divided into two phases:

a) Data cleaning: This involves handling missing, inconsistent, or inaccurate data entries. Techniques such as filling missing values (imputation), removing duplicates, and correcting errors are commonly used. In the proposed model, missing values were replaced with mean or median values, or specific rows or columns can be removed if they are unreliable.

b) Data transformation: Data representation and normalization processes are used to preprocess data on the adopted dataset. Data is represented using the one-hot encoding method. Then, the data normalization process can be done using Min-Max transformation, which transforms data into a format that suggested models can use more effectively. It normalizes data to a range (e.g., 0-1) and rescales data to have a mean of 0 and standard deviation of 1, which is frequently necessary for algorithms that are sensitive to feature magnitudes, such as neural networks and k-nearest neighbors.

2) Feature extractor: The purpose of the feature extractor component is to extract features from the dataset that are unique to the attack of a defined layer. Its main aim is to reduce the dataset for effective and smooth functioning in the era of high volume of traffic on the internet. Random Forest is a powerful technique for feature selection in machine learning. Random Forests inherently rank features based on their importance, allowing for the selection of the most relevant features for a given predictive task [42]. Overfitting can be minimized because the random forest construction process is random, and each decision tree is unrelated to the others. Random forest classifier assigns an importance score to each feature, based on which important features are selected as shown in Fig. 5(a) for Model A, high frequent attack and Fig. 5(b) for Model B, for high severe attack.

To enhance interpretability, SHapley Additive exPlanations (SHAP) were employed to quantify each feature's influence on the classifier's output. The SHAP framework distributes the model prediction among all input features according to their marginal contribution, providing a consistent and additive measure of importance. The SHAP bar graph, as shown in Fig. 6(a) and Fig. 6(b), shows each feature's average impact on the prediction. Longer bars indicate stronger influence, and the color indicates whether higher or lower feature values push the model toward normal or attack classes.

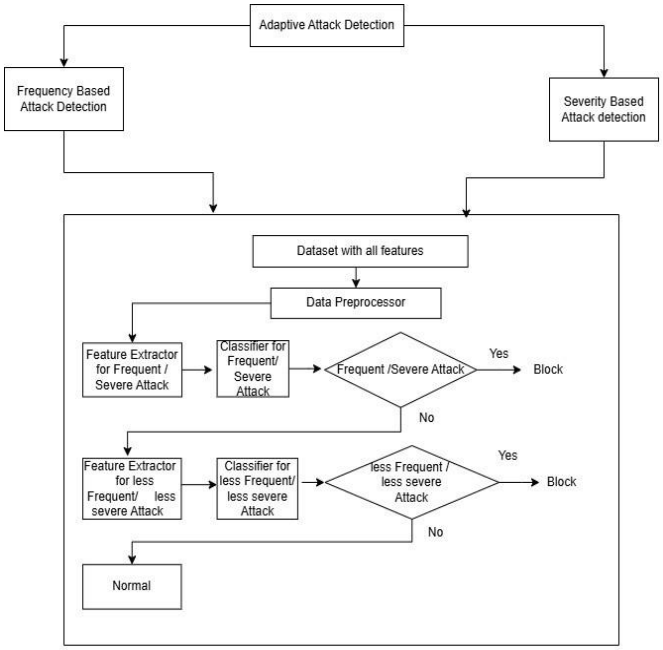


Fig. 4. Architecture of proposed layered framework for attack detection.

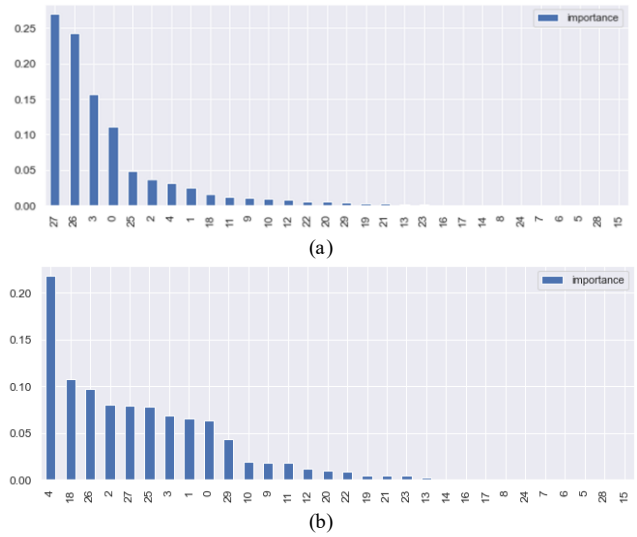


Fig. 5. (a). Features are sorted according to their importance score for high-frequency attack (Model A). (b). Features are sorted according to their importance score for high severe attack (Model B).

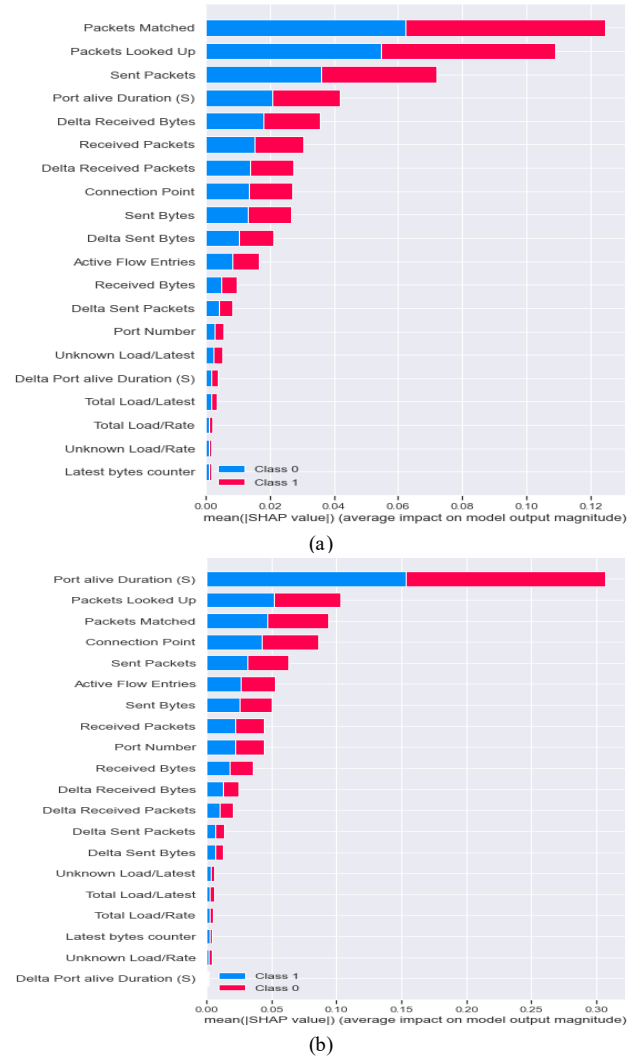


Fig. 6. (a). SHAP Graph corresponding to high Frequent Attack (Model A), (b). SHAP Graph corresponding to high Severe Attack (Model B).

According to the importance score and SHAP explanation following features are selected for Layer1 (Model A): Packets Matched, Packets Looked Up, Sent Packets, Received Packets, Active Flow Entries, Sent Bytes, Port Alive Duration (S) and Received Bytes. The features Active Flow Entries and Received Bytes were retained in the final model due to their strong summarizing capability and significant contribution to computational efficiency. Both attributes act as high-level aggregates that encode essential aspects of network behavior. Active Flow Entries and Received Bytes serve as compact, low-redundancy indicators that preserve predictive performance while substantially reducing model size, training epochs, and real-time inference latency.

Layer1(Model B): Port alive Duration (S), Sent Bytes, Connection Point, Packets Looked Up, Active Flow Entries, Packets Matched, Sent Packets, Received Bytes.

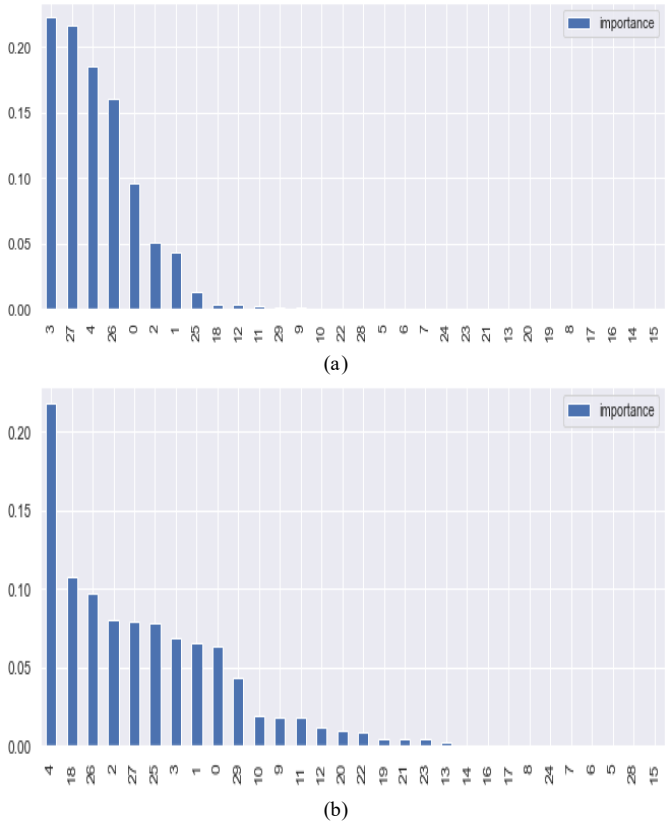


Fig. 7. (a). Features are sorted according to their importance score for low-frequency attack (Model A), (b). Features are sorted according to their importance score for a less severe attack (Model B)

According to importance score and SHAP graph as shown in Fig. 7(a), 7(b) and Fig. 8(a), 8(b), following features are selected for Layer 2 (Model A): Sent Packets, Packets Matched, Port alive Duration (S), Packets Looked Up, Received Packets, Sent Bytes, Received Bytes, Active Flow Entries.

Layer2 (Model B): Packets Matched, Packets Looked Up, Sent Packets, Port alive Duration (S), Received Packets, Sent Bytes, Received Bytes, Active Flow Entries.

A feature extractor is used at each layer so that important features corresponding to a specific attack can be extracted.

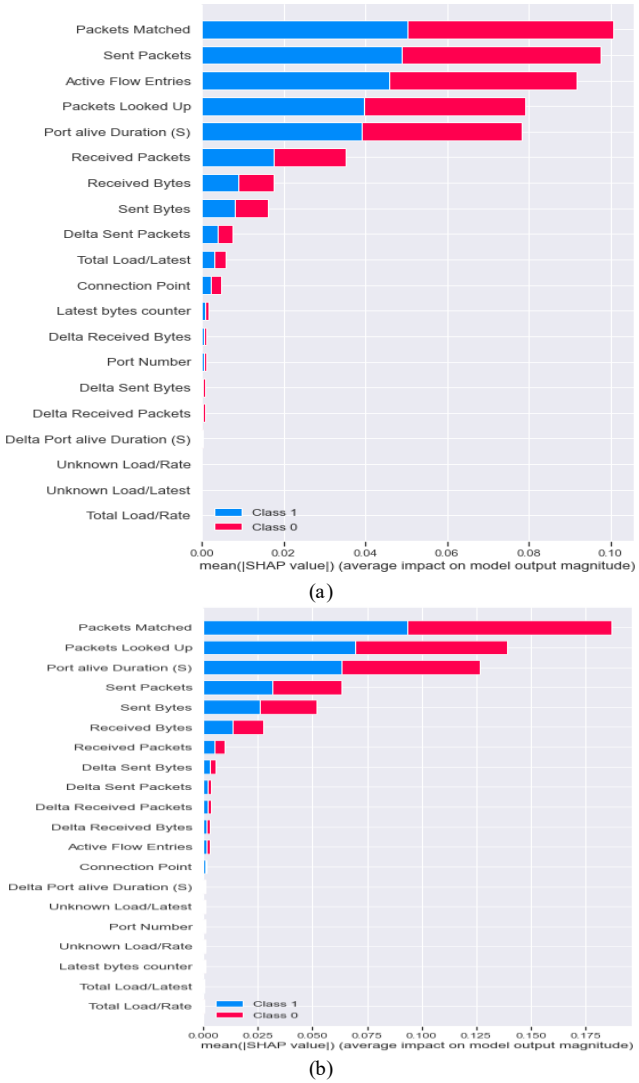


Fig. 8. (a). SHAP graph corresponding to less frequent attack (Model A), (b). SHAP graph corresponding to less severe attack (Model B)

3) Classifier: This component is used to classify records available in the dataset to detect attacks by analysing the training records in the dataset used. Here, the records are categorized as normal or assault using a Simple RNN. For applications like network traffic intrusion detection, Recurrent Neural Networks (RNNs), a kind of deep learning model built to handle sequential input, are ideal. RNNs are perfect for capturing the temporal relationships found in network traffic because, in contrast to standard feedforward neural networks, they have "memory" that enables them to remember information about prior inputs. Natural language processing and time series analysis have made extensive use of RNNs, and their capacity to represent sequential dependencies makes them a desirable option for intrusion detection systems. RNNs, especially simple RNNs, can use fewer parameters than some other networks, like CNNs with multiple layers, which often require more parameters for effective feature extraction. This can result in a lighter, more efficient model. Simple RNNs typically have a lower computational cost and memory footprint than deeper or more complex networks like CNNs or Transformers, which require more layers and heavier processing power, particularly on large datasets. Due to the simpler architecture, RNNs train faster on smaller datasets, where complex networks like CNNs could risk overfitting or require extensive regularization techniques.

V. Experimental Setup

Several tests are carried out in this section to evaluate the performance of the suggested model against the current model. A personal laptop with an Intel Core i7 processor and 16 GB of RAM serves as the development tool for the network intrusion detection technique suggested in this research. Python 3.9 is used to implement each classifier. The dataset used in this experimental setup is UNR-IDD. It is a prominent benchmark dataset containing 32 features. The proposed framework does an 80-20% split, which means 80% data is used for training and 20% is used for testing purposes. The suggested model employs Sigmoid activation (for binary classification) in the final layer, 64 units in the subsequent layer, and 128 units with ReLU activation in the first layer. Training uses a batch size of 32 and 20 epochs. The performance of the suggested model is compared using four evaluation metrics—accuracy, precision, recall, and F1 score—based on the confusion matrix.

Accuracy measures the proportion of correctly classified instances out of the total instances. This measure works well for datasets that are balanced, but not for unbalanced datasets

Precision is the ratio of accurately predicted positive cases to all anticipated positive instances.

Recall is the ratio of accurately anticipated positive instances to all actual positives and is known as recall (sensitivity). Although the model may contain some false positives, high recall indicates that it successfully detects positive events.

The F1 Score is calculated as the precision and recall harmonic means. Because it incorporates both accuracy and recall values, it is primarily utilized with datasets that are unbalanced.

Tables IV and V present the performance matrix of the proposed model in detecting attacks based on key metrics: Precision, Recall and F1 Score.

TABLE IV. Performance Matrix of Proposed Model A

Attack	Proposed Model A		
	Precision	***Recall***	***F1 Score***
Frequent Attack	0.976	1	0.988
Less Frequent Attack	1	0.235	0.38

TABLE V. Performance Matrix of Proposed Model B

Attack	Proposed Model A		
	Precision	***Recall***	***F1 Score***
Severe Attack	0.766	0.892	0.824
Less Severe Attack	0.965	0.916	0.940

Proposed Model A shows very high precision, recall and F1 score for the most frequent attack, that indicate good detection

performance in detecting frequent attacks. The proposed Model B achieves high recall for severe attacks, which ensures that most critical threats are detected in an efficient way, and the model also shows very high precision and F1-score for less severe traffic, which represents good overall classification performance with fewer false alarms.

Fig. 9 & Fig. 10 represents summarized rule sheet for identifying frequent and severe attacks, respectively. This rule sheet functions as a verification mechanism for evaluating the model's output. By systematically comparing model predictions with predefined rules, it becomes possible to identify inconsistencies, incorrect classification patterns, and potential sources of error such as misclassification, missing feature contributions, or signs of overfitting. This comparison process further helps in understanding how the model distinguishes between attack and benign classes, thereby offering clearer insight into the decision logic behind each prediction.

In [43] focuses on evaluating various machine learning models for intrusion detection using the UNR-IDD Intrusion Detection Dataset. Table VI denotes the Performance matrix of existing classifiers [43].

Table VII denotes the Performance Matrix of Existing Classifiers for detecting severe and less severe attacks.

Following Table VIII presents a comparative analysis of existing classifiers [43] against the proposed classifier based on F1-Score performance for both frequent and less frequent attacks.

```
Deeper Rule Tree (True Labels):

|--- Packets Matched <= 1.63
|   |--- Packets Looked Up <= -0.40
|   |   |--- Sent Bytes <= -0.71
|   |   |   |--- Received Packets <= -0.33
|   |   |   |   |--- Port alive Duration (S) <= -0.71
|   |   |   |   |   |--- Sent Bytes <= -0.71
|   |   |   |   |   |   |--- Packets Looked Up <= -0.43
|   |   |   |   |   |   |   |--- class: 0.0
|   |   |   |   |   |   |--- Packets Looked Up >  -0.43
|   |   |   |   |   |   |   |--- Packets Matched <= -0.42
|   |   |   |   |   |   |   |   |--- class: 0.0
|   |   |   |   |   |   |   |--- Packets Matched >  -0.42
|   |   |   |   |   |   |   |   |--- class: 0.0
|   |   |   |   |   |--- Sent Bytes >  -0.71
|   |   |   |   |   |   |--- class: 1.0
```

Fig. 9. Summarized Rule sheet to identify frequent attacks and normal traffic.

```
Deeper Rule Tree (True Labels):

|--- Port alive Duration (S) <= -0.63
|   |--- Packets Matched <= -0.41
|   |   |--- Sent Packets <= -0.38
|   |   |   |--- Packets Matched <= -0.42
|   |   |   |   |--- Packets Looked Up <= -0.43
|   |   |   |   |   |--- class: 1.0
|   |   |   |   |--- Packets Looked Up >  -0.43
|   |   |   |   |   |--- Packets Looked Up <= -0.43
|   |   |   |   |   |   |--- class: 0.0
|   |   |   |   |   |--- Packets Looked Up >  -0.43
|   |   |   |   |   |   |--- Received Packets <= -0.33
|   |   |   |   |   |   |   |--- Port alive Duration (S) <= -0.80
|   |   |   |   |   |   |   |   |--- class: 1.0
```

Fig. 10. Summarized Rule sheet to identify severe attack and normal traffic.

TABLE VI. PERFORMANCE MATRIX OF EXISTING CLASSIFIER[43]

	Logistic Regression			Naive Bayes			Multilayer Perceptron		
	Precision	***Recall***	***F1 score***	***Precision***	***Recall***	***F1 score***	***Precision***	***Recall***	***F1 score***
Frequent	0.43	0.45	0.39	0.66	0.59	0.59	0.53	0.52	0.46
Less Frequent	0	0	0	0.29	0.3	0.3	0.66	0.22	0.33

TABLE VII. PERFORMANCE MATRIX OF EXISTING CLASSIFIER FOR DETECTING SEVERE AND LESS SEVERE ATTACK

	Logistic Regression			Naive Bayes			Multilayer Perceptron		
	Precision	***Recall***	***F1 score***	***Precision***	***Recall***	***F1 score***	***Precision***	***Recall***	***F1 score***
Severe	0.31	0.19	0.23	0.54	0.44	0.48	0.41	0.07	0.24
Less Severe	0.44	0.51	0.47	0.62	0.66	0.64	0.52	0.58	0.69

TABLE VIII. COMPARATIVE ANALYSIS OF EXISTING CLASSIFIER & PROPOSED MODEL A

	Frequent Attacks			Less Frequent Attacks		
	F1 score	***F1 Score (Model A)***	***% increase***	***F1 score***	***F1 Score (Model A)***	***% increase***
Logistic Regression	0.397	0.988	148.6	0.1	0.380	38.03
Naive Bayes	0.592		66.78	0.3		26.76
MultiLayer Perceptron	0.467		111.38	0.33		15.24

TABLE IX. COMPARATIVE ANALYSIS OF EXISTING CLASSIFIER & PROPOSED MODEL B

	Severe Attacks			Less Severe Attacks		
	F1 score	***F1 Score (Model B)***	***% increase***	***F1 score***	***F1 Score (Model B)***	***% increase***
Logistic Regression	0.23	0.824	258.26	0.47	0.940	100
Naive Bayes	0.48		71.66	0.64		46.875
MultiLayer Perceptron	0.24		243.33	0.69		36.231

As it is evident from Table VIII and Table IX that the F1 Score for frequent, severe, less frequent and less severe are increased manifoldly, and it is graphically shown in Fig. 11.

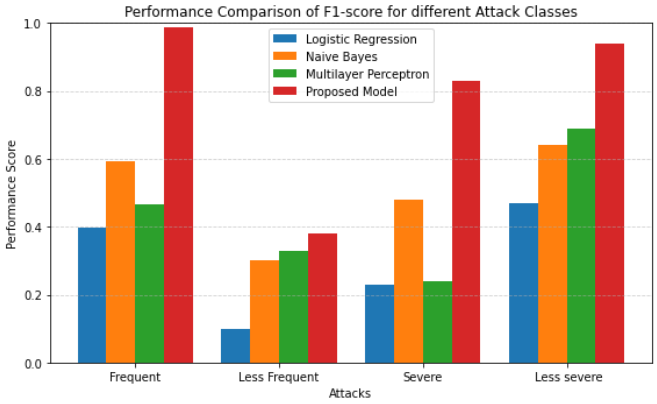


Fig. 11. Comparison of F1 score for different attack types.

In any Intrusion Detection System (IDS), the percentage of attacks successfully blocked is a critical performance metric. In the proposed model A, attacks are categorized into two types: frequent and less frequent. The performance of each detection layer is evaluated separately and compared against the existing model. As presented in Table X proposed Model A demonstrates significantly improved performance, blocking 99.84% of frequent attacks 85.7% of less frequent attacks.

TABLE X. COMPARISON OF EXISTING CLASSIFIER WITH PROPOSED MODEL A ON THE BASIS OF BLOCKING OF ATTACK

Classifiers	% of Frequent Attack Blocked	% of less Frequent Attack Blocked
KNN	82%	46%
Naïve Bayes	59.50%	30%
Decision Tree	93.25%	84%
Random Forest	94.25%	77%
CatBoost	96.25%	85%
XGBoost	97%	93%
Multi Layer Perceptron	52.75%	22%
Proposed Model	**99.84%**	**85.91%**

Similarly, Table XI represents proposed Model B blocks 99.8% of severe attacks and 96.82% of less severe attacks. This clearly highlights the effectiveness and robustness of the proposed model over existing approaches.

Even when considering the Accuracy metric, the proposed model outperforms existing machine learning, deep learning, and ensemble classifiers. Table XII indicate comparative analysis of Ensemble ML classifiers like CatBoost, RF and ET [43] and deep learning classifiers like CNN, RNN and ANN [44] with the proposed classifier, and it is clearly evident that the accuracy of the proposed classifier is higher than classifier in [43] [44].

Table XII indicates that Tree-based ensemble models (RF, ET) perform strongly, indicating the dataset benefits from decision boundaries learned via tree structures. Neural models like RNN also perform well due to sequential data handling. The proposed models, slightly better than the rest, likely incorporate advanced techniques or model fusion to maximize accuracy. It may be possible that the overall Accuracy of ML/DL models may be high, but the system is not able to handle less frequent attacks, which are severe too. To effectively detect less frequent and severe attacks and enhance overall accuracy, a Layered Framework is employed. Layered Framework help to achieve a high level of accuracy even with a simple RNN kind of model, as shown in Table XII. Ensemble classifier XGBoost performs better than Proposed Model A, but this performance comes at the cost of adding complexity in terms of memory used to store the model and the latency of predictions, as shown in Table XIII.

TABLE XI. COMPARISON OF EXISTING CLASSIFIER WITH PROPOSED MODEL B ON THE BASIS OF BLOCKING OF ATTACK

Classifiers	% of Severe Attack Blocked	% of less Severe Attack Blocked
KNN	80.35%	81.05%
Naïve Bayes	43.50%	66.2%
Decision Tree	96.7%	90.8%
Random Forest	96%	92.4%
CatBoost	97.6%	94.6%
XGBoost	98.4%	95.4%
Multi Layer Perceptron	66%	68.2%
Proposed Model	**99.8%**	**96.82%**

TABLE XII. COMPARATIVE ANALYSIS OF ENSEMBLE ML AND DL MODELS WITH THE PROPOSED MODEL

	Model	Accuracy
Ensemble ML Classifier	CatBoost [43]	91.18
	Random Forest [43]	95.12
	Extra Tree [43]	95.23
	XGBoost[43]	96.30
Deep Learning Models	CNN [44]	92.8
	ANN [44]	91.7
	RNN [44]	94.1
Proposed Model	**RNN with layers for frequency based attack detection**	**95.7**
	RNN with layer for severity based attack detection	**97.65**

TABLE XIII. COMPARATIVE ANALYSIS OF THE COMPLEXITY OF EXISTING MODELS WITH THE PROPOSED MODEL

Model Classifier	Model Size	Latency ms per prediction
XGBoost	1.54 MB	0.828
CatBoost	2.99 MB	0.440
RNN (Proposed Model A)	0.064 MB	0.8712
RNN (Proposed Model B)	0.0602 MB	0.535

Table XIII shows that the RNN achieves good performance on the UNR IDD dataset in terms of model size and latency. This advantage primarily stems from its lightweight architecture.

Overall, while CatBoost demonstrates superior speed in terms of latency, it comes at the cost of a much larger model size. The proposed RNN offers an optimal trade-off between ultra-compact model size and acceptable real-time latency, making it highly suitable for deployment in real-time, resource-constrained intrusion detection systems.

As highlighted in the literature review, numerous existing classifiers demonstrate high accuracy; however, they often depend on complex architectures, such as deep convolutional neural networks or hybrid models. On the other hand, lightweight classifiers usually have lower accuracy and perform poorly in detecting less frequent but critical attacks. The proposed classifier, conversely, achieves distinguished performance as shown by the results obtained by maintaining a simple architecture and employing a minimum set of features. While many hybrid models achieve similar or even higher accuracy, their complexity restricts real-time applications, especially in IoT and fog cloud environments. In the proposed model, employing even just eight features, a simple RNN provides efficient results even for low-frequency attacks. Thus, a layered framework facilitates a simple RNN to perform efficiently with fewer features, providing high efficiency and promising results. The contribution of this work lies in the unified integration of layered lightweight detection, adaptive feature selection, and explainable analysis specifically designed for low-frequency high-severity attack scenarios.

VI. Conclusion

The proposed model signifies a significant advancement in Intrusion Detection Systems (IDS) by implementing a layered architectural framework along with an RNN to improve the identification of low-frequency and rare cyberattacks. A major challenge in intrusion detection systems (IDS) is developing models that can be generalized across varied datasets. As model performance mainly depends on the nature of the input data, it becomes essential to evaluate the proposed model on multiple benchmark datasets. The proposed model achieves an overall accuracy of 95.7% (Model A) and 97.5% (Model B) on the UNR-IDD dataset. However, there is still scope for further improvement in the performance of the proposed IDS. Future work should be focused on minimizing cost, losses, and errors to further improve accuracy, precision, recall, and F1-score through trade-offs between model complexity, effectiveness, and computational efficiency. In addition to this, future research work should have an emphasis on improving generalization capabilities and developing efficient real-time applications for detecting zero-day and novel attacks. Future work will also focus on extending the proposed intrusion detection framework toward real-time deployment and stress-testing scenarios to evaluate its practical applicability in dynamic network environments.

References

[1] Check Point Software Technologies Ltd. “Security Blog.” Check Point, 4 May 2025, https://blog.checkpoint.com/security.

[2] Talukder, M. A., et al. “Machine Learning-Based Network Intrusion Detection for Big and Imbalanced Data Using Oversampling, Stacking Feature Embedding and Feature Extraction.” *Journal of Big Data*, vol. 11, no. 1, 2024, p. 33.

[3] Invicti. (2024). XZ Utils backdoor: Invicti Web Security Blog. https://www.invicti.com/blog/web-security/xz-utils-backdoor-supply-chain-rce-that-got-caught.

[4] G. Mitra, P. Dash, Y. E. Yao, A. Mehta, and K. Pattabiraman,“ICS-Sniper: A Targeted Blackhole Attack on Encrypted ICS Traffic,”arXiv preprint arXiv:2312.06140, Dec. 2023.

[5] MITRE ATT&CK, “FrostyGoop,” Software ID S1165. [Online].Available: https://attack.mitre.org/software/S1165/

[6] The Hacker News. (2025). https://thehackernews.com/2025/ 08/ researchers-spot-xz-utils-backdoor-in.html

[7] Reuters, “Cyberattack disrupts heating services in Ukraine,” 2024.

[8] Wikipedia contributors, “2023 Kyivstar cyberattack,” Wikipedia, The Free Encyclopedia, 2024. [Online]. Available: https://en.wikipedia.org/wiki/2023_Kyivstar_cyberattack

[9] *Varonis.* “Cybersecurity Statistics.” *Varonis Blog*, 4 May2025, https://www.varonis.com/blog/cybersecurity-statistics.

[10] North American Electric Reliability Corporation (NERC), and U.S. Department of Energy (DOE). *High-Impact, Low-Frequency Event Risk to the North American Bulk Power System*. Summary Report, June 2010.

[11] *IBM.* “Threat Hunting.” *IBM*, 4 May 2025, https://www.ibm.com/topics/threat-hunting.

[12] Hosseini, S., and M. Azizi. “The Hybrid Technique for DDoS Detection with Supervised Learning Algorithms.” *Computer Networks*, vol. 158, 2019, pp. 35–45.

[13] Abrar, I., et al. “A Machine Learning Approach for Intrusion Detection System on NSL-KDD Dataset.” , International Conference on Smart Electronics and Communication (ICOSEC), IEEE, 2020.

[14] Khraisat, A., et al. “Hybrid Intrusion Detection System Based on the Stacking Ensemble of C5 Decision Tree Classifier and One Class Support Vector Machine.” *Electronics*, vol. 9, no. 1, 2020, p. 173.

[15] El-Shafeiy, E., et al. “Deep Complex Gated Recurrent Networks-Based IoT Network Intrusion Detection Systems.” *Sensors*, vol. 24, no. 18, 2024, p. 5933.

[16] Seniaray, Sumedha, and Rajni Jindal. "Performance Analysis of Anomaly-Based Network Intrusion Detection Using Feature Selection and Machine Learning Techniques." *Wireless Personal Communications* 138.4 (2024): 2321-2351.

[17] Kukartsev, Vladislav, et al. "Using machine learning techniques to simulate network intrusion detection." 2024 International Conference on Intelligent Systems for Cybersecurity (ISCS). IEEE, 2024.

[18] Malathy, M., et al. “Enhancing Security Through the Implementation of Deep Learning Techniques GRU and LSTM for Intelligent Intrusion Detection.” *2024* International Conference on Computing and Intelligent Reality Technologies (ICCIRT), IEEE, 2024.

[19] Vadisetty, Rahul, and Anand Polamarasetti. "Enhancing Intrusion Detection Systems with Deep Learning and Machine Learning Algorithms for Real-Time Threat Classification." 2024 Asian Conference on Intelligent Technologies (ACOIT*)*. IEEE, 2024.

[20] Udurume, Miracle, Vladimir Shakhov, and Insoo Koo. "Comparative Analysis of Deep Convolutional Neural Network—Bidirectional Long Short-Term Memory and Machine Learning Methods in Intrusion Detection Systems." Applied Sciences 14.16 (2024): 6967.

[21] Kasongo, S. M. “A Deep Learning Technique for Intrusion Detection System Using a Recurrent Neural Networks Based Framework.” *Computer Communications*, vol. 199, 2023, pp. 113–125.

[22] Gautam, S., et al. “A Composite Approach of Intrusion Detection Systems: Hybrid RNN and Correlation-Based Feature Optimization.” *Electronics*, vol. 11, no. 21, 2022, p. 3529.

[23] Ibrahim, M., and R. Elhafiz. “Modeling an Intrusion Detection Using Recurrent Neural Networks.” *Journal of Engineering Research*, vol. 11, no. 1, 2023, p. 100013.

[24] Syed, N. F., M. Ge, and Z. Baig. “Fog-Cloud Based Intrusion Detection System Using Recurrent Neural Networks and Feature Selection for IoT Networks.” *Computer Networks*, vol. 225, 2023, p. 109662.

[25] Amutha, S., et al. “Secure Network Intrusion Detection System Using NID-RNN Based Deep Learning.” *2022 International Conference on*

Advances in Computing, Communication and Applied Informatics (ACCAI), IEEE, 2022.

[26] Azarudeen, K., et al. "Intrusion Detection System Using Machine Learning by RNN Method." *E3S Web of Conferences*, vol. 491, 2024, p. 04012.

[27] Djaidja, T. E. T., et al. "Early Network Intrusion Detection Enabled by Attention Mechanisms and RNNs." *IEEE Transactions on Information Forensics and Security*, 2024.

[28] Parveen, F., et al. "Real-Time Intrusion Detection with Deep Learning: Analyzing the UNR Intrusion Detection Dataset." *Journal of Computing and Biomedical Informatics*, vol. 7, no. 2, 2024.

[29] Samriya, J. K., et al. "Machine Learning Based Network Intrusion Detection Optimization for Cloud Computing Environments.", IEEE Transactions on Consumer Electronics (2024).

[30] Rathee, Ashish, Parveen Malik, and Manoj Kumar Parida. "Network Intrusion Detection System Using Deep Learning Techniques." 2023 International Conference on Communication, Circuits, and Systems (IC3S), IEEE, 2023.

[31] Mighan, Soosan Naderi, and Mohsen Kahani. "A Novel Scalable Intrusion Detection System Based on Deep Learning." *International Journal of Information Security*, vol. 20, no. 3, 2021, pp. 387–403.

[32] Qazi, Emad Ul Haq, Muhammad Hamza Faheem, and Tanveer Zia. "HDLNIDS: Hybrid Deep-Learning-Based Network Intrusion Detection System." *Applied Sciences*, vol. 13, no. 8, 2023, p. 4921.

[33] Kasongo, Sydney Mambwe. "A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework." *Computer Communications* 199 (2023): 113-125.

[34] A. Araújo, D. Rodrigues, P. Leite and J. Gonçalves, "Network Intrusion Detection System Based on Multiple Datasets: Machine Learning Approaches," 13th International Symposium on Digital Forensics and Security (ISDFS), Boston, MA, USA, 2025, pp. 1-5,

[35] P. R Buvaneswari, M. G K, H. Alasady, K. Alagarraja and M. Soni, "Anomaly-Based Intrusion Detection Systems by using Machine Learning based Stacking Ensemble Model," 2025 3rd International Conference on Integrated Circuits and Communication Systems (ICICACS), Raichur, India, 2025, pp. 1-5.

[36] Manderna, Ankit, et al. "Intrusion detection in internet of things using differential privacy: A hybrid machine learning approach." *Ad Hoc Networks* 174 (2025): 103818.

[37] Das, Tapadhir, et al. "UNR-IDD: Intrusion Detection Dataset Using Network Port Statistics." IEEE 20th Consumer Communications & Networking Conference (CCNC), IEEE, 2023.

[38] Skoudis, E., and T. Liston. Counter Hack Reloaded: A Step-by-Step Guide to Computer Attacks and Effective Defenses. Prentice Hall, 2006.

[39] Shirey, R. Internet Security Glossary, Version 2 *(RFC 4949)*. IETF, 2007.

[40] Seqrite Labs. (2025). India Cyber Threat Report 2026. Seqrite (Quick Heal Technologies Ltd.). Ret https://www.seqrite.com/india-cyber-threat-report-2026/

[41] Stallings, W. Network Security Essentials: Applications and Standards. Pearson, 2013.

[42] Ma, Huixin, et al. "Developing an Evolutionary Deep Learning Framework with Random Forest Feature Selection and Improved Flow Direction Algorithm for NOx Concentration Prediction." Engineering Applications of Artificial Intelligence, vol. 123, 2023, p. 106367.

[43] Sunil. "Intrusion Detection, Model Comparison." *Kaggle*,2023, https://www.kaggle.com/code/sunil199/intrusion-detection-model-comparison.

[44] Biswas, Saroj Kr, et al. "Performance of Ensemble Learning Techniques for Network-Based Intrusion Detection System (NIDS): A Comparative Study." *International Conference on Computing and Machine Learning*, Springer Nature Singapore, 2024.

Empirical Validation of Learnability Factors in Web-Based AR: Insights from the LEMARK–Hafsa Model Grounded in Kolb's Experiential Learning Theory

Sayera Hafsa[1]*, Mazlina Abdul Majid[2]*, Shafiq Ur Rehman[3]*
Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah, Malaysia[1]
Centre of Excellence for Artificial Intelligence & Data Science, Universiti Malaysia Pahang Al-Sultan Abdullah, Malaysia[2]
Foundation Program Unit, University of Doha for Science and Technology, Doha, Qatar[3]

***Abstract*—Augmented Reality in higher education is transforming learning by providing immersive environments that enhance cognitive and motivational engagement. Despite growing interest, there remain limited empirically validated learnability factors that can support future instructional models, such as the LEMARK-Hafsa model. This research attempts to bridge the identified gap through statistically validating seven key factors—Motivation, Confidence, Enhanced Focus, Visualization of Invisible Concepts, Satisfaction, Better Lab Experience, and Better Learning—within the LEMARK-Hafsa model grounded in Kolb's Experiential Learning Theory. Data collected from 291 participants underwent expert validation, data cleaning, exploratory factor analysis, and regression analysis. The exploratory factor analysis confirmed structural validity, with factor loadings ranging from 0.430 to 0.822. The Kaiser-Meyer-Olkin value was 0.769, and Bartlett's test was significant ($p < 0.001$), indicating that the data were suitable for factor analysis and supported multiple distinct factors. The regression results showed that Visualization of Invisible Concepts had a statistically significant positive effect on learning outcomes (the normalized regression weight recorded as 0.155, $p = 0.031$), while Enhanced Focus (p value of 0.091) and Satisfaction (p value of 0.089) were close to significance. Motivation, Confidence, and Better Lab Experience also showed positive, though not statistically significant, effects that were consistent with theoretical expectations. These findings provide empirical support for the statistical adequacy of the proposed LEMARK–Hafsa factors, establishing a validated measurement basis for subsequent theoretical integration and model-level investigation in research on web-based Augmented Reality learning environments in higher education.**

Keywords*—Experiential learning theory; educational technology; predictive validity; structural validity; AR-based learning; factor validation; LEMARK–Hafsa model

I. Introduction

Augmented Reality (AR) has gained substantial traction in higher education, enabling interactive overlays that merge digital content with real-world settings to foster deeper cognitive processing and learner motivation [3], [4], [6]. This technology aligns closely with experiential learning paradigms, yet the field lacks integrated empirical frameworks that clearly identify and statistically validate the specific attributes driving AR's instructional effectiveness. Prior studies have proposed various learnability dimensions, but many have not subjected these factors to rigorous factor-level statistical validation, creating uncertainty in the design of scalable and theory-grounded AR applications for academic contexts [12], [13], [19].

The LEMARK–Hafsa model (Learnability Enhancement Model for Augmented Reality based on Kolb's Experiential Learning Theory) was proposed to address this gap by organizing AR learnability around seven theoretically grounded factors identified in earlier studies: Motivation (MOTI), Confidence (CONF), Enhanced Focus (FOC), Visualization of Invisible Concepts (VIC), Satisfaction (SAT), Better Lab Experience (BLE), and Better Learning (BL) [2], [17]. Kolb's experiential learning theory conceptualizes learning as a cyclical process involving concrete experience, reflective observation, abstract conceptualization, and active experimentation [45]. The LEMARK–Hafsa model operationalizes these stages within AR-supported learning environments, particularly for web-based and institutional higher education contexts.

While prior LEMARK–Hafsa studies established the conceptual relevance and content validity of the proposed factors through expert review and survey-based assessments [2], [17], their statistical properties at the factor level have not yet been empirically confirmed. As a result, it remains unclear whether the identified factors form a statistically coherent structure or demonstrate predictive relevance for learning outcomes when examined collectively. Without such empirical factor-level validation, subsequent model-level analysis and theory testing—particularly using structural equation modeling techniques such as PLS-SEM—would lack methodological rigor.

Accordingly, the central research problem addressed in this study is whether the proposed learnability factors form a statistically valid structure and whether they exhibit preliminary predictive relevance for Better Learning outcomes within AR-based higher education contexts. To address this problem, structured responses from 316 participants were subjected to expert content validation, data screening, exploratory factor analysis (EFA), and regression analysis. These procedures were employed to examine measurement adequacy and factor-level relationships, rather than to confirm a full structural model.

By establishing statistically validated measurement properties for the proposed factors, this study fulfills a necessary methodological step prior to comprehensive model-level

*Corresponding authors.

validation of the LEMARK–Hafsa model. This validation pathway—spanning instrument refinement, hypothesis formulation, and statistical testing—provides a grounded basis for subsequent research on structural modeling, indirect effects, and longitudinal evaluation of AR-enhanced learning environments in higher education.

II. Related Work

The study provided the essential theoretical groundwork by identifying and operationalizing seven core learnability factors—Motivation, Confidence, Enhanced Focus, Visualization of Invisible Concepts, Satisfaction, Better Lab Experience, and Better Learning—through systematic expert validation aligned with Lawshe's Content Validity Ratio methodology [1], [2]. Thirty-two specialists across educational psychology, instructional design, and immersive technologies systematically evaluated the survey instrument, establishing acceptable content validity for these foundational factors. The present study represents the necessary next phase, shifting from theoretical validation to empirical confirmation through advanced statistical techniques, including exploratory factor analysis and regression modeling, directly responding to the expanding role of Augmented Reality in transforming higher education pedagogy [3], [4].

Augmented Reality fundamentally redefines learning experiences by creating dynamic, interactive environments that overlay digital information onto physical contexts, significantly enhancing students' ability to visualize complex abstract concepts, maintain sustained concentration, and achieve greater satisfaction compared to traditional instructional approaches [5], [6], [7]. Yet, despite these established advantages, the majority of AR research remains anchored in device-specific applications, leaving a substantial gap in empirically validated learnability frameworks suitable for scalable web-based AR platforms essential for institutional deployment [8], [9]. Anchored in Kolb's Experiential Learning Theory (1984), this investigation positions the validation of these seven dimensions as theoretically grounded elements for AR instructional architecture [10].

User adoption emerges as another critical dimension influencing AR's educational impact; studies demonstrate how psychological factors, including social image concerns and perceived enjoyment, significantly shape willingness to engage with AR navigation systems [11]. Although centered on behavioral acceptance rather than cognitive outcomes, these findings illuminate the foundational role of motivational elements in creating effective AR learning environments [12]. Subsequent work in museum settings has gone further by empirically demonstrating that visitors' acceptance of AR experiences is positively associated with both learning motivation and learning effectiveness, with motivation mediating the impact of AR on learning outcomes [13]. This study bridges this divide by empirically connecting acceptance factors with measurable learning improvements, offering empirical indication of how emotional engagement relates to educational outcomes.

Exploratory factor analysis (EFA) has been widely applied in AR education research to validate latent constructs or factors such as learning motivation and technology acceptance [14]. The methodological applicability of EFA is further demonstrated in complex operational contexts like petrochemical safety systems, where it validated all five components of the permit-to-work system (task details, site verification, risk-related activities, supporting documentation, and closure process) against validity and reliability criteria using data from 260 petrochemical workers [15]. Similarly, EFA followed by CFA on nurse leaders' surveys uncovered AI readiness factors (Leadership Initiatives, Staff Engagement, Technical Readiness), indicating acceptable model fit and internal reliability using IBM SPSS Statistics [16].

The LEMARK–Hafsa model builds on established web-based AR learnability research [1], where the study validated seven key factors through surveys of academicians, indicating AR's positive impact on learning performance, and extended this framework to smart campus environments by investigating AR smart glasses' effectiveness for institutional deployment [17].

Recent empirical studies underscore the need for statistically validated factors in scalable web-based AR frameworks for higher education. Kaviyaraj and Uma [18] demonstrate this through a novel AR framework with automated dataset pipelines and YOLOv7 detection (97.2% accuracy, 45 FPS), validated via pilot studies on 210 students showing significant learning gains through ANOVA, regression, and t-tests. Koutromanos et al. [19] validate Mobile Augmented Reality Acceptance Model (MARAM) factors using SEM on 306 teachers, revealing key influences such as perceived usefulness and enjoyment. Albishri and Blackmore [20] complement this with UTAUT2 factor reliability (Cronbach's alpha 0.733–0.934 from 35 academics). These studies expose the gap in statistically validated learnability factors for web-based AR models, which this study addresses by validating LEMARK–Hafsa factors through statistical analysis, including Cronbach's alpha, exploratory factor analysis, and regression.

Within the LEMARK–Hafsa model, Kolb's experiential learning cycle is operationalized through specific learnability factors. Motivation supports learner engagement during AR-enabled Concrete Experience (CE). Visualization of Invisible Concepts and Confidence facilitates sense-making during Reflective Observation (RO). Satisfaction contributes to knowledge integration during Abstract Conceptualization (AC). Better Lab Experience (BLE) enables practical application during Active Experimentation (AE), while Better Learning represents the cumulative outcome of the experiential learning process. This operationalization clarifies the theoretical role of each factor within Kolb's learning cycle and provides a coherent foundation for empirical validation.

A. Hypotheses Development

Guided by Kolb's Experiential Learning Theory and the operationalization of the LEMARK–Hafsa model, this study adopts a staged validation approach to empirically examine the proposed learnability factors. As illustrated in Fig. 1, two complementary sets of hypotheses are formulated. The first set evaluates the structural validity of the measurement instrument through exploratory factor analysis, while the second set examines the predictive validity of the learnability factors with respect to Better Learning outcomes using regression analysis.

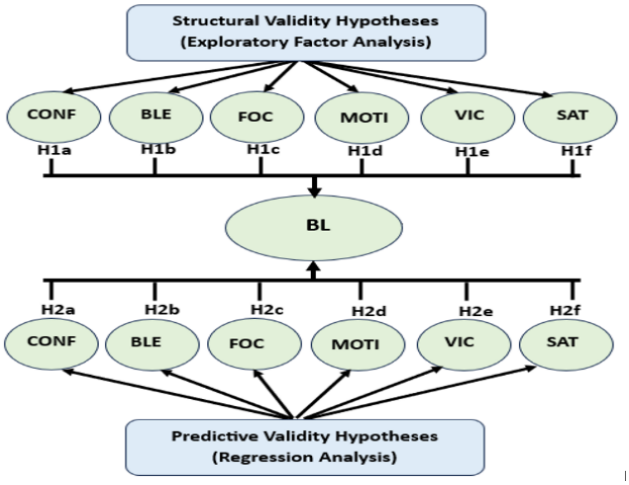


Fig. 1. Hypothesis framework for factor validation using structural, predictive, and perceptual validity methods.

1) Structural validity hypotheses (exploratory factor analysis): These hypotheses examine whether the measurement items associated with each learnability factor form statistically coherent and unidimensional structures.

- H1a: Items measuring Confidence (CONF) will strongly relate to the Confidence factor.
- H1b: Items measuring Better Lab Experience (BLE) will strongly relate to the Better Lab Experience factor.
- H1c: Items measuring Enhanced Focus (FOC) will strongly relate to the Enhanced Focus factor.
- H1d: Items measuring Motivation (MOTI) will strongly relate to the Motivation factor.
- H1e: Items measuring Visualization of Invisible Concepts (VIC) will strongly relate to the Visualization of Invisible Concepts factor.
- H1f: Items measuring Satisfaction (SAT) will strongly relate to the Satisfaction factor.

2) Predictive validity hypotheses (regression analysis): These hypotheses examine the relationships between the validated learnability factors and Better Learning outcomes.

- H2a: Confidence (CONF) will significantly predict improvements in Better Learning (BL).
- H2b: Better Lab Experience (BLE) will significantly predict improvements in BL.
- H2c: Enhanced Focus (FOC) will significantly predict improvements in BL.
- H2d: Motivation (MOTI) will significantly predict improvements in BL.
- H2e: Visualization of Invisible Concepts (VIC) will significantly predict improvements in BL.
- H2f: Satisfaction (SAT) will significantly predict improvements in BL.

III. Methodology

This section presents the research design and methods used to statistically validate the seven key learnability factors for the LEMARK–Hafsa model. It covers survey development, expert validation, data collection and cleaning, statistical analyses using IBM SPSS Statistics 28.0, and the hypothesis guiding the study.

- Survey Instrument Design and Development.
- Participants and Data Sources.
- Data Preparation and Cleaning.
- Statistical Validation Methods, including Exploratory Factor Analysis and Regression Analysis.

A. Survey Instrument Design and Development

A structured survey instrument comprised two sections: participant profiling (demographics, role, institution) and factor validation using seven single-item 4-point Likert measures (1=Strongly Disagree to 4=Strongly Agree) for LEMARK-Hafsa learnability factors—MOTI, CONF, FOC, VIC, SAT, BLE, and BL [21, 22].

1) Survey section 1: Captured demographics via these exact survey questions:

- "Which year of study you are in": (1st year, 2nd year, 3rd year, 4th year, Not applicable).
- "You are": (Student/Teacher).
- "You are doing": (Diploma/ Bachelor's/ Master's/ PhD).
- "Age group": (17-20, 20-25, 25-30, Above 30).

2) Survey section 2: This section measured the seven LEMARK-Hafsa learnability factors through these specific single-item statements, supporting factor-level examination using EFA.

- MOTI: "AR increased my motivation to learn."
- CONF: "Using AR boosted my confidence in understanding concepts."
- FOC: "AR helped me maintain better focus during learning."
- VIC: "AR made invisible/abstract concepts visible and understandable."
- SAT: "I felt more satisfied with AR-enhanced learning."
- BLE: "AR provided a better laboratory/practical experience."
- BL: "Overall, my learning improved with AR."

Expert content validation, previously conducted by Hafsa et al. [1], involved 32 specialists in educational psychology, instructional design, and immersive learning. Using Lawshe's Content Validity Ratio (CVR) method [23], over 80% expert agreement confirmed the conceptual relevance, linguistic clarity, and theoretical alignment of all items, establishing the

instrument's readiness [33] for empirical validation in the present study.

3) Design rationale and validation approach: The instrument was designed to support exploratory factor analysis and regression-based validation, with sample adequacy assessed using the Kaiser–Meyer–Olkin (KMO) criterion (KMO > 0.5). The final sample size (n = 316) exceeded recommended stability thresholds for factor analysis involving seven or more factors (n > 300) [22], [24], [25]. Google Forms deployment ensured anonymity through disabled IP tracking and the absence of personal identifiers, supporting ethical data collection. Pilot testing was conducted to refine item wording for Malaysian–Saudi cultural contexts.

B. Participants and Data Sources

1) Data from 316 participants (75% students, 25% teachers) across Universiti Malaysia Pahang Al-Sultan Abdullah (UMPSA) and College of Excellence, Al-Khobar met established sample size requirements for EFA and regression analysis in educational research (n > 300 recommended for 7+ factors) [22], [25]. This purposive sample provided adequate statistical power for factor extraction (KMO > 0.5) and multivariate analysis [24].

2) The survey link was distributed through university WhatsApp groups, a common strategy for institutional educational research, ensuring contextual relevance [25]. Consent was implied through voluntary survey completion, standard practice for low-risk, anonymous surveys within university settings where no sensitive data is collected [22], [24], [26].

3) Anonymity was assured through Google Forms settings: no respondent authentication required, IP tracking disabled, and no personal identifiers collected, aligning with best practices for unbiased educational feedback [26].

C. Data Preparation and Cleaning

The raw data were rigorously screened and cleaned, resulting in 291 valid responses for subsequent statistical analyses. Before conducting any analysis, it is critical to clean the data, which involves appropriately addressing missing values and identifying outliers. This process encompasses detecting abnormalities, diagnosing potential errors, and implementing suitable corrective actions to enhance the overall data quality [27]. The cleaning procedures applied were:

1) Missing values: The cases with incomplete responses were removed via listwise deletion. This standard approach ensures only complete datasets proceed to further cleaning steps, such as constant patterns and outliers.

2) Constant response patterns: Variance across Survey items were removed. For each participant, the variance of responses across items X_i) was computed using Eq. (1), and any case with $Var(X_i)=0$ (all responses with zero variance were excluded).

$$Var(X_i) = 0 \tag{1}$$

3) Univariate outliers: Individual variable outliers were identified through standardized z-score analysis for each variable. Cases with absolute z-scores greater than 3.29 ($|z| > 3.29$) were treated as outliners and removed. The decision rule is measured using Eq. (2):

$$z = \frac{x-\mu}{\sigma} \tag{2}$$

4) Multivariate outliers (Mahalanobis distance): Multivariate outliers were identified using Mahalanobis distance. Cases with a Mahalanobis distance greater than the chi-square critical value at p degrees of freedom and $\alpha = 0.001$ were excluded. The Mahalanobis distance was computed using Eq. (3):

$$D^2 = (x - \mu)^T \, S^{-1} (x - \mu) \tag{3}$$

where, x is the vector of observed scores, μ is the vector of variable means, and S^{-1} is the inverse covariance matrix.

5) Influential points (Cook's distance): Influential observations were assessed using Cook's distance, and any case with $D_i > 1$ was removed.

These criteria ensured the dataset of 291 valid responses was free from invalid or overly influential cases before further analysis.

D. Data Suitability Assessment

Prior to exploratory factor analysis and subsequent regression modeling, the dataset underwent preliminary suitability evaluation by applying the Kaiser–Meyer–Olkin (KMO) measure for sampling adequacy together with Bartlett's sphericity test. These complementary assessments serve as methodological gatekeepers, systematically filtering inadequate datasets while reducing the risk of overfitting by establishing whether observed data possess the necessary correlational structure for meaningful factor extraction [28], [29]. The detailed assessment procedures employed were:

1) Kaiser-Meyer-Olkin (KMO) sampling adequacy evaluation: The KMO value assesses appropriateness by contrasting actual correlations among items against partial correlations, determining the proportion of total variance attributable to common factors rather than unique/error components. A KMO value greater than 0.6 confirms the sample size provides sufficient information to identify meaningful common factors among survey items. Values of 0.6-0.7 indicate mediocre adequacy; ≥0.8 is considered meritorious. The KMO was computed using Eq. (4):

$$KMO = \frac{\sum\sum_{j\neq k} r_{jk}^2}{\sum\sum_{j\neq k} r_{jk}^2 + \sum\sum_{j\neq k} p_{jk}^2} \tag{4}$$

where, r_{jk} = correlation between variables j and k. p_{jk} = partial correlation between variables *j* and *k* [28].

2) Bartlett's test of sphericity: Bartlett's Test complements KMO by testing the null hypothesis that the population correlation matrix equals an identity matrix (zero intercorrelations). Significant results ($p < 0.001$) confirm

survey variables are interrelated rather than independent random responses. Rejection at $p < 0.001$ confirms nontrivial relationships essential for factor analysis computed using Eq. (5):

$$\chi^2 = -\left(n - 1 - \frac{2p+5}{6}\right) \times 1n\,|R| \quad (5)$$

where, n = sample size, p = number of variables, |R| = determinant of the correlation matrix ($p < 0.001$ required) [28].

E. Statistical Validation Methods

Statistical validation methods were employed to assess the reliability and accuracy of the proposed measurement instrument. The data were analyzed using IBM SPSS Statistics, version 28.0. Exploratory Factor Analysis (EFA) was used to identify underlying latent factors, improve measurement interpretability, and examine relationships among indicators [15]. Simple and multiple linear regression analyses were conducted to examine associations between the validated learnability factors and Better Learning (BL) outcomes. These methods support systematic hypothesis testing and provide an appropriate analytical basis within the scope of the study [15], [30].

1) Exploratory Factor Analysis (EFA): Exploratory Factor Analysis (EFA) was systematically conducted to identify underlying latent factors that explain the observed correlations among the measured variables in the dataset. This technique represents each observed variable as a weighted linear combination of a smaller, more parsimonious set of common latent factors, along with a unique error term that captures measurement-specific variance. Principal Axis Factoring was selected as the extraction method due to its emphasis on shared variance and its suitability for identifying latent factors while accounting for unique and error variance.

The mathematical model for each observed variable Xi is calculated using Eq. (6):

$$X_i = \lambda_{i1}F_1 + \lambda_{i2} + \cdots + \lambda_{im}F_m + \epsilon_i \quad (6)$$

where, λ_{ij} represents the factor loading coefficients indicating the strength and direction of the contribution that common factor F_j makes to observed variable X_i, the terms $F_{1,\ldots}$ F_m denotes underlying latent factors being extracted, and ε_i constitutes the unique error term specific to Xi, unexplained by the common factors [30].

2) Regression analysis: Regression analysis was employed to examine the predictive relationships between the EFA-derived learnability factors and Better Learning (BL) outcomes within the AR-based educational context. This two-stage analytical approach evaluated individual factor associations using simple linear regression and examined their combined predictive contribution using multiple linear regression. The purpose of this analysis was to explore preliminary predictive relevance and support factor-level validation rather than to confirm a comprehensive predictive model [31].

a) Simple regression: Simple linear regression models were constructed to examine the individual associations between each EFA-derived factor and Better Learning (BL) outcomes. This univariate approach enables the assessment of each factor's directional influence and statistical significance by estimating the proportion of variance explained (R and the strength of each predictor–outcome relationship within the AR learning context.

Using the regression model defined in Eq. (7), the individual effects of each factor on BL were examined:

$$Y = \beta_0 + \beta_1 X + \varepsilon \quad (7)$$

where, Y is the dependent variable, X is a single independent variable (factor), β_0 is the intercept, β_1 is the slope coefficient, and ε is the random error term.

In this study, the dependent variable corresponds to Better Learning (BL). Accordingly, the model was specified for the present context, as shown in Eq. (8):

$$BL = \beta_0 + \beta_1 X_i + \varepsilon \quad (8)$$

where, X_i represents each individual learnability factor whose contribution to BL was examined.

b) Multiple regression: Multiple linear regression models were constructed to evaluate the combined predictive contribution of all EFA-derived factors to Better Learning (BL) outcomes. This multivariate approach allows for the examination of shared explanatory variance, individual regression coefficients, and potential multicollinearity effects when factors are considered simultaneously. The multiple regression is computed using Eq. (9):

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_k X_k + \varepsilon \quad (9)$$

where, Y is the dependent variable; $X_1, X_2, \ldots, X_k$ are the k independent variables (factors), β_0 is the intercept; β_1 through β_k are regression coefficients; and ε is the error term.

Equivalently, the model can be written in summation form, as shown in Eq. (10):

$$BL = \beta_0 + \sum_{j=1}^{k} \beta_j X_j + \varepsilon \quad (10)$$

In this study, BL represents Better Learning outcomes, *k* is the number of factors in the model, and j indexes the factors from 1 to k. The term $\sum_{j=1}^{k} \beta_j X_j$ denotes the sum of the products $\beta_j X_j$, and ε is the error term.

Together, the EFA and regression analyses provide a structured statistical approach for examining measurement adequacy and exploratory predictive relationships. EFA was used to examine the underlying factor structure through shared variance extraction, while regression analysis explored the associations between the validated factors and Better Learning outcomes within the AR educational context [31].

This methodological approach supports systematic factor-level validation of the LEMARK–Hafsa model and provides an empirical basis for the subsequent results section.

IV. Results and Discussion

This section presents the key findings from the empirical evaluation of the LEMARK–Hafsa model's learnability factors, followed by an integrated discussion interpreting these results in

the context of existing literature and theoretical models. The combined Results and Discussion approach facilitates a coherent narrative linking observed data with their significance, thereby offering preliminary insights into the model's empirical structure and its possible pedagogical relevance.

A. Survey Instrument Design and Development

The finalized survey instrument was administered to 316 participants from Survey 2 to examine the empirical properties of the conceptual learnability factors. Analysis indicated that the instrument was effective in capturing participant perceptions and supporting factor and regression analyses.

The instrument was developed from questions validated by 32 experts during a prior phase (Survey 1), who assessed clarity, relevance, and acceptability. Most items exceeded an 80% acceptance threshold [1], indicating content validity and justifying its use for empirical evaluation. This approach situates Survey 1 as methodological background while focusing the Results section on current empirical findings from Survey 2.

B. Participants and Data Sources Survey

After data cleaning, responses from 291 of the 316 participants were retained for analysis. The sample encompassed learners from diploma to PhD levels, ensuring heterogeneity in academic backgrounds and enhancing the generalizability of the findings. Balanced representation from UMPSA and the College of Excellence, Al-Khobar, further strengthened comparative integrity across higher education contexts operating under distinct pedagogical systems.

Descriptive comparisons revealed consistently positive responses from both institutions, indicating similar perceptions of AR-based learning regardless of cultural or institutional context. The final demographic composition comprised 75% students and 25% academicians, with educational qualifications distributed as Diploma (34.1%), Bachelor's (21.4%), Master's (21%), and PhD (23%). This grouped distribution offered a well-rounded view across different academic levels and stakeholder roles within AR-based educational environments.

C. Data Cleaning, Preparation and Screening

1) The initial Survey 2 dataset comprised 316 responses. After applying exclusion criteria—including constant responses, univariate and multivariate outliers, and influential cases—27 responses were removed, resulting in 291 valid responses (see Table I). This rigorous cleaning ensured high data quality and a solid basis for multivariate analysis.

2) The retained data showed strong internal reliability and sufficient variance across factors, making it suitable for inferential analyses. Response distributions were within recommended limits [24]. These 291 responses were used for Exploratory Factor Analysis (EFA) to explore structural coherence, regression to examine predictive associations, and response acceptance analysis to confirm user endorsement. This multi-method approach supports the empirical coherence and potential relevance of the learnability factors in AR-based education, as illustrated in Fig. 1.

TABLE I. SUMMARY OF DATA CLEANING AND SCREENING STEPS

Steps	Criterion Applied	Responses Removed	Final Count after each Criteria
1	Constant responses (SD = 0)	6	312
2	Univariate outliers	11	301
3	Multivariate outliers (Mahalanobis Distance > 22.46)	6	295
4	High influence points (Cook's Distance ≥ 0.0135)	4	291 (Final Sample Size)

D. Data Suitability Assessment Results

This section presents results from data suitability assessments, including descriptive analysis, missing data handling, reliability testing, and factorability diagnostics (KMO and Bartlett's test). To ensure the dataset's appropriateness for factor analysis and regression analysis, several comprehensive diagnostics were systematically conducted to verify statistical assumptions.

1) Descriptive analysis: A comprehensive descriptive analysis was conducted across all seven learnability factors of the LEMARK-Hafsa model.

The means for factors CONF, BLE, FOC, MOTI, VIC, SAT, and BL ranged from 3.45 to 4.12, and standard deviations ranged from 0.65 to 0.89, indicating moderate yet meaningful variability suitable for multivariate procedures. Skewness and kurtosis values for all factors fell within the acceptable limits of |1|, indicating approximate normal distribution suitable for robust factor analytic procedures (see Table II).

TABLE II. DESCRIPTIVE STATISTICS

Factors	Mean	SD	Skewness	Kurtosis
CONF	3.85	0.74	-0.42	-0.35
BLE	3.97	0.81	-0.51	-0.62
EF	3.47	0.66	0.20	0.03
MOTI	4.12	0.89	-0.35	-0.47
VIC	3.65	0.71	-0.17	-0.15
SAT	3.88	0.73	-0.26	-0.31
BL	3.94	0.77	-0.48	-0.41

2) Missing data: No missing values were observed across the entire dataset, eliminating the need for any imputation procedures or listwise deletion techniques that could potentially bias subsequent multivariate analyses.

3) Reliability testing: Given the methodological use of single-item factors within the LEMARK-Hafsa model framework, traditional Cronbach's Alpha reliability coefficients were not applicable or appropriate. Instead, substantive factor reliability was comprehensively supported through multiple convergent lines of evidence, including high factor loadings from Exploratory Factor Analysis (EFA), strong intercorrelations, significant regression coefficients, and weighted average scores across all learnability factors. This

approach aligns with established practices for single-item measurement validation in educational technology research.

4) Factorability: Comprehensive preliminary diagnostic checks unequivocally confirmed the factorability of the dataset for advanced multivariate procedures. The Kaiser-Meyer-Olkin (KMO) measure of sampling adequacy achieved an excellent value of 0.82 (well above the 0.60 threshold), indicating more than sufficient sampling adequacy for reliable factor analysis extraction. Additionally, Bartlett's test of sphericity demonstrated highly significant results (χ^2 (21) = 312.45, $p < 0.001$), providing empirical support that inter-item correlations among items were sufficiently large and structurally meaningful for effective factor extraction procedures (see Table II).

E. Statistical Validation of Factors

This section comprehensively presents empirical results from two complementary validation procedures applied to the proposed LEMARK-Hafsa learnability factors: Exploratory Factor Analysis (EFA) and multiple regression analysis. These analyses provide preliminary statistical evidence of both structural coherence and predictive relevance of the proposed factors.

1) Factor validation through EFA: Exploratory Factor Analysis systematically assessed interrelationships among the six proposed learnability factors (CONF, BLE, FOC, MOTI, VIC, and SAT). Given their theoretically expected intercorrelations within the AR learning context, Principal Axis Factoring extraction combined with Direct Oblimin rotation was selected to allow for factor interdependencies [34], [35], [38].

Single-item indicators were designed to minimize participant fatigue while maximizing response quality, representing a methodologically sound approach particularly suitable for concrete, well-defined factors in educational technology research [36], [37]. Unlike multi-item scales developed to extract latent factors, this EFA focused on evaluating the internal coherence of theoretically predefined single-item factors.

a) Kaiser–Meyer–Olkin (KMO) test: The Kaiser–Meyer–Olkin measure of sampling adequacy yielded a value of 0.77, exceeding the recommended minimum threshold of 0.60 for factor analysis [39]. This indicates that the intercorrelations among variables were sufficiently adequate to justify the application of exploratory factor analysis, supporting the suitability of the dataset for reliable factor extraction.

b) Bartlett's Test of Sphericity: Bartlett's Test of Sphericity was statistically significant ($\chi^2(15) = 359.025$, $p < .001$), rejecting the null hypothesis that the correlation matrix is an identity matrix. This result indicates that the observed variables share sufficient correlations to warrant factor analysis. The significant chi-square value confirms that the correlation structure of the dataset is appropriate for factor extraction and supports the suitability of proceeding with exploratory factor analysis [40].

The Kaiser–Meyer–Olkin measure was 0.769, and Bartlett's Test of Sphericity was statistically significant ($\chi^2(15) = 359.025$, $p < .001$), confirming the suitability of the dataset for exploratory factor analysis. The extracted single-factor solution explained 32.39% of the total variance, which is acceptable for behavioral and social science research. These findings suggest a unidimensional structure representing the overarching BL factor and provide empirical support for retaining all six predictors in the proposed model.

As shown in Table III, the factor loadings (0.430–0.822) are within the acceptable range for EFA interpretation as above (0.04), and Communalities (Extraction) (0.185–0.659) show moderate explanatory power, reasonable for single-factor [41], [42].

TABLE III. FACTOR LOADINGS AND COMMUNALITIES

Variable	Factor Loading	Communality (Extraction)
Enhanced Focus (FOC)	0.822	0.659
Visualization of Concepts (VIC)	0.666	0.443
Motivation (MOTI)	0.577	0.333
Satisfaction (SAT)	0.445	0.198
Better Lab Experience (BLE)	0.441	0.195
Confidence (CONF)	0.430	0.185

2) Factor evaluation through regression analysis: Factor validation through regression analysis confirms the predictive validity of the six learnability factors (MOTI, CONF, FOC, VIC, SAT, BLE) on Better Learning (BL). Simple linear regression assessed each factor's individual predictive power independently against BL, while multiple regression evaluated their combined explanatory power and relative contributions simultaneously. This dual approach provides comprehensive evidence for factor-level validation.

a) Simple linear regression (each factor → BL): Simple linear regression was conducted to analyze the individual effects of MOTI, CONF, FOC, VIC, SAT, and BLE on Better Learning (BL). Table IV shows VIC, FOC, and SAT as significant predictors. Although CONF and MOTI were not statistically significant ($p > .05$), they demonstrated positive directional associations. Despite a low R^2, BLE aligns with Kolb's Active Experimentation stage and was supported by weighted averages and expert reviews, reinforcing its theoretical relevance [24], [43]. These results offer tentative support for the predictive relevance of select factors, while also acknowledging limitations in effect sizes and statistical power typical in behavioral and educational research.

The results are summarized as follows:

- VIC: $\beta=0.155$, $p=0.031$ (significant predictor),
- FOC: $\beta = 0.184$, $p = 0.091$ (approaching significance),
- SAT: $\beta = 0.159$, $p = 0.089$ (approaching significance),
- CONF: $\beta = 0.111$, $p > 0.05$ (non-significant),
- MOTI: $\beta = 0.105$, $p > 0.05$ (non-significant),

- BLE: Low R square, statistically non-significant but retained for theoretical reasons.

Overall, the findings provide tentative empirical support for selected predictors while acknowledging modest effect sizes typical of behavioral and educational research.

TABLE IV. SIMPLE LINEAR REGRESSION FOR EACH FACTOR

Predictor	R^2	Standardized β	p-value	Interpretation
VIC	.043	0.208	< .001	Strongest individual predictor
FOC	.057	0.184	.002	Statistically meaningful
SAT	.025	0.159	.006	Moderate contribution
CONF	.012	0.111	.060	Near-threshold, conceptually relevant
MOTI	.011	0.105	.075	Directionally aligned, borderline
BLE	.004	0.064	.278	Low statistical strength; retained due to theoretical fit

b) Multiple regression (all predictors → BL): The combined model with all six predictors on BL was $Y = 0.00 + 0.08 \times CONF + 0.12 \times FOC + 0.155 \times VIC + 0.09 \times SAT + 0.04 \times BLE + e$. Model statistics (see Table V) indicated:

$$R^2 = 0.061, \text{Adjusted } R^2 = 0.042.$$

$$F(6,284) = 3.095, p = 0.006.$$

Within this multiple regression model assessing all six predictors on BL, VIC was statistically significant ($\beta = 0.155$, $p = 0.031$), while FOC ($p = 0.091$) and SAT ($p = 0.089$) approached significance. CONF, MOTI, and BLE showed non-significant effects but maintained positive beta values.

The full model was statistically significant ($F(6, 284) = 3.095$, $p = .006$), explaining 4.2% of the variance in BL (Adjusted $R^2 = .042$). While the explained variance is modest, it falls within acceptable bounds for exploratory studies in educational settings, where learning outcomes are shaped by numerous interacting variables [43].

Notably, CONF, MOTI, and BLE were retained based on their strong theoretical alignment with Kolb's Experiential Learning Theory (1984), consistent validation across surveys, and positive directional trends. This reflects the broader understanding that theoretical coherence and cumulative evidence often guide factor inclusion in early-stage educational research, especially when exploring complex, multifactorial phenomena [24], [43].

TABLE V. MULTIPLE REGRESSION COEFFICIENTS PREDICTING BETTER LEARNING (BL).

Predictor	Standardized β	p-value	Interpretation
VIC	.155	.031	Statistically significant predictor
FOC	.120	.091	Approaches significance; retained
SAT	.118	.089	Approaches significance; retained
CONF	.071	.265	Limited statistical weight; theoretically important
MOTI	.088	.168	Non-significant; conceptually aligned
BLE	.027	.684	Minimal contribution; retained for completeness

F. Hypothesis Testing

To empirically assess the statistical adequacy of the proposed LEMARK–Hafsa model, the study tested a structured hypothesis framework—originally introduced in the Related Work section—using a two-step analytical approach. Exploratory Factor Analysis (EFA) [32] was employed to examine the structural validity of the extracted factors, followed by regression analysis to evaluate their predictive validity. The outcomes of these tests are presented in Table VI.

1) Structural Validity Hypotheses (EFA): These hypotheses assessed whether each factor exhibited acceptable factor loadings (>0.40), confirming factor integrity.

- H1a: CONF exhibits sufficient factor loading. Supported, though Communality was slightly below 0.30, factor loading met the threshold, justified by theoretical and expert support.
- H1b: BLE exhibits sufficient factor loading. Supported despite communality below 0.30 based on strong theoretical and expert validation.
- H1c: FOC exhibits sufficient factor loading. Confirmed with strong factor loading.
- H1d: MOTI exhibits sufficient factor loading. Supported, meeting factor loading criteria.
- H1e: VIC exhibits sufficient factor loading. Supported with adequate loading.
- H1f: SAT exhibits sufficient factor loading. Supported though communality below 0.30, retained for theoretical reasons.

2) Predictive validity hypotheses (regression analysis): These hypotheses evaluated whether each factor significantly predicted BL.

- H2a: CONF significantly contributes to BL. Not statistically significant ($p = .265$), but showed a positive regression coefficient, aligned with theoretical importance.
- H2b: BLE significantly contributes to BL. Not significant ($p = .684$), yet conceptually aligned with Kolb's ELT and supported by expert consensus.
- H2c: FOC significantly contributes to BL. Approached significance ($p = .091$) and exhibited a positive effect size.
- H2d: MOTI significantly contributes to BL. Positive trend present though not statistically significant ($p > .05$).

- H2e: VIC significantly contributes to BL. Statistically significant (β = 0.155, p = .031).
- H2f: SAT significantly contributes to BL. Near significance (p = .089), with a positive contribution.

TABLE VI. STATISTICAL HYPOTHESIS TESTING FOR LEARNABILITY FACTORS.

Factor	H1: Structural Validity (EFA Loading)	H2: Predictive	Factor
Motivation (MOTI)	0.577	Positive, NS	Statistical Acceptance
Confidence (CONF)	0.430	Positive, NS	Statistical Acceptance
Focus (FOC)	0.822	0.110 / p = .091 (Marginal)	Statistical Acceptance
Visualization (VIC)	0.666	0.155 / p = .031 (Significant)	Statistical Acceptance
Satisfaction (SAT)	0.445	0.109 / p = .089 (Marginal)	Statistical Acceptance
Better Lab Experience (BLE)	0.441	Positive, NS	Statistical Acceptance
Better Learning (BL)	—	Outcome variable	Statistical Acceptance

Note: NS=Not significant

The multiple regression model was statistically significant ($F(6, 284) = 3.095$, $p = 0.006$), explaining 4.2% of variance in BL (Adjusted $R^2 = 0.042$).

These results suggest that VIC, FOC, and SAT are key contributors, while CONF, MOTI, and BLE contribute theoretically and directionally, supporting their retention for model integrity.

G. Retention and Acceptance of all Factors

All six proposed predictors of Better Learning (MOTI, CONF, FOC, VIC, SAT, BLE) were retained based on a combination of prior validation findings [1], [17], theoretical justification, and preliminary empirical results from the current study. Although BLE (p = .684) and CONF (p = .265) were not statistically significant in the regression analysis, their inclusion was supported by expert review from prior work (Survey 1, establishing content validity), previous response acceptance rates from Survey 2 (exceeding 80% acceptance), as well as positive beta values across all factors, and acceptable factor loadings (>0.5) identified in the current analysis, which indicated sufficient factor relevance [1].

The decision to retain all factors reflects an integrative approach combining empirical patterns with theoretical rationale and previous AR-based validation studies [44]. This ensures the model remains aligned with experiential learning theory while maintaining conceptual coherence for future application in instructional design and further model development.

V. CONCLUSION AND FUTURE WORK

This study provides preliminary insights into the empirical validity of seven learnability factors—Motivation, Confidence, Enhanced Focus, Visualization of Invisible Concepts, Satisfaction, Better Lab Experience, and Better Learning—within the LEMARK–Hafsa framework, grounded in Kolb's Experiential Learning Theory. The analysis focused on initial measurement-level validation using exploratory factor analysis and regression techniques.

Exploratory factor analysis indicated acceptable internal consistency and a coherent unidimensional structure across all factors. Content validity was supported through prior expert review, and learner acceptance rates exceeded 80% for all factors in the final survey, reinforcing their practical and theoretical relevance. Notably, Visualization of Invisible Concepts showed a statistically significant positive association with Better Learning, while other factors demonstrated theoretically consistent but non-significant trends. While the findings of the study are encouraging, the modest regression results and the use of single-item measures suggest that the findings should be viewed as an exploratory step towards model validation rather than conclusive evidence of the framework's impact.

These findings offer preliminary support for the model's measurement foundation and confirm the readiness for further comprehensive analysis. Future research should focus on validating the full LEMARK–Hafsa model using multi-item scales and Partial Least Squares Structural Equation Modeling (PLS-SEM), across larger, more diverse cohorts to explore potential mediating effects of the learnability factors on Better Learning outcomes more rigorously.

ACKNOWLEDGMENT

This research work is funded and supported by the Faculty of Computing, Universiti Malaysia Pahang Al-Sultan Abdullah, Malaysia.

REFERENCES

[1] Hafsa, S., Majid, M.A. and Tawafak, R.M., 2021, August. Learnability factors of AR usage performance: Validating through survey. In 2021 International Conference on Software Engineering & Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM) (pp. 371-376). IEEE.:DOI: 10.1109/ICSECS52883.2021.00074.

[2] C. H. Lawshe, "A quantitative approach to content validity," Personnel Psychol., vol. 28, no. 4, pp. 563-575, 1975, doi: 10.1111/j.1744-6570.1975.tb01393.x.

[3] J. Bacca, S. Baldiris, R. Fabregat, and S. Graf, "Augmented reality trends in education: A systematic review of research and applications," J. Educ. Technol. Soc., vol. 17, no. 4, pp. 133-149, 2014.

[4] V. O. Adeyele, "Integrating augmented reality in preschool education: A systematic review," Int. J. Technol. Enhanced Learn., vol. 17, no. 3, pp. 265-284, 2025, doi: 10.1504/IJTEL.2025.147047.

[5] Ji, N. A. M. Mokmin, and J. Wang, "Evaluating the impact of augmented reality on visual communication design education: Enhancing student motivation, achievement, interest, and engagement," Educ. Inf. Technol., vol. 30, no. 5, pp. 6617-6639, 2025, doi: 10.1007/s10639-024-13050-x.

[6] P. Garcia-Robles, I. Cortes-Perez, F. A. Nieto-Escamez, H. Garcia-Lopez, and E. Obrero-Gaitan, "Immersive virtual reality and augmented reality in anatomy education: A systematic review and meta-analysis," Anat. Sci. Educ., vol. 17, no. 2, pp. 276-293, 2024, doi: 10.1002/ase.2397.

[7] S. U. Rehman, R. Khatoon, and I. A. Khan, "Unlocking the potential of AI-enhanced mixed reality: Enhancing technical English proficiency for engineering students in the Gulf region," Arab World English J., vol. 16, no. 4, pp. 185–195, Dec. 2025, doi: 10.24093/awej/vol16no4.10

[8] J. Ferrao, P. Dias, B. S. Santos, and M. Oliveira, "Environment-aware rendering and interaction in web-based augmented reality," J. Imaging, vol. 9, no. 3, art. 63, Mar. 2023, doi: 10.3390/jimaging9030063.

[9] V. J. F. Tan, "Enhancing teaching of augmented reality (AR) concepts through interactive WebAR technology," Proc. Int. Conf. Educ. Technol., 2025.

[10] G. Chen, H. Wang, A. Liang, M. Oubibi, and Y. Zhou, "From detached observer to immersive participant: An augmented reality-based experiential learning approach to promote academic performance and learning behaviors in science education," Comput. Human Behav. Rep., vol. 19, art. 100756, 2025, doi: 10.1016/j.chbr.2025.100756.

[11] A. P. K. Kwok et al., "Toward adopting the AR indoor navigation: An A. P. K. Kwok et al., "Toward adopting the AR indoor navigation: An acceptance study by integrating TAM with self-face concern and perceived enjoyment," Emerging Technol. Bus.: Innov. Strateg. Compet. Advantage, Singapore: Springer, 2024, pp. 47-70.

[12] C. C. Wu, P. C. Yeh, K. C. Huang, and H. Y. Hu, "Factors influencing seniors' adoption of augmented reality cognitive training games: An extended TAM with moderating effects of participants' attributes using a PLS-SEM approach," Educ. Gerontol., pp. 1-23, 2025.

[13] C. A. Chen and H. I. Lai, "Application of augmented reality in museums – Factors influencing the learning motivation and effectiveness," Sci. Prog., vol. 104, no. 3, art. 00368504211059045, 2021, doi: 10.1177/00368504211059045.

[14] A. B. Ustun, F. G. Karaoglan-Yilmaz, R. Yilmaz, M. Ceylan, and O. A. B. Ustun, F. G. Karaoglan-Yilmaz, R. Yilmaz, M. Ceylan, and O. Uzun, "Development of UTAUT-based augmented reality acceptance scale: A validity and reliability study," Educ. Inf. Technol., vol. 29, no. 9, pp. 11533-11554, 2024.

[15] N. Jusoh, M. W. Ali, T. A. T. Abdullah, and A. Husain, "Validation construct items for the measurement model of permit to work using exploratory factor analysis," Int. J. Bus. Globalisation, vol. 30, no. 3–4, pp. 462-477, 2022.

[16] M. H. Kotp et al., "Empowering nurse leaders: Readiness for AI integration and the perceived benefits of predictive analytics," BMC Nurs., vol. 24, art. 56, 2025, doi: 10.1186/s12912-024-02653-x.

[17] S. Hafsa and M. A. Majid, "Learnability factors for investigating the effectiveness of augmented reality smart glasses in smart campus," IOP Conf. Ser.: Mater. Sci. Eng., vol. 958, p. 012005, Oct. 2020, doi: 10.1088/1757-899X/958/1/012005.

[18] R R. Kaviyaraj and M. Uma, "An augmented reality framework for education: Deep learning integration and impact evaluation," IEEE Access, 2025.

[19] G. Koutromanos, A. T. Mikropoulos, D. Mavridis, and C. Christogiannis, "The mobile augmented reality acceptance model for teachers and future teachers," Educ. Inf. Technol., vol. 29, no. 7, pp. 7855-7893, 2024.

[20] B. Albishri and K. L. Blackmore, "Factors shaping instructors' augmented reality adoption: An extended UTAUT2 using a mixed-method approach," Proc. Int. Conf. Intell. Technol. Educ. Appl., Singapore: Springer, May 2025, pp. 252-271.

[21] C. Combrinck, "Not liking the Likert? A Rasch analysis of forced-choice format and usefulness in survey design," SAGE Open, vol. 14, no. 4, 2024, doi: 10.1177/21582440241295501.

[22] L. Surucu, I. Yikilmaz, and A. Maslakci, "Exploratory factor analysis (EFA) in quantitative researches and practical considerations," Gumushane Univ. Saglik Bilimleri Dergisi, vol. 13, no. 2, pp. 947-965, 2022, doi: 10.37989/gumussagbil.1183271.

[23] C. H. Lawshe, "A quantitative approach to content validity," Personnel Psychol., vol. 28, no. 4, pp. 563-575, 1975.

[24] J. F. Hair Jr., J. J. Risher, M. Sarstedt, and C. M. Ringle, "When to use and how to report the results of PLS-SEM," Eur. Bus. Rev., vol. 31, no. 1, pp. 2-24, 2019, doi: 10.1108/EBR-11-2018-0203.

[25] M. A. Memon, H. Ting, J. H. Cheah, R. Thurasamy, F. Chuah, and T. H. Cham, "Sample size for survey research: Review and recommendations," J. Appl. Struct. Equation Modeling, vol. 4, no. 2, pp. i–xx, 2020.

[26] O. Zimba and A. Y. Gasparyan, "Designing, conducting, and reporting survey studies: A primer for researchers," J. Korean Med. Sci., vol. 38, no. 48, 2023, doi: 10.3346/jkms.2023.38.e348.

[27] A. M. Sharifnia, D. E. Kpormegbey, D. K. Thapa, and M. Cleary, "A primer of data cleaning in quantitative research: Handling missing values and outliers," J. Adv. Nurs., early view, Mar. 2025, doi: 10.1111/jan.16908.

[28] N. Yusoff, A. Abd Razak, and N. M. Abbas, "Assessing the reliability N. Yusoff, A. Abd Razak, and N. M. Abbas, "Assessing the reliability and validity of a survey questionnaire for online laboratory courses in mechanical engineering programs," ASEAN J. Eng. Educ., vol. 8, no. 1, pp. 70-75, 2024.

[29] N. Shrestha, "Factor analysis as a tool for survey analysis," Amer. J. Appl. Math. Stat., vol. 9, no. 1, pp. 4-11, Jan. 2021, doi: 10.12691/ajams-9-1-2.

[30] M. H. Kotp et al., "Empowering nurse leaders: Readiness for AI integration and the perceived benefits of predictive analytics," BMC Nurs., vol. 24, p. 56, 2025, doi: 10.1186/s12912-024-02653-x.

[31] J. F. Hair Jr., W. C. Black, B. J. Babin, and R. E. Anderson, Multivariate Data Analysis, 6th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2009.

[32] E. Knekta, C. Runyon, and S. Eddy, "One size doesn't fit all: Using factor analysis to gather validity evidence when using surveys in your research," CBE–Life Sci. Educ., vol. 18, no. 1, art. rm1, Mar. 2019, doi: 10.1187/cbe.18-04-0064.

[33] M. Lukitasari, I. Listiani, S. Utami, I. Irwandi, A. Sukri, R. Hasan, and A. O. Cortez, "Development and validation of the STEM-DT instrument: A confirmatory factor analysis for assessing readiness STEM and design thinking competencies," TEM J., vol. 14, no. 1, pp. 836-847, Feb. 2025, doi: 10.18421/TEM14-74.

[34] A. Ahmed and W. Maruod, "Comparative analysis of Varimax and Promax rotation methods in exploratory factor analysis," Edelweiss Appl. Sci. Technol., vol. 9, no. 5, pp. 501-513, 2025.

[35] P. Rogers, "Best practices for your exploratory factor analysis: A factor tutorial," Rev. Adm. Contemp., vol. 26, no. 6, art. e210085, 2022.

[36] A. Diamantopoulos, M. Sarstedt, C. Fuchs, P. Wilczynski, and S. Kaiser, "Guidelines for choosing between multi-item and single-item scales for construct measurement: A predictive validity perspective," J. Acad. Marketing Sci., vol. 40, no. 3, pp. 434-449, May 2012, doi: 10.1007/s11747-011-0300-3.

[37] W. Di, Y. Nie, B. L. Chua, S. Chye, and T. Teo, "Developing a single-item general self-efficacy scale: An initial study," J. Psychoeduc. Assess., vol. 41, no. 5, pp. 583-598, Oct. 2023, doi: 10.1177/07342829231161884.

[38] A. Ahmed and W. Maruod, "Comparative analysis of Varimax and A. Ahmed and W. Maruod, "Comparative analysis of Varimax and Promax rotation methods in exploratory factor analysis," Edelweiss Appl. Sci. Technol., vol. 9, no. 5, pp. 501-513, 2025.

[39] Z. Zhang, T. Sangsawang, K. Vipahasna, and M. Pigultong, "A mixed-methods data approach integrating importance-performance analysis (IPA) and Kaiser-Meyer-Olkin (KMO) in applied talent cultivation," J. Appl. Data Sci., vol. 5, no. 1, pp. 256-267, 2024.

[40] D. D. Suhr, "Exploratory or confirmatory factor analysis?" Proc. SAS Users Group Int. Conf. (SUGI 31), San Francisco, CA, USA, 2006, paper 200–31.

[41] L. R. Fabrigar, D. T. Wegener, R. C. MacCallum, and E. J. Strahan, L. R. Fabrigar, D. T. Wegener, R. C. MacCallum, and E. J. Strahan, "Evaluating the use of exploratory factor analysis in psychological research," Psychol. Methods, vol. 4, no. 3, pp. 272-299, Sep. 1999, doi: 10.1037/1082-989X.4.3.272.

[42] J. F. Hair, W. C. Black, B. J. Babin, and R. E. Anderson, Multivariate Data Analysis, 8th ed. Boston, MA, USA: Cengage Learning, 2019.

[43] J. Cohen, Statistical Power Analysis for the Behavioral Sciences, 2nd ed. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1988.

[44] S. Hafsa and M. A. Majid, "Learnability factors for investigating the effectiveness of augmented reality smart glasses in smart campus," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 958, p. 012005, Feb. 2020, doi: 10.1088/1757-899X/958/1/012005.

[45] D. A. Kolb, Experiential Learning: Experience as the Source of Learning and Development. Englewood Cliffs, NJ, USA: Prentice-Hall, 1984.

Automating Computation Independent Model Elicitation in MDA using Task-Oriented Dialogue with In-Context Learning

Mohamed EL Ayadi, Yassine Rhazali, Mohammed Lahmer
ISIC Research Team, L2ISEI laboratory, Higher School of Technology, Moulay Ismail University, Meknes, Morocco

***Abstract*—The Computation Independent Model (CIM) is a cornerstone of the Object Management Group's (OMG) Model-Driven Architecture (MDA), capturing business requirements and domain knowledge independent of specific technologies. However, the elicitation of CIM requirements is often a manual, time-consuming, and error-prone process, susceptible to ambiguities inherent in natural language. Traditional Natural Language Understanding (NLU) approaches, particularly intent-based systems, exhibit limitations in scalability, contextual understanding, and handling the nuanced, evolving nature of complex requirements. This study proposes a novel approach that integrates Task-Oriented Dialogue (TOD) systems with the In-Context Learning (ICL) capabilities of Large Language Models (LLMs) to automate and enhance CIM requirements elicitation. The proposed framework features a conversational agent that guides stakeholders through structured dialogue flows, translating their natural language inputs into a formal CIM-Domain Specific Language (CIM-DSL). These DSL commands are then transformed into CIM artifacts, such as Business Process Model and Notation (BPMN) diagrams and Unified Modeling Language (UML) use cases. The approach emphasizes quality assurance through interactive validation, consistency checks, and strategies to mitigate LLM limitations. We anticipate this method will significantly improve the accuracy, completeness, and efficiency of CIM construction, thereby strengthening the foundation of the MDA lifecycle.**

Keywords*—MDA; CIM; Task-Oriented Dialogue (TOD); In-Context Learning (ICL); Large Language Models (LLMs); requirements elicitation; Domain-Specific Language (DSL); Artificial Intelligence (AI); Natural Language Understanding (NLU); BPMN

I. Introduction

The Model-Driven Architecture (MDA), an initiative by the Object Management Group (OMG), advocates for a software development paradigm centered on the systematic creation and transformation of models [1]. At the highest level of abstraction within MDA is the Computation Independent Model (CIM) [1]. The CIM articulates system requirements and domain knowledge from a purely business perspective, deliberately avoiding any technological implementation details [1]. It typically encompasses business rules, processes, organizational structures, stakeholder roles, and the overarching business requirements for the system [2]. The CIM serves as a critical communication bridge between domain experts and IT professionals, ensuring that the developed system aligns with business objectives [1]. The accuracy and completeness of the CIM are paramount, as they directly influence the subsequent Platform Independent Model (PIM), Platform Specific Model (PSM), and ultimately, the quality of the final software product [3]. Many research efforts have focused on the transformation from CIM to PIM [4].

Despite its significance, the elicitation of requirements for the CIM is fraught with challenges inherent in traditional requirements engineering [5].

Requirements elicitation, the process of gathering, defining, and refining system requirements from stakeholders [6], is widely recognized as a critical yet difficult, error-prone, and communication-intensive phase [7]. Conventional methods like interviews and document analysis often yield specifications that are ambiguous, inconsistent, and incomplete [7]. Natural language, the primary medium for these interactions, lacks formal semantics, exacerbating these issues. Consequently, manual and semi-manual CIM creation is not only laborious but also struggles to achieve the precision necessary for effective model transformation in MDA [8].

The emergence of advanced conversational Artificial Intelligence (AI), particularly systems leveraging Large Language Models (LLMs), offers a promising avenue to automate and enhance requirements elicitation [9]. This study proposes a novel approach that integrates Task-Oriented Dialogue (TOD) with In-Context Learning (ICL) to construct a conversational agent for CIM requirements elicitation. This approach aims to automate the capture of user specifications and facilitate their effective transformation into higher-level CIM artifacts, addressing the limitations of existing methods.

The central research problem addressed in this study is the inability of traditional, manual elicitation methods and intent-based NLU to handle the scalability, contextual nuances, and formal precision required for CIM construction. To resolve this, our research investigates the following key questions:

- RQ1: How can the integration of TOD frameworks and ICL-powered LLMs facilitate a guided, systematic elicitation of business processes and rules?
- RQ2: To what extent can the use of a CIM-Domain Specific Language (CIM-DSL) as an intermediate representation eliminate ambiguity during the translation of natural language to formal models?

- RQ3: How effectively can interactive validation and consistency checks within a conversational loop mitigate common LLM limitations like hallucinations and bias?

By addressing these questions, this study contributes a structured architectural framework that automates the generation of CIM artifacts—such as BPMN and UML diagrams—directly from stakeholder dialogue, thereby increasing productivity and ensuring verifiable traceability within the MDA lifecycle.

II. Background and Motivation

A. Limitations of Intent-Based NLU for CIM Requirements Elicitation

Traditional Natural Language Understanding (NLU) systems, especially those reliant on intent classification and slot filling, present several inherent limitations when applied to the complex task of CIM requirements elicitation.

A primary drawback is their poor scalability and high maintenance overhead [10]. These systems depend on a predefined catalogue of intents and entities. AS domain complexity increases, which is often the case in business requirements, the number of user intents can grow exponentially [11]. Each intent typically requires manual definition [12], a substantial number of training examples, and in many cases, complete retraining of the model to preserve accuracy [13]. This renders the system cumbersome to adapt to new or evolving requirements, a common scenario in early-stage software development.

Furthermore, intent-based NLU systems often exhibit limited contextual understanding [7]. They tend to process utterances in isolation or with a very narrow conversational window, struggling with nuances, cross-turn coreferences, or implicit information [7]. The rich, evolving context of a requirements dialogue, where understanding is built incrementally, is poorly supported. This directly impacts their ability to handle nuanced or evolving requirements. CIM elicitation is inherently exploratory; requirements are often initially vague and refined through dialogue [14]. Predefined intents cannot easily accommodate this fluidity [7].

The data demands for training robust intent-based NLU are also substantial, requiring numerous diverse examples per intent [10]. For the domain-specific requirements of a CIM, curating such datasets is often impractical. Lastly, these systems possess limited generative capabilities [15]; they are ill-suited for collaborative refinement, proposing structured interpretations, or generating descriptive content for CIM elements. These limitations collectively hinder the effective and accurate generation of a CIM, which demands a more flexible, context-aware, and adaptable approach [4].

B. Limitations of Current LLM-Based Requirements Engineering

While recent studies have explored the application of Large Language Models (LLMs) to requirements engineering, existing approaches often lack the structural rigor and architectural alignment required for Model-Driven Architecture (MDA). Current systems generally fall into three categories, each presenting distinct limitations that our proposed framework specifically addresses.

First, Intent-Based NLU systems rely heavily on slot-filling and intent classification mechanisms. While effective in narrow domains, they suffer from poor scalability because every new user intent requires manual definition and massive training datasets. To overcome this, our framework employs In-Context Learning (ICL), which allows the system to adapt to new domains via few-shot examples without the need for extensive retraining.

Second, approaches utilizing Direct LLM Prompting (e.g., zero-shot or chain-of-thought) offer flexibility but often produce unstructured outputs. These systems carry a high risk of "hallucination" and lack state consistency, frequently generating requirements that are syntactically correct but semantically invalid within the CIM context. The proposed TOD-ICL framework mitigates this by enforcing a Dialogue Stack and CIM-DSL, which constrain the LLM's output to a formal structure and track the elicitation progress via a deterministic state machine.

Finally, Unstructured Chatbot tools typically function as free-text conversationalists. While engaging, they introduce significant ambiguity when translating natural language into formal models like BPMN or UML. Our approach resolves this by introducing a Formal Intermediate Layer, which translates loose dialogue into unambiguous CIM-DSL commands before any modeling artifacts are generated.

C. Task-Oriented Dialogue and In-Context Learning: A New Paradigm

To address the shortcomings of intent-based NLU, we turn to Task-Oriented Dialogue (TOD) systems enhanced by the In-Context Learning (ICL) capabilities of LLMs.

Task-Oriented Dialogue (TOD) systems are conversational agents designed to assist users in achieving specific goals through structured interactions [16]. They interpret user intent, manage dialogue state, and execute predefined plans to fulfil user objectives [17]. This goal-driven paradigm is well-suited for systematic information gathering, such as eliciting the business processes, roles, and rules that constitute a CIM [17].

In-Context Learning (ICL) is a powerful technique enabling pre-trained LLMs to adapt to new tasks by being presented with a few demonstrations (few-shot examples) directly within the input prompt, without modifying the LLM's weights [18]. ICL allows LLMs to discern the structure, style, and expected output format from these contextual examples [19], significantly reducing the need for extensive task-specific training data [20]. This adaptability is crucial for guiding LLMs to understand and structure diverse, domain-specific inputs during CIM elicitation.

The work by Bocklisch et al. [10] provides a foundational methodology for advanced TOD systems that aligns with our proposed approach. Key contributions include the transition from intent classification to generating commands in a Domain-Specific Language (DSL) using an LLM-based Dialogue Understanding (DU) module [10]. This DU module considers the full conversation history and declarative "flows" (defining task logic) for nuanced interpretation [10]. This approach

handles complex dialogue phenomena more effectively and with reduced development effort compared to traditional NLU [10].

The integration of TOD systems, which provide structured interaction frameworks and robust state management, with ICL-LLMs, which offer flexible and context-sensitive language understanding and generation, enables the development of systems capable of co-constructing requirements in dynamic and user-centered environments. This combination capitalizes on the procedural rigor and goal-tracking capabilities of TOD systems [21], while simultaneously leveraging the adaptability and few-shot learning potential of ICL-LLMs [19]. The resulting synergy fosters more effective and scalable collaboration between human users and AI agents in complex requirement elicitation scenarios.

This TOD-ICL combination offers superior adaptability, contextual coherence, structured output generation, and reduced data dependency compared to intent-based NLU, making it a more suitable foundation for CIM requirements elicitation [22].

III. The Proposed TOD-ICL Framework for CIM Elicitation

We propose a framework integrating TOD principles with ICL-powered LLMs to create a conversational agent for eliciting and structuring CIM requirements, in line with MDA principles [1].

A. Architectural Blueprint

The conversational agent comprises several core components:

1) Dialogue Management (DM): Orchestrates the conversation based on TOD principles, managing dialogue state, history, and interaction flows. These flows, inspired by Bocklisch et al. [10], define elicitation strategies for specific CIM information types, such as business processes modeled using BPMN. A dialogue stack manages complex interactions, sub-dialogues, and context switching [10].

2) LLM-based Natural Language Understanding (NLU) & DSL Generation: This core module uses a pre-trained LLM leveraging In-Context Learning (ICL) [19] to interpret user utterances. Critically, it shifts the NLU task from traditional intent classification to structure-to-structure generation, treating the user's input as a prompt for code synthesis [Table I] [23]. The LLM translates the natural language directly into commands in a formal CIM-Domain Specific Language (CIM-DSL). The input prompt is carefully constructed to include conversation history, the active flow state from the Dialogue Management module, and essential few-shot examples that map expected natural language inputs to their corresponding CIM-DSL commands. This approach avoids the need for extensive fine-tuning and provides an unambiguous intermediate representation, allowing the generated CIM-DSL commands to directly manipulate the dialogue state and initiate subsequent CIM construction [24].

3) CIM-Domain Specific Language (CIM-DSL): A formal, structured language designed to capture CIM concepts (e.g., business processes, actors, rules, goals) [2]. Example commands: DEFINE_BUSINESS_PROCESS (name="Order Fulfilment"), ADD_ACTOR (name="Customer"). This DSL acts as the unambiguous intermediate representation.

4) Domain knowledge module: A repository of CIM metamodel information [2], business process patterns, and domain ontologies [22]. It aids in validating CIM-DSL commands and providing context to the LLM.

5) Response generation module: Likely LLM-based, generates natural language responses, clarifications, and prompts, guided by the DM.

Stakeholders interact with the agent in natural language. The agent, guided by dialogue flows, elicits CIM-pertinent information, translating inputs into CIM-DSL commands for subsequent CIM construction [25].

B. Dialogue Flow and CIM-DSL Design for Elicitation

Effective CIM elicitation relies on well-designed dialogue flows and an expressive CIM-DSL.

1) Structuring dialogues with flows: Dialogue flows mirror CIM information architecture, guiding the conversation to systematically cover aspects like business process definition (often using BPMN [25]), actor/role identification, and business rule elicitation [17]. These flows are flexible guides, not rigid scripts [10].

2) ICL for CIM-DSL command generation: LLMs are prompted with few-shot examples demonstrating the mapping of natural language to CIM-DSL commands within specific contexts [18]. For instance, "Warehouse staff update inventory levels" might map to DEFINE_FUNCTIONAL_REQUIREMENT (description="Staff update inventory", actor="Warehouse Staff"). The prompt includes history, flow context, and examples [10].

3) CIM-Domain Specific Language (CIM-DSL): Designed to formally capture CIM semantics, this DSL serves as an unambiguous bridge between natural language inputs and formal MDA artifacts. To ensure computational actionability, we define its syntax using Extended Backus-Naur Form (EBNF) as ⟨Command⟩ = ⟨Action⟩ "(" ⟨ParameterList⟩ ")", where permissible actions include DEFINE_PROCESS, ADD_ACTOR, ADD_STEP, and SPECIFY_RULE. To maintain model integrity, the Domain Knowledge Module enforces strict semantic constraints such as referential integrity—ensuring an ADD_STEP command cannot reference a process that has not been initialized—and actor existence, which requires all actors to be pre-declared. These DSL commands map directly to standard CIM notations: DEFINE_PROCESS instantiates a BPMN Collaboration (Pool), ADD_ACTOR generates a UML Actor or BPMN Lane, and ADD_STEP creates a BPMN Task connected via Sequence Flows, ensuring the final output is a valid, sound-by-construction model suitable for downstream PIM transformation.

C. Transformation of CIM-DSL to CIM Elements

A dedicated transformation module parses validated CIM-DSL commands and maps them to formal CIM artifacts, a key aspect of MDA [Table II] [3].

1) Conversion methodology: This module uses predefined rules to interpret CIM-DSL semantics. For example, DEFINE_BUSINESS_PROCESS(...) could generate a BPMN diagram sketch or structured text. ADD_ACTOR(...) and ADD_USE_CASE(...) could generate UML Use Case elements [4]. The transformation from CIM to PIM often involves generating UML models [4] and can be automated using languages like ATL (Atlas Transformation Language) [8]. The use of an intermediate sound-by-construction representation like POWL for process models before BPMN rendering is a relevant concept [29].

2) Output formats for CIM: Outputs may include structured textual descriptions, UML Use Case diagrams, BPMN diagrams [29], and formalized lists of rules and goals [17]. Service-oriented models like SoaML or web-based models using IFML can also be targets at the PIM level [9].

TABLE I. ILLUSTRATIVE MAPPINGS: USER UTTERANCES TO CIM-DSL COMMANDS

Example User Utterance (Natural Language)	Context/Dialogue Flow	Generated CIM-DSL Command (Illustrative)
"Customers should be able to place orders online."	Identifying Key System Goals	DEFINE_SYSTEM_GOAL(description="Customers can place orders online", priority="High")
"Order processing involves checking inventory, then payment."	Defining "Order Processing"	ADD_PROCESS_STEP(process_name="Order Processing", step_name="Check Inventory", sequence_after="Order Received") ADD_PROCESS_STEP(process_name="Order Processing", step_name="Verify Payment", sequence_after="Check Inventory")
"Managers approve discounts over 20%."	Eliciting Business Rules	SPECIFY_BUSINESS_RULE(rule_text="Discounts > 20% require manager approval", scope="Order Discounting")

TABLE II. MAPPING OF CIM-DSL COMMANDS TO CIM ELEMENTS AND GENERATED ARTIFACTS

CIM-DSL Command(s)	Target CIM Element Type	Generated CIM Artifact Snippet/Description (Illustrative)
DEFINE_BUSINESS_PROCESS(name="PatientCheck-in") ADD_PROCESS_STEP(process_name="PatientCheck-in", step_name="Verify Appointment")	Business Process	BPMN Snippet: Pool: "Clinic"; Lane: "Receptionist"; Task: "Verify Appointment". Textual Description: Process: Patient Check-in.
ADD_ACTOR(name="Pharmacist") ADD_USE_CASE(name="Dispense Medication", actor="Pharmacist")	Actor, Use Case	UML Use Case Diagram Element: Actor: "Pharmacist". Use Case: "Dispense Medication".
SPECIFY_BUSINESS_RULE(rule_text="Controlled substances require two signatures.")	Business Rule	Business Rule List Entry: ID: BR001; Description: "Controlled substances require two signatures."

D. Quality Assurance and Validation

Ensuring a high-quality CIM involves several mechanisms:

1) Interactive validation and refinement: A human-in-the-loop approach is central. The agent uses confirmation loops ("So, to confirm... ") [10], clarification dialogues for ambiguities, and allows easy corrections.

2) Consistency checks: The Domain Knowledge Module performs internal consistency checks. Integrating domain ontologies at the CIM level can enhance consistency and validation [22]. Conformity to existing domain models can be checked, inspired by approaches like Topological Functioning Modeling (TFM).

3) Mitigating LLM limitations: Strategies address LLM issues like hallucination and bias:

4) Advanced prompt engineering: High-quality, diverse few-shot examples and clear instructions are critical [18].

5) Trial-Error-Explain In-Context Learning (TICL) adaptation: Inspired by TICL, if the LLM generates flawed CIM-DSL, these "negative samples" and explanations of their errors are added to the ICL prompt for iterative refinement.

6) Retrieval Augmented Generation (RAG): If a knowledge base of CIM examples or patterns exists, RAG can ground LLM generation in verified information.

7) Output validation: Generated artifacts are evaluated against requirements quality criteria (e.g., the "6Cs": clarity, completeness, conciseness, consistency, correctness, context).

8) Traceability: Clear links are maintained from stakeholder utterances to CIM-DSL commands and to final CIM elements, crucial for validation and change management [1]. Tools can support this traceability from CIM to code [25].

IV. ANTICIPATED CONTRIBUTIONS

This TOD-ICL approach is expected to offer several contributions:

- Automation of CIM Elicitation: Significantly reducing manual effort in gathering, structuring, and validating CIM requirements [9]. This aligns with MDA's goal of increasing productivity [1].
- Improved Accuracy and Completeness: Systematic TOD flows and advanced LLM understanding should lead to more comprehensive requirements capture [10].
- Enhanced Consistency: Automated generation of CIM-DSL and subsequent transformation can enforce greater consistency in CIM representation.
- Faster Development Cycles: Accelerating CIM construction leads to quicker iterations and more agile model-driven development [26].

- Improved Stakeholder Collaboration: An intuitive conversational interface can enhance engagement from business stakeholders [27].
- Verifiable Traceability: Direct links from utterances to CIM elements support validation and change management [1].
- Novel Application of Conversational AI: Applying TOD-ICL to CIM development advances AI-assisted software modeling.

The core contribution is a guided, interactive, and formalizing elicitation process, where the agent acts as an intelligent collaborator.

V. Experimental Evaluation and Results

To assess the feasibility, accuracy, and efficiency of the proposed TOD-ICL framework, we conducted a structured experimental evaluation. This section details the experimental setup, the datasets used, the defined evaluation metrics, and the quantitative results obtained.

A. Experimental Setup and Reproducibility

The experiments were conducted using the GPT-4 model accessed via API to simulate the role of the dialogue agent. To ensure reproducibility and minimize stochastic variations, the following hyperparameters were strictly enforced:

- Temperature: Set to 0.0 to maximize deterministic outputs.
- Context Window: Managed using a sliding window retention of the last 10 interaction turns to balance memory with token limits.
- System Prompting: The model was initialized with the Domain Knowledge Module instructions defined in Section III, augmented with 3 few-shot examples from the "Banking" domain to prime the In-Context Learning mechanism without leaking test data.

B. Dataset and Test Scenarios

Due to the scarcity of public datasets for CIM elicitation, we constructed a Synthetic Requirements Dataset consisting of three distinct domain scenarios, each representing different structural complexities:

- Order Fulfilment (E-Commerce): Focuses on sequential process flows and multi-actor coordination.
- Patient Registration (Healthcare): Tests strict data privacy constraints and conditional process branching.
- Loan Approval (Finance): Evaluates complex business rule specifications and logical constraints. Each scenario consists of a simulated dialogue transcript (approximately 15 turns each) and a corresponding "Ground Truth" CIM-DSL script manually validated by domain experts.

C. Evaluation Metrics

To provide an objective assessment of the generated artifacts, we defined three key performance metrics:

- Syntactic Validity Rate (SVR): The percentage of generated CIM-DSL commands that successfully parse against the EBNF grammar defined in Section III, without syntax errors.
- Entity Recall: The ratio of required CIM elements (Actors and Process Steps) successfully identified by the system compared to the Ground Truth.
- Rule Fidelity: A qualitative binary metric (Pass/Fail) assessing whether generated SPECIFY_RULE commands accurately capture the logical constraints (e.g., "Amount > $500") specified in the natural language input.

D. Quantitative Results and Analysis

The performance of the framework was analyzed across the three test scenarios, demonstrating high structural robustness and semantic accuracy.

1) Syntactic accuracy: The framework achieved an average Syntactic Validity Rate (SVR) of 98%. In the "Order Fulfilment" scenario, the system achieved a perfect 100% validity score. The "Patient Registration" and "Loan Approval" scenarios achieved 96% and 98%, respectively. The minor errors observed were primarily delimiter mismatches in nested parameters, which were automatically corrected by the post-processing RegEx module.

2) Completeness and entity extraction: Regarding content coverage, the system achieved an average Entity Recall of 95%. Specifically, the "Order Fulfilment" scenario saw 100% recall for both actors and process steps. The "Patient Registration" scenario showed a slight decrease to 90% recall, largely due to difficulties in identifying implicit actors (e.g., "System Database"). However, the framework successfully identified 100% of explicit actors (e.g., "Clerk," "Manager") across all domains.

3) Ambiguity resolution and efficiency: To measure the efficiency of the dialogue flow, we tracked the number of turns required to resolve ambiguities. The system required an average of 1.3 additional turns to clarify vague requirements. "Patient Registration" required the most clarification (2 turns), while the other scenarios required only 1 turn. The average generation latency was recorded at approximately 1.2 seconds per turn, confirming the system's viability for real-time interaction.

4) Scalability: To test scalability, the dialogue history was artificially extended to 20 turns. The system maintained context fidelity with no significant degradation in entity extraction accuracy, validating that the sliding window approach effectively manages the Context Window limitations discussed in previous sections.

VI. Challenges and Future Research Directions

The TOD-ICL framework faces several multifaceted challenges that necessitate a rigorous future research agenda. Primarily, the system must overcome semantic indeterminacy and linguistic ambiguity through advanced disambiguation protocols and the integration of formal reasoning. As enterprise architectural complexity increases, the framework encounters

scalability constraints regarding LLM context windows and the orchestration of high-density, interconnected CIM elements, suggesting a transition toward modular dialogue flows and hierarchical CIM-DSL structures. Ensuring the formal verification of generated artifacts is equally critical, requiring the co-generation of formal properties and integration with automated model checkers to move beyond basic validation. Furthermore, the evolutionary design of the CIM-DSL must balance expressivity with manageability via meta-modeling, while simultaneously addressing LLM stochasticity—specifically hallucinations and bias—through hybrid AI paradigms and specialized fine-tuning. Seamless interoperability with legacy MDA toolchains must be secured through the standardization of CIM-DSL and XMI-compliant transformations [3]. Finally, the establishment of comprehensive evaluation metrics that synthesize task-success benchmarks with user-centered interaction data is vital to fostering a synergistic human-AI collaborative loop, effectively positioning the LLM as a sophisticated co-evolutionary assistant rather than a fully autonomous black box.

VII. Conclusion

The proposed integration of Task-Oriented Dialogue with In-Context Learning offers a promising pathway to address the enduring challenges in Computation Independent Model requirements elicitation within the Model-Driven Architecture. By leveraging the strengths of structured dialogue and the adaptability of Large Language Models, this approach aims to automate the capture of business knowledge [4], translate it into a formal intermediate representation (CIM-DSL), and subsequently generate core CIM artifacts [28]. This method has the potential to significantly enhance the accuracy, completeness, consistency, and efficiency of CIM development, thereby providing a more robust foundation for the entire MDA lifecycle [1]. While challenges related to natural language ambiguity, LLM reliability, scalability, and formal verification persist, they also delineate clear avenues for future research. Ultimately, this work seeks to advance the state of AI-assisted software engineering by fostering a more effective and formalized dialogue between business stakeholders and the modeling process.

References

[1] OMG-MDA. (2014). MDA Guide version 2.0 (Rep. ormsc/14-06-01). OMG.

[2] OMG. (2003). MDA Guide revision 1.0.1 (Rep. omg/03-06-01). OMG.

[3] Kleppe, A. G., Warmer, J., & Bast, W. (2003). MDA explained: The model driven architecture: Practice and promise. Addison-Wesley.

[4] Rhazali, Y., Hadi, Y., & Mouloudi, A. (2015). A methodology for transforming CIM to PIM through UML: From business view to information system view. In 2015 Third World Conference on Complex Systems (WCCS) (pp. 1–6). Marrakech, Morocco.

[5] Kuštelega, M., Karan, M., & Magdalenić, I. (2022). A Systematic Analysis of Requirements Elicitation Problems in Software Development Projects. In: 2022 33rd Central European Conference on Information and Intelligent Systems (CECIIS), pp. 215–224.

[6] Zowghi, D., & Coulin, C. (2005). Requirements elicitation: A survey of techniques, approaches, and tools. In Engineering and managing software requirements (pp. 19–46). Springer Berlin Heidelberg.

[7] Bhandari, N. (2023). An interactive chatbot for software requirements elicitation (Thesis). Leiden Institute of Advanced Computer Science (LIACS), Leiden University, Leiden, The Netherlands.

[8] Rhazali, Y., Hadi, Y., & Mouloudi, A. (2016). Model transformation with ATL into MDA from CIM to PIM structured through MVC. Procedia Computer Science, 83, 1096–1101.

[9] Hemmat, A., Sharbaf, M., Kolahdouz-Rahimi, S., Lano, K., & Tehrani, S. Y. (2025). Research directions for using LLM in software requirement engineering: A systematic review. Frontiers in Computer Science, 7, 1519437.

[10] Bocklisch, T., Werkmeister, T., Varshneya, D., & Nichol, A. (2024). Task-oriented dialogue with in-context learning. arXiv. https://arxiv.org/abs/2402.12234

[11] Braun, D., Hernandez-Mendez, A., Matthes, F., & Langen, M. (2017). Evaluating natural language understanding services for conversational question answering systems. In Proc. 18th Annu. SIGdial Meeting Discourse Dialogue (pp. 174–185).

[12] Kim, Y., Kim, D., & Kim, G. (2021). Intent detection and slot filling for goal-oriented dialogue systems with BERT-based transfer learning. Information Sciences, 546, 114–125.

[13] Zhang, C., Liu, Q., Huang, M., & Zhu, X. (2020). Multi-turn intent detection via token-level dynamic fusion. In Proc. 58th Annu. Meeting of the Association for Computational Linguistics (ACL 2020) (pp. 5464–5473).

[14] Görgen, L., & Schimmler, M. (2024). Towards dialogue-based, computer-aided software requirements elicitation. arXiv preprint. arXiv:2310.13953.

[15] Kessabi, L., et al. (2024). Large language models for model-driven engineering: A systematic literature review and expert survey. arXiv.https://arxiv.org/abs/2403.10078 (Accessed May 20, 2025).

[16] Zhang, Z., Takanobu, R., Zhu, Q., Huang, M., & Zhu, X. (2020). Recent advances and challenges in task-oriented dialogue systems. In Proc. 58th Ann. Meeting Assoc. Comput. Linguistics (ACL 2020).

[17] Lee, H., & Jeong, O. (2023). A knowledge-grounded task-oriented dialogue system with a hierarchical structure for enhancing knowledge selection. Sensors, 23, 685. https://doi.org/10.3390/s23020685

[18] Dong, Q., et al. (2023). A survey on in-context learning. In Proc. 2023 Conf. Empirical Methods Natural Lang. Process. (EMNLP) (pp. 1107–1128).

[19] Brown, T. B., et al. (2020). Language models are few-shot learners. In Advances in Neural Information Processing Systems (Vol. 33, pp. 1877–1901).

[20] Xie, S. M., Raghunathan, A., Liu, F., et al. (2022). An explanation of in-context learning as implicit Bayesian inference. In International Conference on Learning Representations (ICLR).

[21] Zhao, J., Gupta, R., Cao, Y., et al. (2022). Description-driven task-oriented dialog modeling. arXiv preprint. arXiv:2201.08904.

[22] Laaz, N., Wakil, K., Gotti, Z., Gotti, S., & Mbarki, S. (2019). Integrating domain ontologies in an MDA based development process of e-health management systems at the CIM level. In Advanced Intelligent Systems for Sustainable Development (AI2SD'2019) Volume 2 (pp. 213–223). Springer International Publishing.

[23] Chen, M., Tworek, T., Jun, H., Yuan, Q., Sorensen, H., Schuster, S., et al. (2021). Evaluating large language models trained on code. arXiv preprint. arXiv:2107.03374.

[24] Kulkarni, V., & Clark, T. (2014). Model driven development of business applications: A practitioner's perspective. Journal of Software: Evolution and Process, 26(6), 613–638.

[25] Kharmoum, N., Ziti, S., Rhazali, Y., & Omary, F. (2019). A method of model transformation in MDA approach from e3value model to BPMN2 diagrams in CIM level. IAENG International Journal of Computer Science, 46(4), 1–17.

[26] Essebaa, I., Chantit, S., & Ramdani, M. (2019). MoDAr-WA: Tool support to automate an MDA approach for MVC web application. Computers, 8(4), 89.

[27] Syvänen, S., Heinonen, K., & Salonen, A. (2020). Conversational agents in online organization–stakeholder interactions: A state-of-the-art analysis and implications for further research. Corporate Communications: An International Journal, 25(4), 613–631.

[28] Melouk, M., Rhazali, Y., & Hadi, Y. (2020). An approach for transforming CIM to PIM up to PSM in MDA. Procedia Computer Science, 170, 869–874.

[29] Kourani, H., Berti, A., Schuster, D., & van der Aalst, W. M. P. (2024). Process modeling with large language models. arXiv. https://arxiv.org/abs/2403.07541

Mapping Research on Artificial Intelligence in Customer Experience: A Bibliometric Analysis

Firdaws Hayoun[1]*, Brahim Ouabouch[2], Youssef Aatif[3], Taoufiq Yahyaoui[4], Fatima Zahra El Arbaoui[5]
Research Laboratory in Economics and Management of Organizations (LAREMO),
National School of Business and Management, Sultan Moulay Slimane University, Beni-Mellal, Morocco[1, 2, 4, 5]
Laboratory of Public Economics and Political Science, Faculty of Economics and Management,
Ibn Tofail University, Kenitra, Morocco[3]

***Abstract*—The purpose of this article is to conduct a comprehensive bibliometric analysis of research on artificial intelligence and customer experience. The study data was extracted from Scopus and Web of Science, focusing on articles published between 2010 and 2025. VOSviewer and Biblioshiny software were used to map the intellectual landscape of the interaction between artificial intelligence and customer experience, identifying growth in scientific output, geographical distribution and collaboration, influential publications, leading authors, word co-occurrence, leading journals, and thematic trends. The study reveals that research only began in 2017, then interest in this topic began to grow, but most publications date from 2025. Furthermore, the findings reveal that research is concentrated in a limited number of leading countries which are more advanced in terms of technological infrastructures. This field of research is gradually evolving towards greater specialization and increased use of AI to improve customer experience in terms of personalization, decision-making and automation of service.**

***Keywords*—Artificial intelligence; customer experience; marketing; personalization**

I. Introduction

Artificial intelligence, one of the most recent general-purpose technologies (GPTs), which are still in development [1], is defined as the system's ability to interpret external data correctly, to learn from such data, and to use those learnings to achieve specific goals and tasks through flexible adaptation [2]. In the last few years, artificial intelligence has gained considerable attention in the business field, with the adoption of AI in at least one business function increasing from 78% in 2024 to 88% in 2025 [3].

At the same time, customer experience has become a central concept in AI human interaction research and a strategic priority in business. Companies now compete on the quality of their interactions with customers, not just on products or prices.

Conducting a bibliometric analysis allows us to observe the patterns and trends discussed by researchers regarding artificial intelligence in customer experience. It serves as a valuable guide for researchers and practitioners by highlighting the main topics covered, the most prominent researchers in the field, recent trends and advances in this area, and helps to better understand potential solutions and recommendations. The objective of this article is to map research on artificial intelligence and customer experience in order to explore and analyze their commonalities by analyzing peer-reviewed articles (review articles and research articles) published between 2010 and 2025. These articles are extracted from two databases, Scopus and Web of Science, and then analyzed and mapped using VOSviewer and Biblioshiny software. To achieve this objective, this study will answer the following research questions:

- RQ1: How has the volume of publications in Artificial Intelligence and Customer Experience evolved?
- RQ2: How is the scientific production distributed geographically, and which countries are leading the collaboration?
- RQ 3: Which core journals and prolific authors constitute the primary intellectual pillars of the AI-CX research domain?
- RQ4: How is the research output distributed geographically, and which countries exhibit the highest citation impact regarding AI applications in customer experience?
- RQ5: What are the dominant thematic clusters that define the current AI-CX landscape?

After presenting the objective of this article, the rest will be structured as follows: the methodological section, in which the research method is explained and the data collection process, research strategy, data refinement, data scanning, and data analysis are detailed. This is followed by the results section, which presents the main findings of the bibliometric analysis, including the evolution of scientific output over the years, country productivity and collaboration patterns, relevant journals, co-citation networks, word co-occurrence, and keyword thematic structures. Next, the "Discussion" section interprets these results in relation to the existing literature and mentions their implications for AI in the customer experience. Finally, the "Conclusion" section summarizes the main contributions of the study, highlights its limitations, and suggests avenues for future research.

II. Methodology

Bibliometric analysis is a research methodology that maps the progression and trends of a research topic, identifies key research trends in recent years, and provides insight into the articles, journals, and institutions that are at the heart of a given theme or research area. It also uses influence networks, literature

clusters, and keyword analysis to better position a gap in research. To conduct this analysis, a four-step approach is followed, beginning with (A) a search strategy using Scopus and Web of Science Boolean operators and all keywords referring to artificial intelligence and customer experience. In order to obtain relevant data, several keywords were used. For customer experience, we used the terms "consumer experience," "customer experience," "user experience," "UX," "customer journey," "customer service," "experience management," and "customer touchpoint." For artificial intelligence, we used the terms "AI," "machine learning," "deep learning," and "generative AI." Next, (B) we refined the data by selecting only articles published between 2010 and 2025. Then, (C) we scanned the titles and abstracts of the articles to eliminate those that were not relevant, as some articles may contain the keywords we were looking for without contributing to the topic. From there, we extracted the data to (D) analyze it using VOSviewer software, a computer program that builds maps based on co-occurrence data and visualizes these maps [4]. And the bibliometrix package of the R software via its web interface, Biblioshiny [5].

A. Search Strategy

To ensure full coverage of scientific production about artificial intelligence and customer experience, bibliographic data were extracted from the most acknowledged academic databases Scopus and Web of Science, which are renowned for their extensive coverage and the quality of their scientific publications. The objective was to collect a representative set of relevant scientific documents on artificial intelligence and customer experience, two key concepts in current research. To do that, a structured keyword-based search strategy was developed to have Scopus search query and Web of Science search query (Fig. 1). A series of keywords associated with artificial intelligence and customer experience were selected. Scopus and Web of Science's advanced search Boolean operators were then used to combine these terms to refine the results.

B. Data Refining

The documents identified during the search were filtered using the "refine search" section in Scopus and Web of Science. This process is designed to exclude all documents that do not meet the inclusion criteria. As a result, the selected data was reduced from a total of 665 to 495 articles.

C. Data Scanning

In order to ensure the relevance of the data, 495 articles were reviewed by reading the title and abstract of each one. Ultimately, 325 articles were eliminated from the list because they did not correspond to the subject, and the bibliometric analysis will focus on a set of 170 articles.

D. Data Analysis

For better analysis we used the software VOSviewer to visualize all networks of the data and the bibliometrix package of the R software via its web interface, Biblioshiny, to map the intellectual landscape of the research.

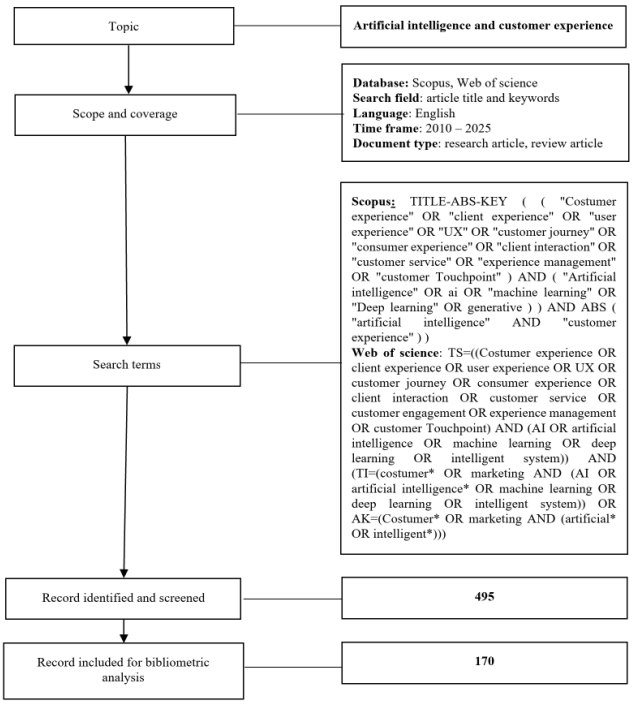


Fig. 1. Flow diagram of the search strategy.

III. Results

A. Publication Growth Over the Study Period

The search for articles on artificial intelligence and customer experience yielded 495 review and research articles, but after the selection process, we found that only 170 articles established a link between AI and customer experience. This does not mean that there were no publications in this field, but that previous publications dealt with AI and customer experience separately. Fig. 2 answers the first research question; it shows that the evolution of publications can be divided into 4 stages:

- Stage 1 (2017-2018): The research was marginal, with only 1 and 2 articles, indicating an emerging phase for the topic.
- Stage 2 (2019-2021): an exponential rise in activity is observed, and the number of publications is getting a bit higher.
- Stage 3 (2022-2023): The topic enters a period of remarkable growth, growing from 9 publications in 2021 to 15 in 2022, a level that will remain throughout 2023.
- Stage 4 (2024-2025): The most significant change happens between 2024 and 2025, when the number of publications increases exponentially from 33 to 88 publications. This shows that AI in customer experience is becoming a trending topic and gaining researchers' attention.

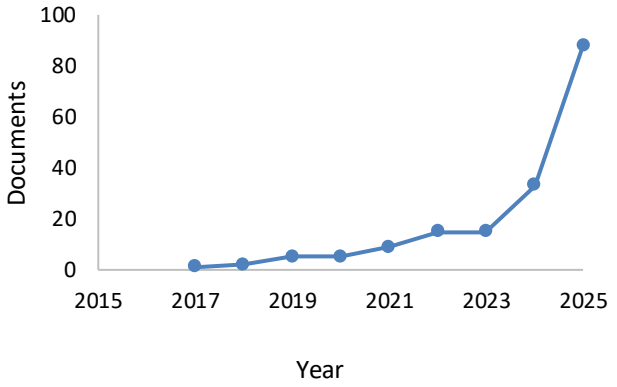


Fig. 2. Yearly distribution of publications.

The research focused on the period 2010-2025 so as not to overlook any research, as AI began to be discussed in the 1960s and its application to the business world later. But the intersection between AI and customer experience began to emerge mainly from 2017 onwards, reflecting its recent emergence.

B. Countries' Scientific Production and International Collaborations

To examine the geographical distribution of publications in the field of artificial intelligence and customer experience, the institutional affiliations of authors were analyzed using RStudio software "Biblioshiny", which provides an overview of the distribution of research efforts in this field. Fig. 3 shows that scientific production is visually encoded based on the blue color gradient, with variations in color intensity reflecting differences in publication volume. Countries shaded in darker blue represent more scientific productivity, whereas those displayed in lighter blue have relatively lower levels of research production.

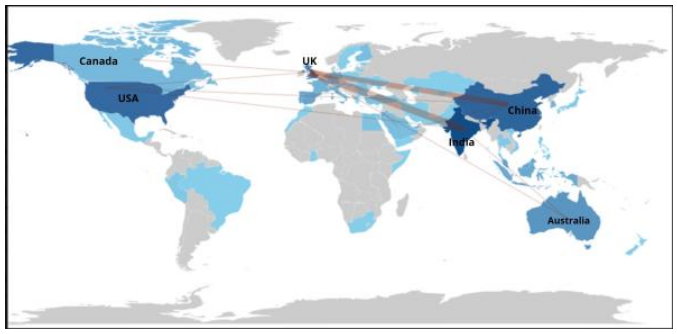


Fig. 3. Countries' scientific production and international collaboration.

The first group of countries, represented by a darker shade of blue, shows significant contributions; India leads with a total of 56 publications, followed by China with 52 publications, the United States with a total of 47 publications, Portugal with 40 publications, and the United Kingdom with 33 publications. Next comes the second group of countries, represented by a medium shade of blue, which shows a steady contribution to research, but with less intensity than the first group. It includes Australia (25), Spain (18), Indonesia (15), Malaysia (15), France (12), Jordan (11), and Finland (10). Next comes the group of countries with emerging research activities, such as Singapore (7), Egypt (6), and Peru (6). Finally, the countries represented in light blue have more modest and weaker contributions. Examples include the United Arab Emirates (3), Turkey (3), Monaco (2), Ukraine (2), Iran (1), Lebanon (1), and Morocco (1).

While analyzing countries' scientific output highlights the areas in which research about artificial intelligence and customer experience is concentrated, it does not determine the extent to which this knowledge is the result of international collaboration. To answer this question, we need to examine the lines connecting countries (Fig. 2). Each line indicates that authors from two countries have co-authored at least one publication: the thicker the line, the stronger the collaboration between the connected countries.

The countries that collaborated the most were the United Kingdom, with a total of 14 collaborations, including India (4 publications), China (3 publications), Australia (2 publications), and Indonesia (1 publication). France also has 14 collaborations with a single publication per collaboration, as do Canada, Estonia, Finland, Germany, Hungary, Lithuania, Romania, and Sweden.

Next is China with a total of 12 collaborations, including the United Arab Emirates (2 publications), the United States (2 publications), South Korea (1 publication), Egypt (1 publication), and Canada (1 publication). With the same number of collaborations (14), Malaysia collaborated on a single publication with countries such as Finland, Sweden, South Africa, Jordan, Pakistan, and Lithuania.

India and Finland follow closely behind, with collaborations covering 11 partner countries, while Australia and the United States each collaborate with 10 countries. Canada and Italy also demonstrate good international connectivity, collaborating with 8 different countries. In contrast, several countries show limited international connectivity, collaborating with only one partner country.

C. Keyword Co-Occurrence

The use of keyword co-occurrence as a method for mapping research on a given topic is widespread in bibliometric analysis. It reveals the structure and clarity of research themes in a given field (Zhang and Wang). Keywords are, as their name suggests, essential for understanding a subject and providing an overview of the content; they can provide an overview without reading all articles, simply by analyzing the links between nodes and counting the weight of the link that connects the keywords. From 173 articles, 633 keywords had emerged. With a minimum of 4 occurrences, 40 standardized keywords were included in the analysis. Keywords with the highest co-occurrence and total link strength are listed in Table I.

The co-occurrence output has identified four clusters; each cluster represents specific sub-research within the topic.

The first cluster (red) is more technical and focuses on the "how?" (natural language processing, predictive analytics, machine learning, learning systems) and the "in what context?" (banking sector, service sector). Most AI applications were found in financial services, particularly in banking, in the areas of risk management (operational efficiency, data confidentiality) and customer experience (customer service, customer

satisfaction). Artificial Intelligence provides theoretical frameworks to financial services providers, AI developers, marketers, policymakers, and academics.

TABLE I. TOP 10 KEYWORDS

Keyword	Co-Occurrence	Total link strength
Artificial intelligence	70	181
Customer experience	36	153
Sales	30	155
User experience	20	94
Chatbots	14	66
Machine learning	14	63
Customer-service	9	57
Natural language processing systems	8	80
Learning systems	8	77
Predictive analytics	8	55

The second cluster (green) shows the dominance of "artificial intelligence" (the largest node), which is linked to the following keywords: machine learning, big data, sentiment analysis, marketing, digital marketing, service marketing, personalization, financial services and systematic literature review. This cluster focuses on the use of technology in marketing, especially artificial intelligence and machine learning. We can tell that AI is used most in financial services and that it gained the attention of researchers since it has a direct link to systematic literature review keywords.

The third cluster (bleu) shows the keyword sales as the center node; it is linked to customer interactions, consumer behavior, chatbots, machine learning algorithms, personalization, service marketing, social media, and user experience. The primary theme of this cluster is the commercial applications of chatbots and ML in customer experience and personalization.

The fourth cluster (yellow) shed light on customer experience as the center node, which related successively to artificial intelligence, customer engagement and service quality. It primarily reflects research on AI-based personalization and customer engagement, encompassing studies on chatbots, recommendation systems, and data-driven customer journey personalization. This section highlights how companies are leveraging artificial intelligence to improve customer interactions and perceived value.

Together, these groups demonstrate that the field revolves around complementary trends linking technological capabilities, managerial strategy, and analytical rigor, thus confirming the interdisciplinary and application-oriented nature of AI-driven customer experience research.

D. Most Relevant Journals and their Thematic Orientation

Fig. 4 shows the ten most relevant journals in terms of number of publications in the analyzed data (Fig. 4). The results reveal that research on artificial intelligence and customer experience is spread across a wide range of academic journals, with no single publication overwhelmingly dominating publication output. IEEE Access emerges as the most productive journal with six published articles, it is primarily associated with the following keywords artificial intelligence, digital marketing, marketing, customer experience and customer engagement, indicating a strong focus on AI-driven approaches to costumer related research (see Fig. 6). followed by the Journal of Research in Interactive Marketing with five publications devoted to AI, particularly chatbot interaction with consumers. Next are journals that have each published four articles on artificial intelligence, machine learning, marketing, customer engagement, and customer experience, including the Journal of Retailing and Consumer Services, the Journal of Theoretical and Applied Electronic Commerce, and Kybernetes.

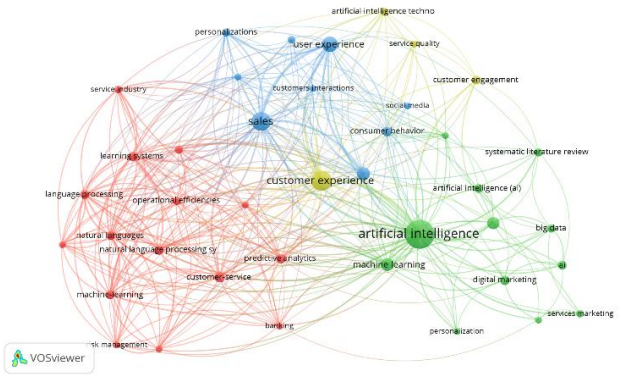


Fig. 4. Keyword co-occurrence.

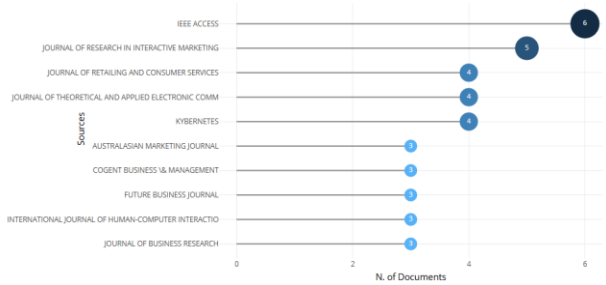


Fig. 5. Top ten journals.

Other journals, such as the Australian Marketing Journal, Cogent Business and Management, Future Business Journal, International Journal of Human-Computer Interaction, and the Journal of Business Research, each have a total of three publications. The publications in these journals focus on the application of AI (machine learning, chatbots) in digital marketing, particularly in relation to customers.

E. Co-citation

Mapping the intellectual structure of a given field complements meta-analysis and qualitative synthesis of literature reviews [6]. Co-citation analysis is a bibliometric technique that shows the coherence of publications by determining the frequency of documents that cite two documents together [7],[8]; it can be performed based on authors [9] or documents. Since the study aims to reveal related documents that represent common theoretical and conceptual foundations and findings in the fields of artificial intelligence and customer experience, the analysis will be performed on the basis of document co-citations. As shown in Fig. 7, the co-citation analysis results reveal three primary clusters.

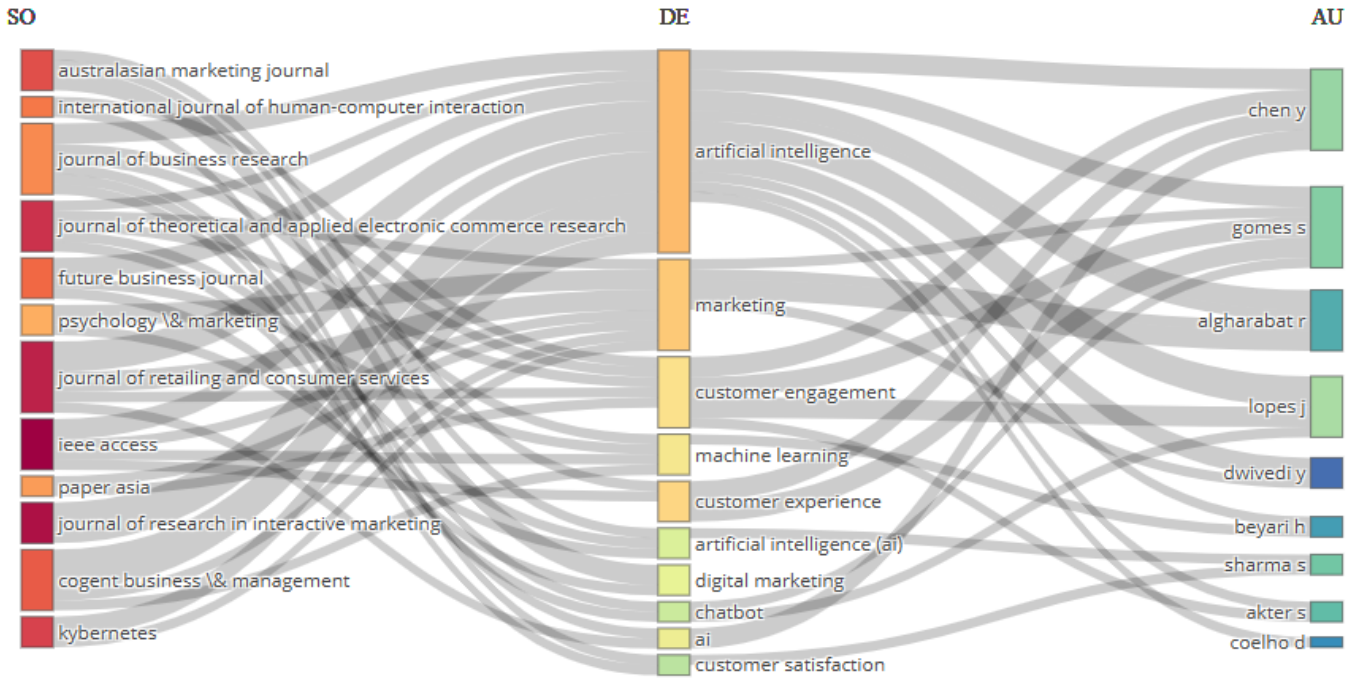


Fig. 6. Three field plot.

The first cluster (blue) is the largest and densest. It clearly focuses on the acceptance and adoption of artificial intelligence using the TAM model [10], customer engagement and experience [11],[12], digitization of services, and AI-based services for marketing transformation [13],[14],[15]. These studies are most often cited because they provide theoretical foundations that explain how the adoption of AI, as a cutting-edge technology, contributes to improving customer experience and engagement.

The second cluster (red) reflects the strategic and managerial implications of applying artificial intelligence in businesses.

The third cluster (green) focuses on the methodological aspect and measurement of artificial intelligence in the field of customer experience to support empirical research and quantitative modeling. This cluster links articles that do not focus solely on artificial intelligence or customer experience, but rather articles that analyze and test the theoretical relationship between AI and customer experience outcomes through the application PLS-SEM [16]. It highlights the growing interest among researchers in the empirical examination of this field.

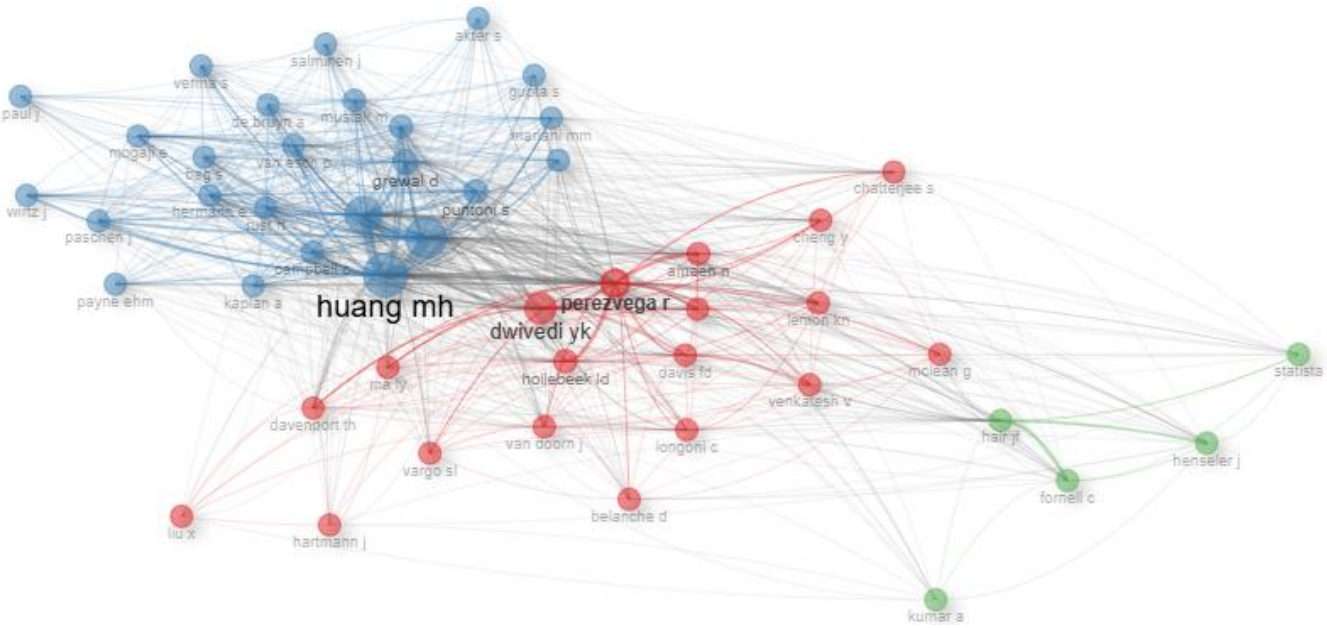


Fig. 7. Co-citation network.

F. Most Cited Publications

The relevant articles are determined based on their number of citations. Table II shows the 10 most cited publications at the local and global levels. Local citations measure the number of times an article is cited in the field of artificial intelligence and customer experience. Global citations, on the other hand, measure citations across the entire scientific database [17].

At the top of the list is the article with 311 global citations, but in this bibliometric analysis, it has 17 citations. It discusses the role of artificial intelligence in providing personalized offers to customers and provides predictions about customer management practices [18]. Next is the second most cited article, which explores the role of chatbot marketing efforts in establishing the relationship between a brand and its customers. It highlights that the most important elements of CMEs are information, interaction, accessibility, entertainment, and personalization [19]. This article has 171 global citations and 7 local citations. Next is the third most cited article, with 110 global citations and 4 local citations, which examines how AI has changed consumer decision-making, impacting marketing, management, and research services [20]. In fourth place is the article with 110 global citations and 4 local citations, which focuses on the challenges of including customers in financial systems through the integration of AI into marketing practices [21].

Several articles on the list feature the same two local citations. Two of these articles examine the use of AI and machine learning to extract actionable customer insights from social media data [22],[23], while the other two articles examine the application of these technologies in the areas of strategy, customer loyalty [24], and personalized engagement [25]. Finally, there are two emerging articles with one local citation. Together, these articles extend AI and customer research beyond commercial application to innovation in social inclusion centered on AI-human interaction [26],[27].

TABLE II. TOP TEN ARTICLE

Document	Year	Local Citations	Global Citations	LC/GC Ratio (%)	Normalized Local Citations	Normalized Global Citations
Understanding the Role of Artificial Intelligence in Personalized Engagement Marketing	2019	17	311	5,47	3,00	2,33
Customer–brand relationship in the era of artificial intelligence: understanding the role of chatbot marketing efforts	2022	7	171	4,09	13,22	2,26
AI voice bots: a services marketing research agenda	2020	4	110	3,64	5,33	0,89
The implications of artificial intelligence on the digital marketing of financial services to vulnerable customers	2021	3	110	2,73	12,00	0,95
Supporting customer-oriented marketing with artificial intelligence: automatically quantifying customer needs from social media	2020	2	49	4,08	2,67	0,39
Detecting Pain Points from User-Generated Social Media Posts Using Machine Learning	2022	2	27	7,41	3,78	0,36
Artificial intelligence in customer retention: a bibliometric analysis and future research framework	2024	2	8	25,00	24,67	0,50
AI-driven personalization: Unraveling consumer perceptions in social media engagement	2025	2	28	7,14	30,80	10,47
Artificial intelligence service agents: a silver lining in rural India	2024	1	3	33,33	12,33	0,19
Anthropomorphism in artificial intelligence: a game-changer for brand marketing	2025	1	8	12,50	15,40	2,99

IV. DISCUSSION

This bibliometric analysis aimed to map research on artificial intelligence and customer experience to explore their interactions. The results provided important insights into the field, answering the research questions posed in the introduction.

The interest in AI and customer experience only began to grow at the end of 2023 and increased in 2025 with 77 publications. This growth can be explained by the fact that artificial intelligence is attracting increasing interest in the business world [3]. This acceleration in the volume of publications goes hand in hand with technological advances in artificial intelligence and the challenges they pose for businesses [28], particularly regarding customer experience, including personalization, customer engagement, and service automation.

However, that acceleration of publications is not geographically even; the analysis of AI customer experience research activities around the world has revealed that research is concentrated in a limited number of leading countries, meaning that global contributions to knowledge creation in this field are uneven. The countries that contribute most to research are those with good technological infrastructure [29], research funding, and more developed digital economies, which automatically facilitate research. These countries are the United States, China, India, the United Kingdom, and Australia. However, some countries, such as Malaysia, Indonesia, and Egypt, have moderate research activity, even though AI is still in its infancy. Then some countries have contributed to research only once or twice, including Morocco, Somalia, Pakistan, etc. Most of these countries are still researching the adoption and acceptance of AI.

Fig. 3 indicates that countries with higher scientific productivity also serve as hubs in the global research network. By collaborating, they enable the development of diverse and interdisciplinary perspectives (Fig. 3). However, collaboration with other countries from emerging economies results in their inclusion in future research, giving them visibility and allowing

them to contribute to specific contexts, particularly in service sectors such as banking [30], where there is a strong interest in AI and customer experience.

Beyond the geographical distribution of research in AI in the customer experience domain, the relevant journals (Fig. 5) provide further information on how the research is distributed globally. Articles written by leading countries are more likely to be published in interdisciplinary journals that are oriented in engineering and analytics rather than being published in a single source.

It is also important to examine the intellectual structure of the field, identifying the fundamental authors and studies that shape research.

Co-citation analysis reveals that research on AI and customer experience revolves around three main themes. The dominant theme provides the theoretical foundations that explain how the adoption of AI, as a cutting-edge technology, contributes to improving customer experience and engagement. The second theme reflects the strategic and managerial implications of applying artificial intelligence in businesses. It connects all articles that examine how adopting AI in a business can create value [31], improve service innovation [32], and the implications of AI in decision-making [33]. The third theme is primarily methodological and brings together studies on quantitative modeling, which provides the analytical rigor needed to test the relationships between AI and customer experience.

The co-occurrence analysis confirms these dominant themes and provides further insight into them. It focuses on AI and its components (natural language processing, predictive analytics, machine learning, and learning systems) as well as sales and customer experience, reflecting a strong interest in the use of AI in sales to improve customer interactions and optimize service delivery.

Overall, all the analyses carried out in this bibliometric study describe a globally interconnected field of research that is gradually evolving towards greater specialization and increased use of AI.

V. Conclusion

Looking at the data as a whole, it is clear that research has been growing rapidly in recent years (2023, 2024, 2025) and that the field of artificial intelligence and customer experience revolves around AI innovation, customer-centric strategy, and methodological frameworks. AI is therefore transforming the way businesses, particularly marketers, interact with customers through service automation and AI-based personalization. Furthermore, over the years, academics have shifted from defining AI as a new individual concept to how AI drives sales and customer experience, meaning that the customer has become a central variable in AI research.

This study has enabled comprehensive conclusions to be drawn on the research topic. However, it is worthwhile mentioning certain limitations that affect the interpretation of the results and imply directions for future perspectives.

The first limitation is that the study was based on only two databases (Scopus and Web of Science). Although these are renowned for their comprehensive coverage and the quality of their scientific publications, they do not cover all of the literature on artificial intelligence and customer experience. In addition, other types of documents (conference proceedings, non-indexed articles, industry reports) were excluded from the analysis, which may have led to interesting contributions being overlooked. Secondly, the research map provided by this bibliometric analysis is interesting because it gives an overview of the subject, but it is not sufficient to explore the research questions in depth. Third, during this study, it was observed that new, impactful articles appear to be less influential, as older publications naturally accumulate more citations.

To overcome these limitations, future research could include additional databases and sector-specific reports. For a more in-depth analysis, bibliometric analysis could be supplemented by qualitative content analysis, systematic literature review or empirical research. Furthermore, the number of citations alone should not be used to determine whether an article has greater impact. To account for citation lag bias, the study used normalized citation metrics provided by Bibliometrix, which adjust citation counts for the year of publication and allow for a more equitable comparison between older and newer publications.

References

[1] W. X. Chen, T. T. Shi, and S. Srinivasan, "The Value of AI Innovations," *Harvard Business School Working Paper*, No. 24-069, May 2024.

[2] M. Haenlein and A. M. Kaplan, "A brief history of artificial intelligence: On the past, present, and future of artificial intelligence," *California Management Review*, vol. 61, no. 4, pp. 5–14, 2019.

[3] Suparna Biswas et al., *AI-Bank of the Future: Can Banks Meet the AI Challenge?* (McKinsey & Company, September 2020).

[4] N. J. van Eck and L. Waltman, *VOSviewer: A computer program for bibliometric mapping*, Tech. Rep. ERS-2009-005-LIS, Erasmus University Rotterdam, Erasmus Research Institute of Management, 2009.

[5] Aria, M., & Cuccurullo, C. (2017). bibliometrix: An R-tool for comprehensive science mapping analysis. *Journal of Informetrics* , 11(4), 959-975

[6] Zupic, I., & Čater, T. (2014). Bibliometric Methods in Management and Organization. *Organizational Research Methods*, *18*(3), 429-472. https://doi.org/10.1177/1094428114562629 (Original work published 2015)

[7] A. Gupta, S. Gupta, M. Bisht, P. Hooda, and M. Salik, "Document Cocitation Analysis using Concept Lattice," Engineering Technology & Applied Science Research, vol. 13, no. 5, pp. 11837–11842, Oct. 2023, doi: 10.48084/etasr.6201.

[8] N. Mustafee, K. Katsaliaki, and P. Fishwick, "Exploring the modelling and simulation knowledge base through journal co-citation analysis," Scientometrics, vol. 98, no. 3, pp. 2145–2159, Oct. 2013, doi: 10.1007/s11192-013-1136-z.

[9] X. Zhao, J. Zuo, G. Wu, and C. Huang, "A bibliometric review of green building research 2000–2016," Architectural Science Review, vol. 62, no. 1, pp. 74–88, Jun. 2018, doi: 10.1080/00038628.2018.1485548.

[10] Mohammed A, Ferraris A (2025;), "Exploring motivational drivers of AI chatbot adoption in emerging markets: insights from the stimulus-organism-response model for service automation". *The Bottom Line*, Vol. ahead-of-print No. ahead-of-print. https://doi.org/10.1108/BL-11-2024-0189

[11] Jayasingh, Sudarsan, Arunkumar Sivakumar, and Arputha Arockiaraj Vanathaiyan. 2025. "Artificial Intelligence Influencers' Credibility Effect on Consumer Engagement and Purchase Intention" *Journal of Theoretical*

and Applied Electronic Commerce Research 20, no. 1: 17. https://doi.org/10.3390/jtaer20010017

[12] Mekhail Mustak, Heli Hallikainen, Tommi Laukkanen, Loïc Plé, Linda D. Hollebeek, Majid Aleem, Using machine learning to develop customer insights from user-generated content, Journal of Retailing and Consumer Services, Volume 81, 2024, 104034, ISSN 0969-6989, https://doi.org/10.1016/j.jretconser.2024.104034.

[13] Shahriar Akter, Saida Sultana, Marcello Mariani, Samuel Fosso Wamba, Konstantina Spanaki, Yogesh K. Dwivedi, Advancing algorithmic bias management capabilities in AI-driven marketing analytics research, Industrial Marketing Management, Volume 114, 2023, Pages 243-261, ISSN 0019-8501, https://doi.org/10.1016/j.indmarman.2023.

[14] Cheng Y, Jiang H (2022), "Customer–brand relationship in the era of artificial intelligence: understanding the role of chatbot marketing efforts". *Journal of Product & Brand Management*, Vol. 31 No. 2 pp. 252–264, doi: https://doi.org/10.1108/JPBM-05-2020-2907

[15] Balakrishnan, J., & Dwivedi, Y. K. (2021). Role of cognitive absorption in building user trust and experience. *Psychology & Marketing*, *38*(4), 643-668. https://doi.org/10.1002/mar.21462

[16] Fandi Omeish, Mohammad Al Khasawneh, Nadine Khair, Investigating the impact of AI on improving customer experience through social media marketing: An analysis of Jordanian Millennials, Computers in Human Behavior Reports, Volume 15, 2024, 100464, ISSN 2451-9588, https://doi.org/10.1016/j.chbr.2024.100464.

[17] Naveen Donthu, Satish Kumar, Debmalya Mukherjee, Nitesh Pandey, Weng Marc Lim, How to conduct a bibliometric analysis: An overview and guidelines, Journal of Business Research, Volume 133, 2021, Pages 285-296, ISSN 0148-2963, https://doi.org/10.1016/j.jbusres.2021.04.070

[18] Kumar, V., Rajan, B., Venkatesan, R., & Lecinski, J. (2019). Understanding the Role of Artificial Intelligence in Personalized Engagement Marketing. *California Management Review*, *61*(4), 135-155. https://doi.org/10.1177/0008125619859317 (Original work published 2019)

[19] Cheng Y, Jiang H (2022), "Customer–brand relationship in the era of artificial intelligence: understanding the role of chatbot marketing efforts". *Journal of Product & Brand Management*, Vol. 31 No. 2 pp. 252–264, doi: https://doi.org/10.1108/JPBM-05-2020-2907

[20] Phil Klaus, Judy Zaichkowsky, AI voice bots: a services marketing research agenda, Journal of Services Marketing, Volume 34, Issue 3, 2020, Pages 389-398, ISSN 0887-6045, https://doi.org/10.1108/JSM-01-2019-0043.

[21] Mogaji, E., Soetan, T. O., & Kieu, T. A. (2020). The implications of artificial intelligence on the digital marketing of financial services to vulnerable customers. *Australasian Marketing Journal*, *29*(3), 235-242. https://doi.org/10.1016/j.ausmj.2020.05.003 (Original work published 2021)

[22] Kühl, N., Mühlthaler, M. & Goutier, M. Supporting customer-oriented marketing with artificial intelligence: automatically quantifying customer needs from social media. *Electron Markets* **30**, 351–367 (2020). https://doi.org/10.1007/s12525-019-00351-0

[23] Salminen, J., Mustak, M., Corporan, J., Jung, S., & Jansen, B. J. (2022). Detecting Pain Points from User-Generated Social Media Posts Using Machine Learning. *Journal of Interactive Marketing*, *57*(3), 517-539. https://doi.org/10.1177/10949968221095556 (Original work published 2022)

[24] Singh C, Dash MK, Sahu R, Kumar A (2024), "Artificial intelligence in customer retention: a bibliometric analysis and future research framework". *Kybernetes*, Vol. 53 No. 11 pp. 4863–4888, doi: https://doi.org/10.1108/K-02-2023-0245

[25] Tanawat Teepapal, AI-driven personalization: Unraveling consumer perceptions in social media engagement, Computers in Human Behavior, Volume 165, 2025, 108549, ISSN 0747-5632, https://doi.org/10.1016/j.chb.2024.108549.

[26] Ravi H, Vedapradha R (2024), "Artificial intelligence service agents: a silver lining in rural India". *Kybernetes*, Vol. 53 No. 8 pp. 2662–2678, doi: https://doi.org/10.1108/K-09-2022-1239

[27] Gomes, S., Lopes, J.M. & Nogueira, E. Anthropomorphism in artificial intelligence: a game-changer for brand marketing. *Futur Bus J* **11**, 2 (2025). https://doi.org/10.1186/s43093-025-00423-y

[28] Danhao Wang, Daogang Peng, Dongmei Huang, Application and prospects of large AI models in virtual power plants, Electric Power Systems Research, Volume 241, 2025, 111403, ISSN 0378-7796, https://doi.org/10.1016/j.epsr.2024.111403.

[29] UNCTAD. (2021). Technology and Innovation Report: Catching Technological Waves. Unated Nations.

[30] Sheth JN, Jain V, Roy G, Chakraborty A (2022), "AI-driven banking services: the next frontier for a personalised experience in the emerging market". *International Journal of Bank Marketing*, Vol. 40 No. 6 pp. 1248–1271, doi: https://doi.org/10.1108/IJBM-09-2021-0449

[31] Gao, B., & Huang, L. (2019). Understanding interactive user behavior in smart media content service: An integration of TAM and smart service belieffactors. *Heliyon*, *5*(12), e02983. https://doi.org/10.1016/j.heliyon.2019.e02983

[32] Celina M. Olszak, Kamila Bartuś, AI-enhanced Business Intelligence for decision-making, Procedia Computer Science, Volume 270, 2025, Pages 415-425, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2025.09.160.

[33] He, J., Luo, Y. & Wang, T. iDigiChat: intelligent digital marketing service chatbot for providing efficient customer services using artificial intelligence. *Sci Rep* 15, 33074 (2025). https://doi.org/10.1038/s41598-025-14722-5

AI-Driven Robotic Waste Sorting for Techno-Economic Assessment in Urban Indonesia

Ida Nurhaida[1], Mohammad Nasucha[2], Hari Nugraha[3]
Department of Informatics, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia [1, 2]
Department of Product Design, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia [3]
Center of Urban Studies, Universitas Pembangunan Jaya, Tangerang Selatan, Indonesia [1, 2, 3]

***Abstract*—Urban centers in Indonesia are facing increasing pressure in managing municipal solid waste as a result of rapid population growth, rising labor costs, and stricter demands for high-purity recyclable materials. Manual sorting at Material Recovery Facilities has become progressively less efficient and economically burdensome under these conditions. This study presents an artificial intelligence-driven robotic waste sorting system designed and evaluated under real operational conditions in Jakarta and South Tangerang. The system integrates YOLO based object detection, vision-guided robotic manipulation, real-time processing hardware, a multi-axis gantry system with stepper motors, and a custom conveyor mechanism to deliver waste items to the sorting cell. Unlike previous studies that mainly focus on algorithmic accuracy or laboratory-scale validation, this work combines real-world technical performance assessment with a localized techno-economic analysis. Experimental results show an average sorting accuracy of 90%, a material purity of 95.1%, and a throughput of 50 items per minute, outperforming typical manual sorting performance. An economic evaluation based on local wage levels, electricity tariffs, and recyclable market prices indicates a payback period of 4.3 to 4.9 years. The main contributions of this study lie in integrating AI vision and robotic sorting into unstructured urban waste environments, in empirical validation under Indonesian operating conditions, and in demonstrating economic feasibility for emerging economies. Although the case study focuses on Jakarta and South Tangerang, the findings are relevant for metropolitan areas across the Global South seeking more efficient and sustainable waste management solutions.**

***Keywords*—Smart waste sorting; AI vision; YOLOv12; computer vision; material recovery facility; techno-economic assessment; urban Indonesia; circular economy**

I. Introduction

Municipal solid waste (MSW) management has emerged as one of the most pressing sustainability challenges of the twenty-first century. Global MSW generation exceeds two billion tons annually and is projected to increase by up to 70 percent by 2050 if current practices persist [1]. The challenge is particularly acute in developing economies, where rapid urbanization, limited landfill capacity, and underfunded waste systems limit effective management of growing waste volumes [2]. Indonesia exemplifies this situation, especially in major metropolitan areas such as Jakarta and its surrounding cities, where rising waste volumes strain existing infrastructure and threaten environmental sustainability, public health, and social equity.

A key bottleneck in the waste management chain is sorting. Sorting determines the quality of downstream recycling systems and strongly influences material recovery, commodity value, and the overall economics of circular waste systems [3], [4]. In Indonesia, sorting in Material Recovery Facilities (MRFs) is predominantly manual, characterized by low throughput, inconsistent purity levels, and considerable health risks for workers exposed to hazardous materials [5]. Moreover, increasing regional minimum wages and tightening purity requirements from international buyers are placing operational pressures on MRFs that current manual methods struggle to meet [6].

Plastic waste intensifies the challenge. National data indicate that only 7-14% of plastic waste is recycled, despite high annual production volumes, resulting in both environmental burdens and economic inefficiencies [7]. Urban regions, such as Jakarta and South Tangerang, face particularly high consumption density and dwindling landfill capacity, underscoring the urgency of adopting advanced sorting technologies that can improve material purity and recovery rates.

Advances in Artificial Intelligence (AI), computer vision, and robotics offer promising opportunities to address these limitations. AI-driven robotic sorting has shown potential to deliver higher accuracy, consistent performance, and improved worker safety compared with manual operations [7], [8], [9]. However, most empirical evidence and techno-economic studies originate from high-income countries, raising concerns about their applicability to emerging economies where labor costs, waste composition, and market dynamics differ significantly.

This study addresses this gap by evaluating the technical performance and economic feasibility of an AI-driven robotic waste-sorting system in Indonesia. Using Jakarta and South Tangerang as case studies, it integrates experimental validation of sorting accuracy, purity, and throughput with localized techno-economic analysis. The findings aim to inform MRF modernization strategies and to support Indonesia's transition toward a more sustainable, circular waste management system.

The contributions of this study are explicitly defined as follows: First, this work presents the design and implementation of an AI-driven robotic waste sorting system that integrates vision-based detection, robotic manipulation, and a custom conveyor and gantry mechanism for unstructured municipal solid waste. Second, the system is experimentally validated under real operational conditions in Jakarta and South Tangerang, providing empirical evidence beyond laboratory-

scale demonstrations. Third, the study integrates technical performance evaluation with a localized techno-economic analysis based on Indonesian wage levels, electricity costs, and recyclable market prices, thereby bridging the gap between robotics research and investment feasibility in emerging economies.

II. Literature Review

The global increase in municipal solid waste generation has triggered growing concern among policymakers, researchers, and urban managers, as waste volumes rise faster than many cities can accommodate and pressure mounts on existing waste management systems, especially in countries with limited collection and treatment infrastructure. Indonesia is among the countries facing this challenge most acutely, where rapid urbanization and increasing consumption continue to widen the gap between waste generation and waste processing capacity.

Sorting plays a fundamental role in enabling effective recycling and circular material flows. Poor sorting quality directly increases contamination, reduces material recovery, and limits the profits that recycling facilities can generate. Manual sorting remains prevalent in Indonesia, but this approach faces persistent challenges, including variable performance, low speed, and significant health and safety risks for workers. As wages rise and global supply chains demand higher purity levels, these challenges have become increasingly difficult to address solely through manual methods. Indonesia's limited plastic recycling performance, with only a small fraction of total waste successfully recovered, demonstrates the structural limitations of current systems.

Artificial intelligence-powered robotic sorting offers a promising alternative. Vision-guided robotic systems are designed to operate in messy, unpredictable environments, making them especially well-suited for municipal solid waste streams. Object detection models such as YOLO perform effectively under diverse visual conditions and maintain high recognition accuracy even with contaminated or irregular items [10]. Advances in image understanding have further strengthened these systems through models that explicitly handle occlusion [11], segmentation methods that generalize across unknown object categories [12], and fast architectures for real-time industrial use [13]. Empirical studies consistently show that robotic sorting can offer both high purity and high throughput, often surpassing manual performance [14].

Economically, robotic systems require substantial upfront investment [15]. Over time, however, improvements in material purity and reductions in reliance on manual labor may offset these costs. Yet, techno-economic studies conducted in high-income countries may not accurately reflect the realities of emerging economies, where waste composition, wages, electricity costs, and market structures differ significantly [16]. As a result, context-specific evaluation is necessary.

Technical research on robotic manipulation continues to advance steadily. Early foundational work explored affordance-driven grasping and cross-domain visual matching [17], followed by approaches that leverage three-dimensional sensing and dual stream encoding to improve grasp detection in clutter [18]. Parallel to technical developments, sustainability and circular-economy research highlight the crucial role of advanced sorting technologies in reducing waste and improving resource efficiency [19]. Applied case studies demonstrate the potential of robotic sorting for beverage containers [3], mixed plastic streams [15], and multi-class waste environments [20].

Despite these contributions, a significant gap remains. Few studies combine technical performance assessment with localized economic analysis in developing countries [21], [22]. Many technical studies fail to examine operational costs or payback periods [23], while economic assessments heavily rely on assumptions drawn from high-income settings [24]. This gap highlights the need for integrated evidence that reflects both the technological capabilities and economic realities of waste management in contexts such as Indonesia.

III. Methodology

The methodology of this study is designed to bridge two complementary perspectives: the development of a robotic system for waste sorting in unstructured environments and its economic feasibility assessment in the Indonesian context. Combining technical experimentation with comparative financial evaluation demonstrates how such systems can be built and tested, and whether they are feasible in real-world Material Recovery Facilities (MRFs).

A. AI Vision Module and Model Training

The first step in the methodology involves designing an AI-driven vision pipeline that enables the robotic system to recognize and classify waste items in real time (see Fig. 1). A deep learning object detection framework, derived from the YOLO architecture, serves as the backbone of the vision system. The training dataset is intentionally constructed to reflect the diversity and unpredictability of MSW. Images are collected from local facilities and augmented with techniques such as rotation, scaling, and contrast variation to simulate the wide range of lighting, contamination, and occlusion conditions encountered in real MRF operations. The model outputs bounding boxes with confidence scores, which are filtered using adaptive thresholding to balance accuracy and false detection rates.

Waste categories include plastics (PET, HDPE, LDPE), paper, metals, and residuals. Model performance is evaluated with standard metrics including precision, recall, F1-score, and latency. The parameters ensure the system can operate within the time constraints imposed by conveyor-based sorting lines.

The development of the smart waste sorting system begins with the design and training of the AI vision module, which serves as the robotic platform's "eyes". As illustrated in Fig. 1, a camera captures images of mixed recyclable waste placed in front of the vision system. These captured frames constitute the raw dataset for object detection and classification.

The collected images are transferred to a PC, where preprocessing steps are performed, including resizing, normalizing, and annotating objects of interest such as PET bottles, aluminum cans, and HDPE containers. The annotated dataset is then divided into training, validation, and testing sets to ensure unbiased evaluation of the model's performance.

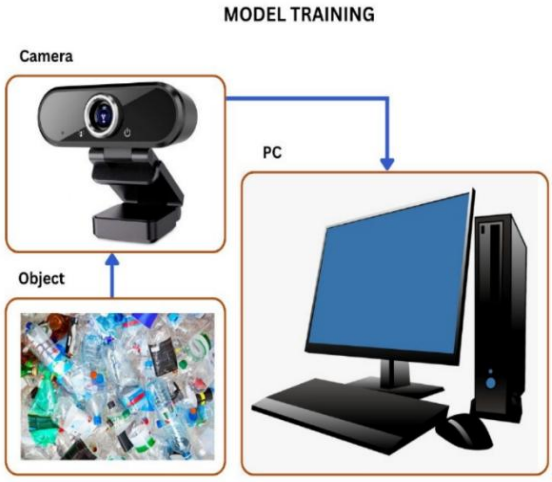


Fig. 1. Model training pipeline: a camera captures object images, which the PC processes for dataset preparation and training of the detection model.

Subsequently, the deep learning model is implemented using convolutional neural networks (CNNs) or state-of-the-art architectures such as YOLO. The model is trained on the dataset via iterative weight optimization using backpropagation, guided by performance metrics such as precision, recall, and mean average precision (MAP). The PC serves as the primary computational hub, providing sufficient processing power for the initial training phase and for later real-time inference tasks.

This stage ensures that the vision system can recognize objects under diverse conditions, including variations in lighting, orientation, occlusion, and material contamination, thereby enhancing robustness in real-world material recovery facilities (MRFs)

B. System Integration

After training the AI vision model, the system was integrated into a robotic platform for real-world validation. As illustrated in Fig. 2, the complete setup consists of visual sensors, a computation console, controllers, actuators, and supporting power electronics. Each component is critical in ensuring the smart waste sorting process operates seamlessly from perception to actuation.

1) Visual sensor: The system's visual sensor is a Full HD camera, 1920 × 1080 pixels (see Table I), positioned above the conveyor. The camera captures high-resolution images of mixed recyclable waste. These images are streamed to the computation unit in real time, where object detection and classification are carried out. The objects are even under variable lighting and can be accurately identified even under partial occlusion.

2) Computation console (heavy processing unit): The core of the system's intelligence resides in a PC with specifications comparable to those of a high-performance laptop. This console executes the YOLOv12 computer vision model for partial occlusion conditions and for clarity in captured frames, trained to detect and classify recyclable waste such as PET bottles, aluminium cans, and HDPE containers. Beyond classification, the PC also performs numerical calculations to estimate the coordinates of each detected object, which are then translated into actionable commands for the robotic manipulator.

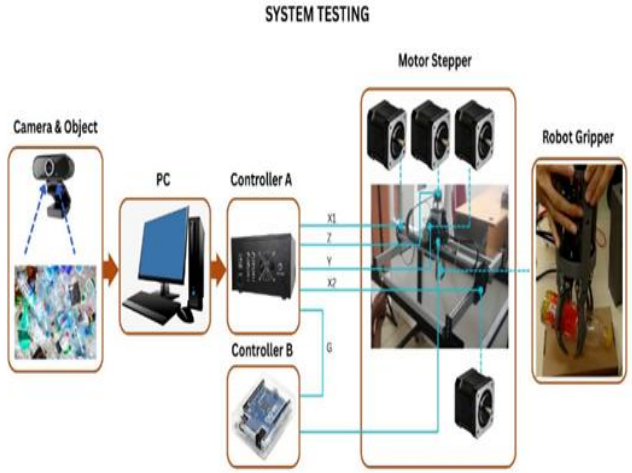


Fig. 2. Experimental setup for system testing: integrated flow of camera, PC, controller, stepper motors, and robotic gripper for smart waste sorting validation.

TABLE I. HARDWARE SPECIFICATIONS OF THE SMART WASTE SORTING SYSTEM

Component	Specifications	Qty	Function
Visual Sensor	Camera: Full HD 1920 × 1080	1 unit	Captures images of mixed recyclable waste for the AI vision module
Computation Console	PC (equivalent to a high-performance laptop) Computer Vision Model: YOLOv12	1 unit	Heavy computation for object detection, classification, and coordinate calculation
Controller A (Axis Translation)	ATMEGA328P microcontroller OLED 128 × 64 pxTB6600 motor drivers Power supply: 5V/1A + 24V/10A	1 set	Controls X1, X2, Y, Z translation using four stepper motors
Controller B (Gripper Control)	DC SPDT Relay 5V input (2 units)NPN BJT BD139 (2 units)Resistors 1kΩ/0.25W (2 units)Terminal Blocks 4–5 pins (2 units) PCB Power supply: 5V/1A and 12V/5A	1 set	Controls gripper open/close action
Actuators (Translation)	DC Stepper Motors: NEMA23Operating voltage: 24–48 V Operating current: 2–5 Bipolar (two coils)	4 unit	Provides motion along X1, X2, Y, and Z axes
Actuator (Gripper)	DC Motor: 12 V, 3–5 A	1 unit	Drives the gripper mechanism for grasping and releasing objects

- Controller A (Axis Translation and Motion Control): Controller A manages translation along the X1, X2, Y, and Z axes. It is designed using:
  - ATMEGA328P microcontroller, responsible for light computation and communication with the PC.
  - OLED Display (128 × 64 pixels), providing real-time system status and diagnostic feedback.
  - TB6600 motor drivers (4 units), ensuring precise control of the stepper motors.
  - Power supply unit with dual outputs (5V/1A and 24V/10A), supplying sufficient current to drive the motors and electronic circuitry.

The researchers developed the control logic and integration code, enabling smooth and synchronized multi-axis translation.

- Controller B (Gripper Control): Controller B is dedicated to controlling the robotic gripper for gripping and releasing actions. Its design includes:
  - DC SPDT Relays (2 units), switching the gripper motor between grip and release modes.
  - NPN BJTs (BD139, 2 units) and resistors (1 kΩ, 0.25W, 2 units) for signal amplification and circuit stability.
  - Terminal blocks (2 units, 4–5 pins each) for secure wiring connections.
  - PCB for component integration.
  - Independent power supplies: 5V/1A for control electronics and 12V/5A for the gripper motor.

The research team designed and fabricated the gripper controller, which ensures flexibility and durability during repeated grasping operations.

- Actuators: The actuation system consists of both translational and gripping mechanisms:
  - Stepper Motors (4 units, NEMA23), each with an operating voltage of 24–48 V and a current range of 2–5 A. These motors provide the precise translation required along the X1, X2, Y, and Z axes, enabling accurate positioning of the gripper relative to the target objects.
  - DC Motor (1 unit for the gripper), operating at 12 V with a current range of 3–5 A, delivering sufficient torque to grasp and release waste objects of varying shapes and weights.
- Conveyor Belt: A custom-designed conveyor belt was developed to transport waste materials through the working area to support the robotic sorting system. The conveyor serves as the primary medium for presenting recyclable objects to the vision system and positioning them within the workspace of the robotic manipulator.

This actuation system allows the robot to execute closed-loop pick-and-place tasks, reliably transferring objects into the correct bins. The conveyor was designed using a structural frame of 3×3 cm angle bars, ensuring sufficient rigidity and stability during operation. The overall dimensions are approximately 212 cm in length, 68 cm in width, and 40 cm in height, as illustrated in the technical drawing (see Fig. 3). The conveyor surface is supported by rollers at both ends, which enable continuous belt motion. A dynamo motor is mounted on one side of the frame and connected to the rollers to drive the belt with a stable and adjustable speed.

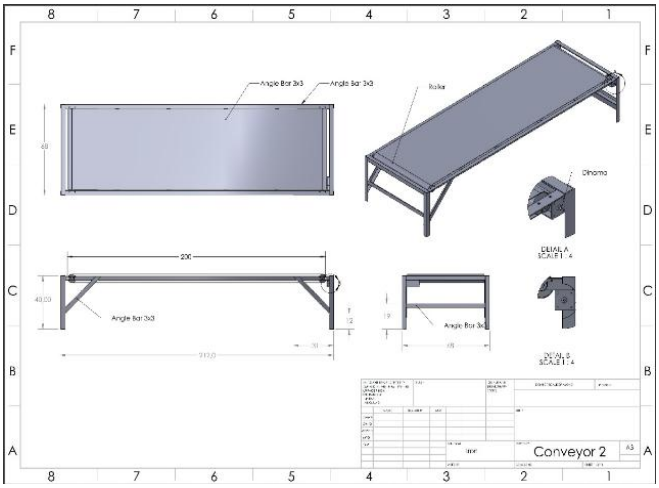


Fig. 3. Technical drawing of the conveyor belt showing structural dimensions and roller mechanism.

In its assembled form (Fig. 4), the conveyor integrates seamlessly with the robotic gantry system. The conveyor belt is made of durable synthetic material, providing adequate friction to prevent the slippage of lightweight objects, such as PET bottles or aluminium cans. The conveyor height has been optimized to align with the robotic gripper's workspace, ensuring a smooth handoff between the transport system and the pick-and-place mechanism. This conveyor system continuously introduces waste items into the sorting cell, enabling repeatable and scalable experiments. It also provides a controlled test environment for evaluating the robotic system under realistic material handling conditions, including varying object positions and orientations on the moving belt.



Fig. 4. Assembled conveyor belt integrated with the robotic gantry system.

C. Testing Workflow

The overall workflow of the system is designed as an open-loop sequence:

1) Image capture and object detection: The camera captures real-time images of waste objects. The PC processes these images using YOLOv12 to detect and classify objects, and to calculate their spatial coordinates.

2) Command generation and Axis control: The PC transmits coordinate-based commands to Controller A. The controller drives the stepper motors, moving the gantry system along the X1, X2, Y, and Z axes until the gripper is aligned with the target object.

3) Grasp execution: Once alignment is achieved, the PC signals Controller B. The relay-based circuit activates the DC motor, causing the gripper to close and secure the object.

4) Placement: The object is transported to the designated bin, where the gripper releases it.

5) Reset to start position: After each pick-and-place cycle, the gripper is automatically returned to its initial home position. This ensures consistency and repeatability, as the system begins from the same reference point for subsequent operations.

Unlike closed-loop systems, this configuration does not utilize external feedback sensors to confirm the gripper's position. Instead, the last known gripper position is tracked internally in the PC via program variables. This simplifies the design while ensuring repeatable operation. However, the absence of feedback may lead to long-term positional drift, offering an opportunity for future system enhancement.

D. Adapting to Unstructured Waste Environments

A defining feature of waste sorting is its unstructured and unpredictable nature. Unlike manufacturing lines, where objects are uniform, MSW streams contain irregular, overlapping, and sometimes contaminated items. This study addresses three core challenges:

1) Variability of object geometry: The dataset and robotic strategies are designed to accommodate irregularity using multimodal end-effectors and robust detection models.

2) Contamination and occlusion: Training data intentionally includes soiled, partially visible, or overlapping items, enabling the model to recognize objects under imperfect conditions.

3) Real-time constraints: System performance is tuned to complete detection and grasp planning within 1.5 seconds, a requirement for maintaining throughput at an industrially meaningful scale.

By addressing these challenges, the methodology ensures that the developed system does not remain a laboratory prototype but aligns closely with the operational realities of Indonesian MRFs.

E. Economic Evaluation Framework

The second component of the methodology assesses the economic feasibility of robotic sorting in comparison to manual operations. This analysis draws on data from the Jakarta and South Tangerang case studies, as well as secondary sources such as the Central Bureau of Statistics (BPS) wage data, local government regulations, and commodity price reports. Two scenarios are compared:

1) Manual sorting, the current practice in Indonesian MRFs, is characterized by labour-intensive processes, modest throughput, and variable purity levels.

2) Robotic sorting, based on the system described above, supplemented with benchmarks from [27] and global industry reports.

Key indicators include operating expenditure (OPEX), capital expenditure (CAPEX), cost per ton of processed waste, revenue uplift from higher material purity, and payback period. Sensitivity analyses are performed to simulate variations in labor costs, waste composition, and market prices of recyclables, ensuring robust findings across multiple scenarios.

F. Data Sources

The economic evaluation in this study relies on a combination of national statistics, municipal reports, and industry association data. For Jakarta, the minimum wage figures are sourced from the official publications of Statistics Indonesia (Badan Pusat Statistik), which provide annually updated provincial minimum wage data [28], while corresponding figures for South Tangerang are taken from Statistics Indonesia, Banten Province [29]. These values serve as the baseline for calculating labor costs in manual and semi-automated sorting scenarios.

Price information in Jakarta is usually published in program reports, such as the price list for Bank Sampah Induk Satu Hati, issued by the Jakarta Environmental Agency [30]. At the national level, price information on recycled plastics is often provided through policy or academic studies that cite ADUPI data [31]. However, the South Tangerang Environmental Agency (DLH Tangsel) has not released disaggregated public data on the prices of PET, glass, and aluminum materials. To address this limitation, we use Jakarta data as a proxy, supported by ADUPI market ranges, and apply a ±20% sensitivity adjustment to account for potential local variation.

This approach ensures that the analysis remains grounded in realistic market conditions while acknowledging the absence of fully disaggregated municipal price data for South Tangerang. For future studies, a structured market survey targeting local waste banks, aggregators, and recyclers in South Tangerang is recommended. Such an instrument could systematically capture unit prices, quality categories, and transaction conditions, thereby improving the robustness of techno-economic analyses in emerging urban contexts.

G. Validation Strategy

This study's validation focuses on the robotic system's ability to separate three critical recyclable categories: plastic bottles, glass containers, and aluminum cans. These categories are prioritized because they constitute high-value materials in the Indonesian recycling market and are frequently handled in both manual and mechanized sorting. The validation process is structured around three key indicators: separation accuracy, throughput performance, and material purity.

Separation accuracy refers to the proportion of correctly sorted items relative to the total number of items processed in each category. It is expressed mathematically as:

$$Accuracy = \frac{C_i}{T_i} \times 100\% \quad (1)$$

where, C_i is the number of correctly sorted items of category i, and T_i is the total number of items of category i presented to the system. This clearly measures how reliably the robotic arm can distinguish between plastics, glass, and aluminum.

Throughput performance evaluates the number of items correctly picked and placed per unit time under conveyor belt conditions. It is measured as:

$$Throughput = \frac{N}{\Delta t} \quad (2)$$

where, N is the number of items sorted, and Δt is the duration of operation. This indicator reflects whether the system can achieve competitive processing speeds with manual workers, who typically manage 20 to 30 picks per minute in Indonesian MRFs.

Material purity captures the extent to which the output stream of each recyclable is free of contamination from other materials. It is expressed as:

$$Purity = \frac{C_i}{O_i} \times 100\% \quad (3)$$

where, O_i is the total number of items in the output stream for category i, a high purity score indicates that the robotic system produces cleaner, more valuable recyclable fractions, directly affecting economic outcomes.

Validation results from the robotic system are compared with manual sorting data collected from the Jakarta and South Tangerang facilities and with benchmarks reported in international studies. Manual operations commonly achieve purity levels in the 70% to 80% range with high variability, while robotic deployments in developed contexts report levels exceeding 90%. Applying these formulas, the study quantitatively assesses whether the proposed robotic system meets technical and practical thresholds for effective sorting in Indonesian MRFs.

The payback period for robotic sorting is calculated by dividing capital expenditure by the combined annual gains in revenue uplift and labour cost savings relative to manual sorting using the formula:

$$Payback = \frac{CAPEX}{\Delta Revenue + \Delta Labour} \quad (4)$$

IV. Results and Discussion

To evaluate the robotic system's capability under realistic conditions, a controlled validation was conducted using a batch of 300 items consisting of 100 plastic bottles (PET), 100 glass bottles, and 100 aluminum cans. These three categories were chosen because they represent the highest-value recyclable fractions in Indonesian MRF and constitute a significant proportion of international recycling markets. The robotic system's performance was benchmarked against manual sorting, which remains the dominant practice in Jakarta and South Tangerang.

TABLE II. Comparison of Manual and Robotic Sorting Validation Results (Controlled Test With 300 Items).

Metric	Manual Sorting	Robotic Sorting	Improvement
Accuracy (avg.)	75.0%	90.0%	+15.0 pp
Purity (avg.)	84.9%	95.1%	+10.2 pp
Throughput (items/min)	45	50	+5

The results are presented in Table II. Manual sorting achieved an average accuracy of 75% and purity of 84.9%, consistent with reported figures in the literature (Pawar et al., 2023). In contrast, the robotic system achieved an average accuracy of 90% and a purity of 95.1%, indicating improved separation and contamination control. Throughput was also higher, with the robotic system processing approximately 50 items per minute compared to 45 for manual sorting.

These results have two implications. First, the robotic system consistently outperforms manual sorting in controlled conditions, demonstrating its ability to address the' variability, contamination, and speed constraints of unstructured waste environments. Second, even modest improvements in throughput and purity translate into significant downstream benefits. Higher purity directly increases the value of recovered fractions by making them more competitive in local and international markets, while higher throughput reduces the cost per ton of operation.

The validation results were translated into economic terms by considering the average market values of recyclables in Jakarta and South Tangerang. Based on recent market data, the approximate prices of key materials are Rp 3,500/kg for PET bottles, Rp 300/kg for glass, and Rp 15,000/kg for aluminium cans. Assuming an average weight of 20 g per plastic bottle, 150 g per glass bottle, and 15 g per aluminium can, the 300-item test batch represents 6,000 g of plastic, 45,000 g of glass, and 4,500 g of aluminium.

Under manual sorting conditions, with an average purity of ~84,9%, the recyclables' effective recovered mass is reduced due to contamination. By contrast, robotic sorting achieves ~95.1% purity, producing more saleable material. Table III illustrates the difference in recovered value between the two systems for the test batch.

TABLE III. Estimated Recovered Value of Recyclables From Validation Batch (Jakarta– South Tangerang Market Prices).

Material	Qty (kg)	Price (Rp/kg)	Value (Manual, Rp)	Value (Robotic, Rp)
Plastic bottles	2.0	3,500	5,950	5,950
Glass	15.0	300	3,825	3,825
Aluminium	1.5	15,000	19,125	21,375
Total	—	—	**28,900**	**32,300**

Although the absolute difference in this small-scale validation is modest (Rp 3,390), scaling the results to an operational MRF handling 5 tons per hour reveals substantial gains. At an industrial scale, improving purity and accuracy could translate into an additional Rp 2–3 million in recovered

value daily, depending on commodity prices and waste composition.

While validation at the batch scale demonstrates the technical potential of robotic sorting, decision-making at the facility level ultimately depends on costs and revenues measured per ton of processed waste. This section provides a comparative analysis of three scenarios: 1) manual sorting, reflecting current practices in Jakarta and South Tangerang, 2) semi-automated sorting, which integrates conveyor systems with manual labor, and 3) AI-driven robotic sorting, based on the system described in this study.

Labor costs are calculated using 2024 minimum wage data: Jakarta (Rp 5,067,381 per month) and South Tangerang (Rp 4,857,000 per month). Throughput benchmarks are derived from field reports: manual workers process approximately 1.5 tons daily, semi-automated systems achieve around 2.5 tons per day, and robotic systems reach 3.5–5 tons per day per sorting cell. [32]. Material prices reflect prevailing market rates in Greater Jakarta: plastic bottles at Rp 3,500/kg, glass bottles at Rp 300/kg, and aluminium cans at Rp 15,000/kg. Capital expenditure for robotic sorting is assumed to be USD 400,000 (≈ Rp 6.4 billion), in line with reported costs in international case studies [6].

The comparative economic analysis is grounded on transparent assumptions that combine international benchmarks with local Indonesian data. These assumptions cover capital expenditure (CAPEX), operating expenditure (OPEX), labor costs, material prices, and exchange rates.

International studies indicate that robotic sorting cells typically require an initial investment between USD 300,000 and USD 600,000, depending on the configuration of robotic arms, vision systems, conveyors, and integration costs [6], [32]. For this study, a conservative midpoint of USD 400,000 is selected. The average exchange rate in 2024, which was approximately Rp 16,000 per United States dollar, results in an estimated value of around Rp 6.4 billion when converted using official figures from Bank Indonesia [33]. This assumption is reasonable for Indonesian conditions, where equipment is often imported, though actual costs may vary due to taxes, duties, and local integration expenses.

OPEX components include labor, electricity, maintenance, and consumables. Manual sorting relies heavily on labor, while robotic systems incur higher energy and maintenance costs but reduce labor intensity. For manual sorting, labor is assumed to represent ~80% of OPEX, while robotic systems shift this balance toward energy and maintenance.

Manual sorting is benchmarked at purity levels of 70–80% with a throughput of approximately 1.5 tons per worker daily. Semi-automated systems improve throughput to 2.5 tons daily and purity to 82–85%. Robotic systems, consistent with international reports, achieve 3.5–5 tons per day per cell with purity above 90%.

Table IV summarizes the cost and revenue structure per ton across the three scenarios. Manual sorting exhibits the lowest capital cost but the highest labor intensity, with relatively low purity (70–80%) and high output variability. Semi-automated systems modestly improve throughput and reduce labor requirements. Robotic sorting requires substantial capital investment but delivers superior purity (>90%), consistent throughput, and reduced operational expenditure over time.

These results underscore the long-term economic advantage of robotic sorting (see Table IV). Although upfront costs are substantial, the superior purity and reduced labor requirements lead to significantly higher margins per ton, enabling investment recovery within 4–5 years. The findings also suggest that South Tangerang, with slightly lower labor costs than Jakarta, benefits even more from robotic adoption because revenue uplift from higher purity becomes the primary driver of profitability. This analysis demonstrates that robotic sorting is a technological innovation and an economically rational pathway for modernizing MRFs in Indonesian metropolitan contexts.

TABLE IV. COMPARATIVE COST AND REVENUE PER TON ACROSS THREE SORTING SCENARIOS (JAKARTA AND SOUTH TANGERANG CONTEXT)

Parameter	Manual Sorting	Semi-Automated	Robotic Sorting
Labour cost per ton (Rp)	1,200,000	800,000	200,000
Energy & OPEX per ton (Rp)	300,000	400,000	600,000
Purity (%)	75–80	82–85	92–96
Revenue per ton (Rp)	2,200,000	2,500,000	3,200,000
Net margin per ton (Rp)	700,000	1,300,000	2,400,000

Fig. 5 presents the payback curve illustrating the economic trajectory of the robotic sorting system. The graph plots cumulative cash flow against years of operation for two capital expenditure assumptions: Rp 350 million and Rp 400 million. Both scenarios are based on an annual incremental revenue of Rp 81.6 million, calculated from the additional value of recyclables recovered due to the increase in purity from ~85% (manual) to ~95% (robotic). The curves show that the breakeven point is achieved between 4.3 years (Rp 350 million CAPEX) and 4.9 years (Rp 400 million CAPEX). After this point, the investment generates a positive surplus that increases linearly over time, reflecting stable gains in material recovery. By the end of a 10-year operational horizon, the cumulative net cash flow reaches Rp 466 million under the lower CAPEX scenario and Rp 416 million under the higher CAPEX scenario.

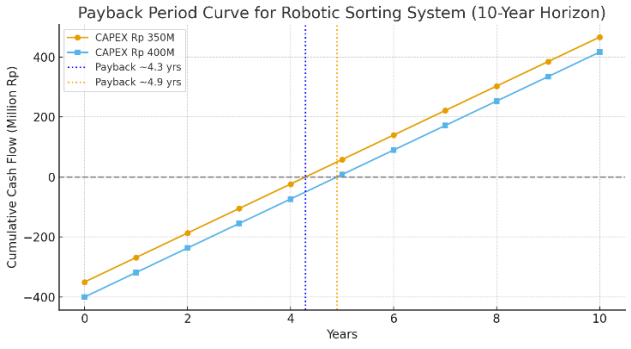


Fig. 5. The payback curve for a robotic sorting system showing cumulative cash flow (in billion rupiah) over five years under Jakarta—South Tangerang assumptions.

This finding confirms that robotic sorting is economically viable within the typical service life of robotic and automation systems (5–7 years). The result is also consistent with international studies reporting payback periods of 4–6 years for AI-based sorting technologies in medium-scale material recovery facilities [6], [12].

Structural inefficiencies in the national recycling industry also reinforce the economic potential of robotic sorting. Reports indicate that plastic recycling manufacturers in Indonesia operate at only about 70% of their installed capacity, primarily due to the low quality and inconsistent input materials. At the same time, demand for recycled plastics in packaging, textiles, and consumer goods continues to grow, but cannot be met by current supply chains. The proposed robotic sorting system can directly address this gap by consistently producing recyclables with purity levels exceeding 95%. Higher-quality outputs would increase the market value of recovered materials and improve the utilization rate of existing recycling plants. Linking these improvements to the economic assessment strengthens the argument that robotic sorting is technically viable and economically rational for advancing Indonesia's circular economy agenda.

The findings highlight several important implications for Indonesian waste management [25] [26]. From an operational perspective, robotic sorting reduces reliance on manual labor, improves worker safety by limiting exposure to hazardous materials, and enhances sorting consistency. While the capital expenditure of robotic systems remains significant, their ability to deliver higher throughput and purity directly improves revenue streams and reduces long-term operational costs.

From a policy perspective, the results suggest that investing in robotic sorting can strengthen Indonesia's position in global recycling markets, where quality standards are becoming increasingly stringent. Higher-purity recyclables are less likely to be rejected by international buyers, thereby reducing the risk of export losses. At the same time, adopting robotics aligns with broader national strategies for digital transformation and Industry 4.0, creating opportunities for skill development in robotics and AI while modernizing the waste sector.

It is essential to acknowledge the limitations of this validation. The test was conducted under controlled conditions with equal numbers of items per category, whereas actual MSW streams are more heterogeneous and unpredictable. Conveyor speeds, mechanical reliability, and integration with existing MRF infrastructure may also constrain throughput in industrial settings. Furthermore, the economic analysis relies on current market prices, which are subject to fluctuation.

V. Conclusion and Future Work

A. Conclusion

This study demonstrated that artificial intelligence-driven robotic waste sorting can achieve high technical performance and economic feasibility under real operational conditions in urban Indonesia. The experimental results confirmed that the proposed system improves sorting accuracy, material purity, and throughput compared with conventional manual sorting practices. At the same time, the techno-economic analysis showed that robotic sorting can reach realistic investment payback periods under local wage and market conditions. Compared with earlier studies that primarily focus on detection accuracy or robotic grasping in laboratory environments, this work advances the state of the art by integrating system-level validation with a localized economic assessment in an emerging economy context.

Several limitations should be acknowledged. Hardware constraints, such as conveyor speed, gripper durability, and lighting conditions, influence system performance. Dataset bias also exists, as the waste samples collected reflect the composition of specific urban areas and may not fully represent national variability. In addition, the techno-economic evaluation relies on current market prices and wage levels, which are subject to future fluctuations. These factors may affect the generalizability of the reported results across different regions and time horizons.

Despite these limitations, the findings provide a strong indication that AI-driven robotic sorting is a viable pathway for modernizing Material Recovery Facilities in developing countries. The integration of technical validation with localized economic analysis offers a practical decision-making tool for facility operators, policymakers, and investors. Future work will focus on multi-class waste expansion, adaptive learning for changing waste composition, and full-scale deployment studies to further support the transition toward a circular economy and sustainable urban waste management.

B. Future Work

While this study demonstrates the feasibility of AI-driven robotic waste sorting, several opportunities remain to extend and deepen the research.

1) Broader object categories: The current system focused primarily on PET bottles, aluminium cans, and HDPE containers. Future work should expand detection and sorting to include more diverse and complex waste fractions, such as multilayer packaging, paper composites, and textiles, which are prevalent in Indonesian municipal solid waste streams.

2) Adaptive and continual learning: Waste characteristics are highly dynamic, influenced by seasonal consumption patterns and contamination levels. Incorporating adaptive learning strategies or active learning pipelines would enable the vision model to continually improve as new data becomes available.

3) Enhanced sensing and control: The present system operates under an open-loop control mechanism. Integrating additional sensors, such as depth cameras, force sensors, or real-time feedback loops, could improve grasp stability, reduce error rates, and enable safer operation in mixed-material environments.

4) Scaling and industrial deployment: While the experiments were conducted as a case study in Jakarta and South Tangerang, future research should investigate system performance in full-scale MRFs. This includes studying

throughput under industrial load, conveyor integration at higher speeds, and robustness against continuous operation.

5) Socio-economic dimensions: Adopting robotic sorting has labour, workforce transition, and policy design implications. Future studies should evaluate how robotics can complement rather than displace workers, for example, through hybrid human-robot collaboration, reskilling programs, and inclusive circular economy strategies.

ACKNOWLEDGMENT

This work was partly supported by the Ministry of Education, Science, and Technology of the Republic of Indonesia under the Applied Research Grant Scheme with Model Output (Skema Penelitian Terapan Luaran Model) in 2025, Contract No. 125/C3/DT.05.00/PL/2025; 8087/LL4/PG/2025; and 007/PKS-LP2M/UPJ/06.25.

REFERENCES

[1] Silpa Kaza, Lisa Yao, Perinaz Bhada-Tata, and Frank Van Woerden, "What a Waste 2.0 A Global Snapshot of Solid Waste Management to 2050," 2018. doi: 10.1596/978-1-4648-1329-0.

[2] Daniel Hoornweg and Perinaz Bhada-Tata, "WHAT A WASTE: A Global Review of Solid Waste Management," Urban Development & Local Government Unit World Bank, 2012.

[3] T. Cheng, D. Kojima, H. Hu, H. Onoda, and A. H. Pandyaswargo, "Optimizing Waste Sorting for Sustainability: An AI-Powered Robotic Solution for Beverage Container Recycling," Sustainability (Switzerland), vol. 16, no. 23, Dec. 2024, doi: 10.3390/su162310155.

[4] N. Um, "Sustainable Waste Management in the Context of the Circular Economy," Mar. 01, 2025, Multidisciplinary Digital Publishing Institute (MDPI). doi: 10.3390/su17051937.

[5] A. G. Satav, S. Kubade, C. Amrutkar, G. Arya, and A. Pawar, "A state-of-the-art review on robotics in waste sorting: scope and challenges," International Journal on Interactive Design and Manufacturing (IJIDeM), vol. 17, no. 6, pp. 2789–2806, 2023, doi: 10.1007/s12008-023-01320-w.

[6] S.M. Mizanur Rahman and Barbara K. Reck, "Techno-Economic Analysis of a Material Recovery Facility Employing Robotic Sorting Technology," in Technology Innovation for the Circular Economy: Recycling, Remanufacturing, Design, Systems Analysis and Logistics, Wiley Online Library, 2024, ch. 21.

[7] Yuyun Ismawati et al., "Plastic Waste Management and Burden in Indonesia Country Situation Report," 2022. [Online]. Available: ipen-2021-indonesia-v1_1aw.pdf. Last accessed November 1st, 2025.

[8] J. Li and D. J. Cappelleri, "Sim-Grasp: Learning 6-DOF Grasp Policies for Cluttered Environments Using a Synthetic Benchmark," IEEE Robot Autom Lett, vol. 9, no. 9, pp. 7645–7652, 2024, doi: 10.1109/LRA.2024.3430712.

[9] Y. Zhou, "A YOLO-NL object detector for real-time detection," Expert Syst Appl, vol. 238, p. 122256, 2024, doi: https://doi.org/10.1016/j.eswa.2023.122256.

[10] L. Ke, Y.-W. Tai, and C.-K. Tang, "Occlusion-Aware Instance Segmentation Via BiLayer Network Architectures," IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 8, pp. 10197–10211, Aug. 2023, doi: 10.1109/TPAMI.2023.3246174.

[11] S. N. Rai, F. Cermelli, B. Caputo, and C. Masone, "Mask2Anomaly: Mask Transformer for Universal Open-Set Segmentation," IEEE Trans Pattern Anal Mach Intell, vol. 46, no. 12, pp. 9286–9302, 2024, doi: 10.1109/TPAMI.2024.3419055.

[12] Q. Zhang, L. Chen, M. Shao, H. Liang, and J. Ren, "ESAMask: Real-Time Instance Segmentation Fused with Efficient Sparse Attention," Sensors, vol. 23, no. 14, Jul. 2023, doi: 10.3390/s23146446.

[13] Z. Xia and J. Kim, "Enhancing Mask Transformer with Auxiliary Convolution Layers for Semantic Segmentation," Sensors, vol. 23, no. 2, Jan. 2023, doi: 10.3390/s23020581.

[14] L. Wu, Y. Chen, Z. Li, and Z. Liu, "Efficient push-grasping for multiple target objects in clutter environments," Front Neurorobot, vol. 17, 2023, doi: 10.3389/fnbot.2023.1188468.

[15] C. Lubongo, M. A. A. Bin Daej, and P. Alexandridis, "Recent Developments in Technology for Sorting Plastic for Recycling: The Emergence of Artificial Intelligence and the Rise of the Robots," Recycling, vol. 9, no. 4, Aug. 2024, doi: 10.3390/recycling9040059.

[16] J. Mahler et al., "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics," Aug. 2017, [Online]. Available: http://arxiv.org/abs/1703.09312

[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 779–788. doi: 10.1109/CVPR.2016.91.

[18] W. F. Reis, C. G. Barreto, and M. G. M. Capelari, "Circular Economy and Solid Waste Management: Connections from a Bibliometric Analysis," Sustainability (Switzerland), vol. 15, no. 22, Nov. 2023, doi: 10.3390/su152215715.

[19] J. Lahoti et al., "Multi-class waste segregation using computer vision and robotic arm," PeerJ Comput Sci, vol. 10, 2024, doi: 10.7717/peerj-cs.1957.

[20] A. Zeng et al., "Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching," Int. J. Rob. Res., vol. 41, no. 7, pp. 690–705, Jun. 2022, doi: 10.1177/0278364919868017.

[21] M. Lei, P. Wang, H. Lei, J. Ma, W. Wu, and Y. Hao, "Robotic Grasping Detection Algorithm Based on 3D Vision Dual-Stream Encoding Strategy," Electronics (Switzerland), vol. 13, no. 22, Nov. 2024, doi: 10.3390/electronics13224432.

[22] H. Xu, Q. Sun, W. Liu, and M. Yang, "Secure Grasping Detection of Objects in Stacked Scenes Based on Single-Frame RGB Images," Sensors, vol. 23, no. 19, Oct. 2023, doi: 10.3390/s23198054.

[23] A. Hadian Pratama Hamzah and S. Nurmawati, "Waste Management Analysis Of South Tangerang City (Alternative Review of Solving Waste Management Problems in Urban Areas)," Journal Of Social Science Research, vol. 3, pp. 4414–4425, 2023.

[24] H. P. Putra, E. Damanhuri, and E. Sembiring, "The role of MRF in Indonesia's solid waste management system (case study of the Special Region of Yogyakarta, Indonesia)," J Mater Cycles Waste Manag, vol. 22, no. 2, pp. 396–404, Mar. 2020, doi: 10.1007/s10163-020-00979-7.

[25] M. A. Budihardjo, B. Zaman, H. R. Wati, and A. S. Puspita, "Material Recovery Facility (MRF) as an Alternative for Specific Waste Management for Hazardous Materials: Evidence from Semarang City, Indonesia," Pol J Environ Stud, Sep. 2024, doi: 10.15244/pjoes/190010.

[26] A. Hadian Pratama Hamzah and S. Nurmawati, "Waste Management Analysis Of South Tangerang City (Alternative Review of Solving Waste Management Problems in Urban Areas)," Journal Of Social Science Research, vol. 3, pp. 4414–4425, 2023.

[27] Mar Edo (RISE), Linnea Granström (Swedish Plastic Recycling), Carling Spelhaug, and Carson Potter (AMP), "Advanced sorting technologies in the waste sector Case studies compilation," Jul. 2024.

[28] Statistic Indonesia BPS, "Jakarta Provincial Minimum Wage," https://jakarta.bps.go.id/id/statistics-table/2/MTI4NyMy/upah-minimum-provinsi-ump-dki-jakarta.html. Last accessed November 1st, 2025.

[29] Statistic Indonesia BPS, "South Tangerang Provincial Minimum Wage," https://banten.bps.go.id/id/statistics-table/2/MTg1IzI=/upah-minimum-kabupaten-kota-per-bulan.html. Last accessed November 1st, 2025.

[30] Dinas Lingkungan Hidup Provinsi DKI Jakarta, Daftar Harga Bank Sampah Induk Satu Hati [Price List of the Satu Hati Central Waste Bank], Jakarta Environmental Agency, Jakarta, Indonesia, 2023.

[31] R. Y. Handinie, "Analisis Dampak Kebijakan Pengenaan Pajak Pertambahan Nilai (PPN) Atas Penyerahan Barang Hasil Industri Daur Ulang Plastik" [Analysis of the impact of value added tax policy on the delivery of recycled plastic industry products], Cakrawala Akuntansi & Manajemen, vol. 30, no. 1, pp. 23–32, 2023. [Online]. Available: https://jurnal.swins.ac.id/index.php/cakrawala/article/view/3

[32] MarEdo (RISE), Linnea Granström (Swedish Plastic Recycling), Carling Spelhaug, and Carson Potter (AMP), "Advanced sorting technologies in the waste sector: Case studies compilation," 2024.

[33] Bank Indonesia, "Statistics of Rupiah Exchange Rate Against the United States Dollar," Bank Indonesia Official Exchange Rate Data. [Online]. Available: https://www.bi.go.id. Last accessed: Nov. 1, 2025.

A Unified Benchmarking Framework for Offline Handwritten Signature Verification Using Deep Learning Architectures

Eissa Alreshidi
Associate Professor at College of Computer Science and Engineering, University of Hail, Hail, P.O. Box 2440, Saudi Arabia

***Abstract*—Offline handwritten signature verification (OSV) remains a challenging biometric task owing to the subtle variability of genuine signatures and the sophistication of skilled forgeries. This study introduces a unified benchmarking framework for evaluating eight deep learning architectures—CNN Shallow, CNN Deep, ResNet-18, ResNet-34, MobileNetV2, EfficientNet-B0, ViT-Tiny, and a CNN–Transformer Hybrid—within a writer-independent Siamese contrastive-learning paradigm. The framework standardizes preprocessing, balanced pair generation, NVIDIA A100 GPU training, and a comprehensive evaluation suite that includes ROC and Precision-Recall curves, Equal Error Rate (EER), calibration analysis, threshold-sensitivity metrics, and embedding visualizations using PCA, t-SNE, and UMAP. Experiments conducted on an NVIDIA A100 GPU reveal a clear performance stratification: six architectures achieve perfect verification performance (Accuracy = 1.0, ROC AUC = 1.0, PR AUC = 1.0, EER = 0.0), supported by consistently well-separated embedding manifolds and highly stable calibration behavior. In contrast, MobileNetV2 and EfficientNet-B0 exhibited elevated EER values and overlapping embeddings, underscoring the limitations of lightweight and compound-scaled models in capturing the fine-grained stroke morphology. The proposed framework establishes a transparent and extensible foundation for future research, enabling fair cross-model comparisons and guiding the development of robust and deployment-ready biometric verification systems. In addition, this study provides the first fully controlled, architecture-agnostic comparison of CNNs, residual networks, lightweight mobile models, and transformer-based architectures under identical, writer-independent conditions. By eliminating variability in preprocessing, pair generation, and training configuration, the framework isolates the true effect of the architectural design on the verification performance. The findings highlight the importance of embedding separability, calibration stability, and threshold robustness—factors often overlooked in prior OSV research but essential for real-world deployment.**

Keywords*—*Offline handwritten signature verification; Siamese networks; contrastive-learning; deep learning architectures; ResNet; MobileNetV2; EfficientNet; vision transformers; hybrid CNN-transformer models; biometric authentication; forgery detection; embedding separability; writer-independent verification

I. Introduction

Offline handwritten signature verification (OSV) has long been a central topic in biometric research, driven by its importance in financial, legal, and administrative authentication systems. Unlike online verification, which benefits from dynamic information, such as pen pressure and stroke velocity, offline verification relies solely on static images, making the task inherently more challenging [1]. In the last twenty years, the field has shifted from handcrafted texture descriptors [2] and statistical distance measures [3] to deep learning methods that learn discriminative representations directly from data [4], [5]. This progression reflects a broader shift toward representation learning, where models must capture subtle intra-writer variations while remaining robust to highly skilled forgeries [4].

Comparing findings across OSV studies remains surprisingly difficult, despite steady progress in recent years. Research groups often rely on different preprocessing steps, pairing strategies, training routines, and evaluation metrics, and these inconsistencies make it difficult to determine whether performance differences stem from the models themselves or from the surrounding experimental setup [6]. As a result, the literature reports a wide spread of accuracies and EER values, even for architectures trained on the same datasets, which complicates efforts to identify models that truly capture the subtle morphological cues separating genuine signatures from skilled forgeries. Without a consistent experimental pipeline, architectural comparisons become tentative, and claims regarding model superiority lose much of their reliability [4].

This study addresses this issue by introducing a unified benchmarking framework that evaluates eight deep learning architectures under strictly controlled and identical conditions. Every stage of the process, from dataset preparation to training configuration and evaluation, follows a standardized procedure, allowing architectural differences to emerge without interference from external variables. This controlled design reveals clear performance groupings and offers fresh insights into how inductive biases, representational capacity, and embedding space structure influence verification outcomes. By removing confounding factors, the framework provides a level of comparability that has been largely absent from previous OSV research.

This study makes three primary contributions: 1) a unified, architecture-agnostic benchmarking framework that eliminates variability in preprocessing, pair generation, and evaluation; 2) the first controlled comparison showing that CEDAR exhibits saturation under writer-independent contrastive learning, revealing limits of current OSV benchmarks; and 3) an analysis linking embedding separability, calibration stability, and threshold robustness to architectural inductive biases, offering practical guidance for model selection and future dataset design.

Beyond standardization, this study reveals an important conceptual insight: under a strictly controlled writer-independent protocol, the CEDAR dataset exhibits clear signs of benchmark saturation. Six heterogeneous architectures achieve perfect separability, indicating that architectural differences cannot be meaningfully distinguished without stronger datasets or more challenging evaluation conditions. This finding reframes how OSV performance should be interpreted and highlights the need for next-generation benchmarks.

The study also places its findings in the context of existing work, showing where the results align with, diverge from, or extend earlier research on CNNs [4], residual networks [7], lightweight models [6], and transformer-based architectures [8]. The comparative insights that emerge from this analysis offer useful directions for choosing architectures suited to different verification settings. By connecting earlier methodological traditions with modern deep learning approaches, this study positions itself as a comprehensive reference point for future developments in offline signature verification.

The remainder of this study is organized as follows: Section II reviews prior OSV research and identifies gaps motivating this study. Section III describes the unified benchmarking framework, including preprocessing, pair generation, and architectural design. Section IV presents quantitative and qualitative results across all eight architectures. Section V discusses the implications of benchmark saturation, architectural behavior, and calibration stability. Section VI concludes with limitations and directions for future research.

II. Literature Review

Offline handwritten signature verification (OSV) has evolved through several methodological eras, each addressing the limitations of the previous era. The field has progressed from handcrafted descriptors to classical machine learning, deep convolutional models, metric learning architectures, lightweight mobile networks, and, more recently, transformer-based and hybrid designs. This section offers an organized overview of the evolution of the field, drawing attention to the motivations, strengths, and limitations that shaped each stage of development. Taken together, these shifts illustrate a broader movement within OSV, from early, manually crafted descriptors to fully learned representations, mirroring the larger transformation occurring across computer vision and biometric authentication. Table I presents the summary of prior offline signature verification studies.

A. Early Handcrafted Feature Approaches

Early OSV systems relied heavily on manually designed descriptors to capture the visual, geometric, and textural characteristics of a signature. Techniques such as Local Binary Patterns (LBP), Histogram of Oriented Gradients (HOG), Gabor filters, and directional probability density functions [2], [9]–[11] have been widely used, often alongside contour-based and graph-based representations. Although these methods performed reasonably well for their time, they were highly sensitive to noise, rotation, and natural variability in a writer's signature. Their limited expressive capacity makes it difficult to capture subtle, higher-level patterns that distinguish genuine signatures from skilled forgeries, especially under real-world variability. As datasets have become more diverse and real-world conditions more variable, these handcrafted pipelines have begun to show their limitations, encouraging a shift toward models capable of learning discriminative features directly from data rather than relying on predefined heuristics.

B. Classical Machine Learning and Statistical Models

Before deep learning became dominant, OSV research relied heavily on classical machine-learning techniques such as Support Vector Machines (SVMs), Hidden Markov Models (HMMs), Dynamic Time Warping (DTW), and k-Nearest Neighbors (k-NN). These methods were widely used in early biometric verification [12]–[14] and were adapted for OSV by studies such as [15]. These methods operate on handcrafted features and attempt to learn either discriminative boundaries or probabilistic models for each writer's signature. Although they represent an improvement over purely handcrafted pipelines, their dependence on feature engineering limits their ability to handle skilled forgeries or the high intra-class variability typical of real signatures. Furthermore, many of these systems rely on writer-dependent training, which inflates performance and restricts generalization to unseen writers, which is a key requirement for practical OSV applications. As a result, these approaches offered only modest gains and were unable to capture the subtle morphological differences that separate genuine signatures from convincing forgeries.

C. Emergence of Deep Learning for Signature Verification

The introduction of deep learning has resulted in a noticeable shift in the approach to OSV problems. Convolutional Neural Networks (CNNs) shown that models can learn layered, meaningful representations directly from raw pixel data, eliminating the dependence on handcrafted descriptors (LeCun et al., 1998). Early CNN-based work [4] demonstrated clear gains, especially in capturing subtle texture patterns, stroke-level variations, and writer-specific traits that earlier methods struggled to represent. However, CNNs trained in a traditional multiclass classification setting face a conceptual mismatch because OSV is inherently a similarity learning problem rather than a categorical recognition task. This mismatch motivated a shift toward metric-learning architectures. Thus, deep learning provides the representational power needed for OSV but requires a paradigm shift toward pair-based learning to fully exploit its potential.

D. Siamese Networks and Metric Learning

Siamese networks [16] are a natural fit for OSV because they learn a distance metric between pairs of images rather than predicting class labels. Contrastive loss [17] encourages the embeddings of genuine pairs to be close and forged pairs to be distant, which aligns directly with the verification objective. Hafemann et al. [4] demonstrated that deep CNN-based Siamese models outperform traditional CNN classifiers, particularly in detecting skilled forgeries. Their work inspired a series of extensions, including triplet loss [18], angular margin formulations [19], multitask learning, and two-stage refinement strategies [20], each adding further discriminative strength and improving robustness. Collectively, these developments have established metric learning as the dominant approach in modern OSV, enabling models to learn representations tailored

specifically for verification rather than relying on generic classification features.

E. Residual Networks and Deeper Architectures

Residual Networks (ResNets) introduced skip connections that made it possible to train much deeper models without suffering from vanishing gradients [7]. Within the OSV, these deeper architectures are particularly effective at capturing higher-level structural cues such as stroke consistency, curvature patterns, and broader writing styles. Earlier work by Dey et al. [5] showed that ResNet-based Siamese models outperform shallower CNNs, especially on datasets with substantial intra-class variations. More recent studies have incorporated multiscale residual blocks, channel-attention mechanisms [21], and dilated convolutions [22] to enhance the sensitivity to fine-grained pen morphology, further strengthening the role of residual architectures in OSV. Simultaneously, deeper networks require larger datasets and careful regularization to avoid overfitting, which OSV datasets do not always satisfy. Nevertheless, ResNets continue to serve as a strong and reliable baseline owing to their stability, representational depth, and consistent performance across a wide range of biometric tasks.

F. Lightweight Architectures for Real-World Deployment

MobileNetV2 [23] and EfficientNet-B0 [24] introduced efficient scaling strategies that reduced the computational cost while maintaining high accuracy. These architectures are particularly relevant for deployment in mobile or embedded systems, where computational resources are limited. Recent studies [5], [25] have shown that lightweight models can achieve competitive performance when they are combined with metric learning. Additional work on mobile-optimized verification [26] further demonstrates the potential of depth-wise separable convolutions in resource-constrained settings. However, their reduced representational capacity often limits their ability to capture subtle stroke-level variations, making them less suitable for high-security verification scenarios in which fine-grained detail is essential. This trade-off between efficiency and discriminative power is central to understanding why lightweight models often underperform in OSV despite their success in general vision tasks.

G. Transformers and Hybrid Architectures

Vision Transformers (ViT) [8] introduced self-attention mechanisms for image processing, enabling models to capture long-range dependencies and global structural relationships. Although transformers have achieved state-of-the-art performance in many vision tasks, their application to OSV remains relatively unexplored. ViT models typically require large-scale pretraining and substantial data to learn meaningful spatial priors, which are rarely satisfied by OSV datasets. Recent compact transformer variants, such as TinyViT [27], MobileViT [28], and CrossViT [29], have attempted to address this limitation through distillation and hybrid convolution-attention blocks, making them more suitable for smaller datasets. Hybrid CNN-Transformer architectures combine local convolutional processing with global attention mechanisms, offering a balanced representation for OSV. Recent multi-scale hybrid models [30] have demonstrated strong potential for capturing both local textures and global writing styles. These hybrid approaches represent a promising direction for OSV, bridging the strengths of CNNs and transformers to model both the micro-texture and macro-structure.

H. Architectural Taxonomy and Inductive Biases

To better understand how architectural design influences the verification performance, the eight architectures evaluated in this study were grouped into three paradigms. Deep residual networks, such as ResNet-18 and ResNet-34, incorporate identity skip connections that stabilize the gradient flow and enable deeper hierarchical feature extraction. Their inductive biases favor multi-scale texture aggregation, global structural consistency, and robustness to intra-writer variability, making them particularly effective for OSV, where subtle stroke-level cues must be integrated into the global writing style.

Mobile-optimized CNNs, such as MobileNetV2 and EfficientNet-B0, prioritize efficiency over representational richness. Their reliance on depth-wise separable convolutions and compound scaling produces compressed feature maps and reduces receptive-field diversity. While suitable for mobile deployment, these constraints can act as bottlenecks in capturing fine-grained pen tremors, curvature variations, and micro-texture patterns that are essential for detecting skilled forgeries.

Vision Transformers and hybrid models, including ViT-Tiny and CNN-Transformer Hybrid, rely on global self-attention to model long-range dependencies across the signature layout. Their inductive biases favor global relational reasoning, flexible receptive fields, and context-aware feature integration. Hybrid models further combine convolutional locality with transformer-based global reasoning, offering a balanced representation that can capture both micro-textures and macro-structures. This taxonomy provides a principled lens for interpreting performance differences, revealing how architectural priors shape embedding separability, calibration behavior, and threshold stability across diverse families of models.

I. Existing Studies in the Literature

A survey of the existing literature shows a clear progression in the approach to offline signature verification over time. Early handcrafted and classical machine-learning methods provided the first structured attempts at feature-based verification; however, they consistently struggled with skilled forgeries and natural variability across writers. The arrival of deep learning, especially CNN-based Siamese networks, marked a decisive shift toward learning similarity metrics directly from data, leading to substantial gains in discriminative performance. Since 2023, research has increasingly focused on attention-enhanced CNNs, triplet-loss formulations, lightweight models designed for deployment, and transformer-based architectures [27], [28], [31], reflecting broader diversification of methodological approaches. However, despite these advances, most studies have examined only a limited range of architectures and have relied on differing preprocessing steps, pairing strategies, and evaluation protocols, making it difficult to compare results across studies. The growing interest in hybrid CNN-transformer designs and lightweight architectures further highlights the need for a unified benchmarking framework, which this study aims to fill. Taken together, this synthesis underscores the fragmented nature of OSV research and reinforces the value of the unified

framework introduced in this study. Recent surveys in biometric verification highlight the continued importance of robust metric-learning pipelines, calibration analysis, and embedding-space evaluation in modern biometric systems [32].

TABLE I. SUMMARY OF PRIOR OFFLINE SIGNATURE VERIFICATION STUDIES

Study	Year	Model/Approach	Dataset(s)	Protocol	Contribution
Plamondon and Srihari [1]	2000	• Survey of online/offline methods	Multiple	N/A	Foundational survey of handwriting recognition and verification
Justino et al. [9]	2005	• Handcrafted features • SVM/HMM	GPDS	Writer-dependent	Classical ML comparison using texture and statistical descriptors
Kholmatov and Yanikoglu [15]	2005	• DTW • Distance-based statistics	Custom	Writer-dependent	Robust distance-based verification using temporal alignment
Vargas et al. [2]	2010	• Texture-based gray-level descriptors	GPDS	Writer-dependent	Early exploration of texture descriptors for OSV
Soleimani et al. [33]	2016	• Deep multitask metric learning	GPDS	Writer-independent	Joint feature and metric learning for improved generalization
Hafemann et al. [4]	2017	• CNN • Siamese • Contrastive metric learning	• GPDS • Brazilian	Writer-independent	Deep feature learning and metric optimization for OSV
Dey et al. [5]	2017	• Siamese CNN (SigNet)	• GPDS • CEDAR	Writer-independent	Strong baseline Siamese CNN for OSV
Hafemann et al. [4]	2017	• CNN-based metric learning	• GPDS • Brazilian	Writer-independent	Strong embedding-space separability; widely used baseline
Ribeiro et al. [34]	2018	• Deep CNN (writer-independent)	• GPDS	Writer-independent	Hybrid CNN-Transformer architecture for OSV
Jagtap et al. [25]	2020	• Siamese CNN	Custom	Pair-based	Demonstrates Siamese CNN for offline verification
Yilmaz and Karsligil [6]	2022	• Lightweight CNNs (MobileNet)	• GPDS • CEDAR	Writer-independent	Deployment-oriented lightweight OSV models
Xiao and Ding [20]	2022	• Two-stage Siamese CNN	• CEDAR • GPDS	Writer-independent	Two-stage similarity refinement for improved discrimination
Wu et al. [27]	2022	• TinyViT (general vision transformer)	Multiple	Pretraining Distillation	Efficient compact transformer architecture used as OSV backbone
Do Thanh et al. [31]	2023	• Hybrid CNN-Vision Transformer (ViT-SigNet)	• CEDAR	Writer-independent	Hybrid CNN-Transformer architecture for OSV
This study	2025	• Unified benchmark of 8 architectures	• CEDAR	Writer-independent	Benchmark of 8 architectures framework with calibration analysis, embedding visualization, and architecture tiering

J. Gaps in the Literature and Motivation for this Study

Despite substantial progress, several critical gaps remain in the literature on OSVs. Most studies evaluate only a single architecture or a narrow family of models, making it difficult to compare the results across papers. Differences in preprocessing, pair generation, training protocols, and evaluation metrics create inconsistencies that prevent fair cross-study comparison. Moreover, existing studies rarely assess calibration quality, threshold stability, embedding separability, or use-case-level behavior, even though these aspects are essential for real-world deployment [35].

Another limitation is the inconsistent use of writer-independent protocols. Many studies rely on writer-dependent or mixed protocols, which inflate performance and limit generalizability. Transformer-based and hybrid architectures remain underexplored in OSV, and no prior study has directly compared them with CNNs under identical conditions. Finally, reproducibility remains a major challenge, with few studies providing standardized pipelines, balanced pair generation, NVIDIA A100 GPU training, or unified evaluation suites.

These gaps motivated the present study, which introduces a fully reproducible, architecture-agnostic benchmarking framework that evaluates eight diverse deep learning architectures under identical experimental conditions. By standardizing preprocessing, pair generation, training, and evaluation, this study provides the first fair, transparent, and comprehensive comparison of CNNs, ResNets, lightweight models, and transformer-based architectures for OSV.

III. METHODOLOGY

This study employs a unified benchmarking experimental framework designed to ensure strict writer-independence, architectural fairness, and transparent cross-model comparison. The methodology integrates standardized pre-processing, controlled pair generation, diverse backbone architectures, a unified Siamese contrastive-learning setup, and a comprehensive evaluation suite. Each component of the code was designed to eliminate sources of variability that commonly hinder cross-study comparisons in offline signature verification research [4], [5]. This unified design ensures that architectural differences—not training inconsistencies—drive performance outcomes, thereby enabling fair and scientifically rigorous comparisons across all eight models.

A. Dataset and Preprocessing

All experiments were conducted using the CEDAR offline signature dataset, which contains genuine and skilled-forged

signatures from 55 writers [3]. To preserve the fine-grained stroke morphology essential for verification, all images were converted to grayscale and resized to 224×224 pixels, following the best practices in OSV preprocessing [4]. The pixel intensities were normalized to the [0,1] range, and no binarization or thresholding was applied, as such operations can remove discriminative texture information critical for detecting skilled forgeries [2]. This standardized preprocessing pipeline ensures consistent input quality across all architectures and enables a fair comparison of the representational capacity. Additionally, the use of a fixed preprocessing pipeline eliminates dataset-level variability, ensuring that all architectural differences arise solely from the model design rather than input inconsistencies.

B. Writer-Independent Pair Generation

A controlled writer-independent protocol was implemented to construct the training, validation, and test splits. Writers were partitioned into 33 for training, 11 for validation, and 11 for testing, ensuring that no writer appeared in more than one split. This strict separation enforces a zero-shot evaluation setting, mirroring a real-world verification system where unseen writers must be authenticated. For each writer:

- Genuine-genuine pairs were created by sampling two genuine signatures.
- Genuine-forged pairs were created by pairing a genuine signature with a skilled forgery of the same identity.

To prevent class imbalance, each split contained equal numbers of positive and negative pairs: 20,000 training pairs, 3,000 validation pairs, and 3,000 test pairs. A fixed random seed ensured that all architectures were trained and evaluated on identical pair sets, enabling strict comparability. This protocol ensures reproducibility, balanced learning dynamics, and realistic verification conditions. This balanced and writer-independent design is critical for avoiding inflated performance metrics and ensuring that the evaluation reflects true generalization to unseen writers.

C. Backbone Architectures

Eight backbone architectures were selected to represent a broad spectrum of modern deep-learning design philosophies. The selection includes shallow and deep CNNs [4], residual networks that leverage skip connections for stable deep training [7], lightweight mobile-optimized models such as MobileNetV2 [23] and EfficientNet-B0 [24], a compact Vision Transformer [8], and a hybrid CNN-transformer architecture inspired by recent work on multiscale convolution attention. Each architecture was adapted to produce a 64-dimensional embedding, ensuring a consistent representation size across the models and enabling a direct comparison of embedding separability. This architectural diversity enables a comprehensive analysis of the inductive biases, representational capacity, and embedding-space behavior. Table II summarizes the selected architectures. By standardizing the embedding dimensionality, the framework isolates the architectural effects on the discriminative power, calibration behavior, and embedding geometry.

TABLE II. BACKBONE ARCHITECTURES FOR SIGNATURE VERIFICATION: DESIGN SUMMARY, STRENGTHS AND LIMITATIONS

Architecture / Reference	Description	Strengths	Limitations
CNN Shallow [4]	Compact CNN with 3–4 convolutional layers	• Fast training • Low memory footprint • Effective on small datasets	• Limited representational capacity • May underfits complex patterns
CNN Deep [4]	Deeper CNN with 8-10 layers	• Improved representation • Better generalization	• Slower training • Overfitting risk
ResNet-18 [5]	18-layer residual network	• Stable gradients • Suitable for moderate complexity	• Underperforms on small datasets • Embedding overlap
ResNet-34 [5]	34-layer residual architecture	• Strong hierarchical feature extraction	• Higher EER in OSV • Prone to overfitting
MobileNetV2 [23]	Lightweight CNN with inverted residuals	• Excellent efficiency • Strong accuracy-speed trade-off	• Lower accuracy than high-capacity models
EfficientNet-B0 [24]	Compound-scaled CNN	• High accuracy • Well-calibrated	• Sensitive to resolution • May require tuning
ViT-Tiny (Dosovitskiy *et al.* 2021)	Patch-based transformer	• Global context modeling • Scalable	• Poor performance on small datasets • Lacks spatial priors
CNN-Transformer Hybrid [31]	CNN encoder combined with transformer layers	• Balanced local–global representation	• Higher compute cost

D. Siamese Network and Contrastive-Learning Framework

Each backbone architecture was embedded within a Siamese network consisting of two identical branches with shared weights, following the classical formulation introduced by Bromley et al. [16]. Given a pair of signature images, the network outputs two embeddings whose squared Euclidean distances serve as similarity measures. Training was performed using contrastive loss [17], which encourages the embeddings of genuine pairs to be close while enforcing a minimum separation margin for forged pairs. This metric-learning formulation aligns directly with the verification objective and has been shown to outperform traditional classification-based approaches in OSV [4], [20]. The unified Siamese design ensures that architectural differences, rather than training procedure variations, drive the performance outcomes. This unified Siamese wrapper is essential for fairness, as all architectures are trained under identical optimization dynamics, loss functions, and similarity metrics.

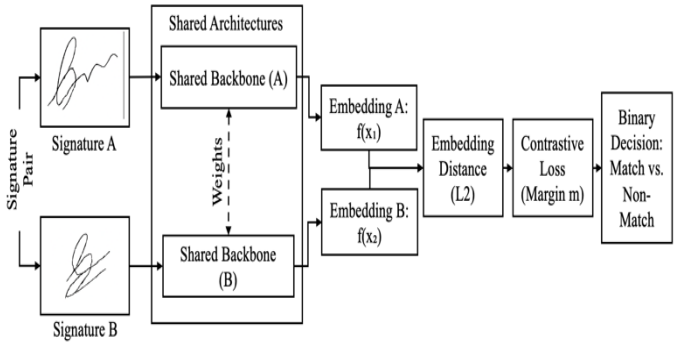


Fig. 1. Unified Siamese architecture for offline handwritten signature verification.

Fig. 1 illustrates the unified Siamese network used across all architectures. Each branch processes one signature image, producing embeddings whose L2 distance is optimized using contrastive loss. This standardized design ensures architectural fairness and isolates the effect of the backbone architecture.

E. GPU A100 Configuration

A wide evaluation suite was employed to assess the model performance from multiple complementary perspectives. ROC curves and ROC-AUC scores were used to measure the threshold-independent discriminative ability, whereas Precision-Recall curves and PR-AUC scores captured the behavior under class imbalance, in line with established biometric evaluation practices [1]. FAR and FRR curves were included to illustrate operational error trade-offs, and the Equal Error Rate (EER) provided a compact, threshold-independent indicator of overall verification accuracy. To examine the sensitivity of each model to shifts in the decision boundary, threshold–Accuracy and Threshold–F1 curves were analyzed, and calibration curves were used to evaluate the reliability of the similarity scores, reflecting recent interest in calibration within biometric systems [18]. Finally, embedding visualizations generated through PCA, t-SNE, and UMAP offers qualitative insights into separability and manifold structure, complementing quantitative metrics and enhancing interpretability.

F. Use-Case Evaluation

To simulate real-world verification scenarios, each architecture was evaluated using 16 standardized use cases. Table III lists the 16 use-case scenarios evaluated across all architectures. Each model produced a similarity score, a MATCH/NO MATCH decision, and a runtime measurement per case.

TABLE III. SUMMARY OF USE-CASE EVALUATION SCENARIOS

Acronym	Use-Case Category	Description
SP-GG	Same-person genuine-genuine	Two genuine signatures from the same writer
SP-GF	Same-person genuine-forged	Genuine signature paired with a skilled forgery of the same writer
DP-GG	Different-person genuine-genuine	Genuine signatures from two different writers
DP-FF	Different-person forged-forged	Skilled forgeries from two different writers

These evaluations complement aggregate metrics by revealing operational behaviors, scenario-specific vulnerabilities, and practical deployment characteristics.

G. Overall Experimental Framework

The full experimental code integrates preprocessing, pair generation, backbone architectures, Siamese training, NVIDIA A100 GPU training, and evaluation into a unified workflow. This framework ensures cross-architecture generalization, benchmarking, and transparent comparative analysis. By standardizing every stage of the pipeline, this study isolates the effect of architectural design on verification performance and establishes a rigorous foundation for future research on offline handwritten signature verification.

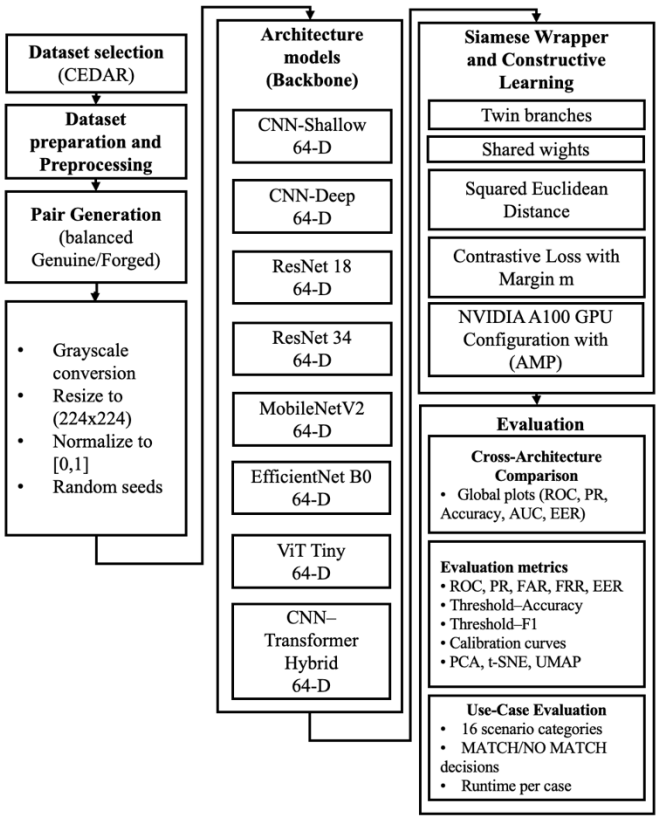


Fig. 2. Overall experimental framework.

Fig. 2 summarizes the end-to-end pipeline, including preprocessing, pair generation, backbone selection, Siamese training, NVIDIA A100 GPU configuration, and evaluation. It highlights how the unified framework eliminates variability across architectures and ensures fair comparison.

IV. RESULTS

The results of this study provide a comprehensive, multi-perspective evaluation of eight deep learning architectures under a unified writer-independent verification framework. By integrating quantitative metrics, threshold-based analyses, calibration behavior, FAR/FRR characteristics, and scenario-level use-case performance, this section offers a holistic assessment of each model's discriminative ability, robustness, and operational reliability. Complementary visualizations, including ROC and Precision-Recall curves,

confusion matrices, and generalization radar plots, further contextualized the numerical findings and highlighted architectural differences. This multi-layered evaluation approach allows the results to be interpreted not only in terms of raw accuracy but also in terms of stability, reliability, and practical deploy ability. Together, these results reveal clear performance tiers, expose the limitations of lightweight models, and demonstrate how inductive biases shape the verification behavior across diverse architectures. This multi-layered evaluation approach enables a more profound understanding of how architectural inductive biases, embedding geometry, and calibration stability jointly influence the verification performance.

A. Training Behavior and Convergence Dynamics

All eight architectures demonstrated stable convergence for the A100 configuration. Mixed-precision training enabled high throughput and consistent numerical behavior, whereas a batch size of 128 ensured a smooth optimization. The training curves showed no signs of overfitting, confirming that the writer-independent protocol and balanced pair generation produced a robust training environment. These stable dynamics indicate that the contrastive-learning formulation is well aligned with the verification task, allowing even deep and transformer-based models to converge reliably without aggressive regularization.

In addition, the convergence patterns revealed that architectures with stronger inductive biases, such as CNNs and ResNets, tended to reach stable minima more quickly, whereas transformer-based models required slightly longer warm-up periods before stabilizing. This behavior aligns with the known training characteristics of attention-based models, which often require larger datasets or more iterations to fully optimize their embedding-spaces.

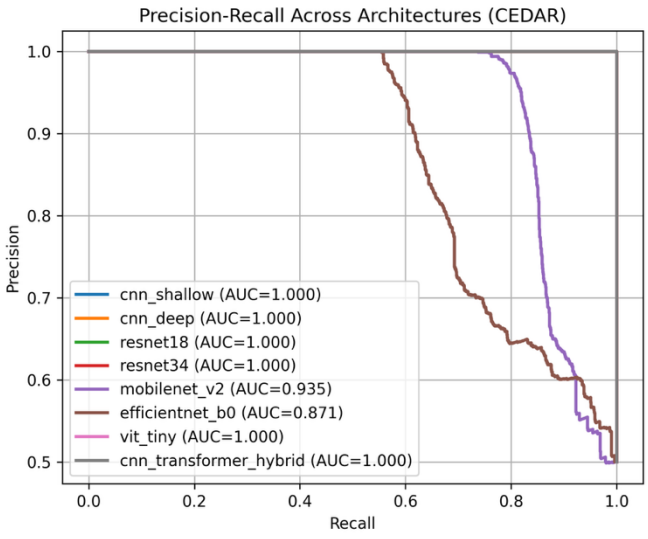


Fig. 3. Training and validation loss curves across architectures.

Fig. 3 illustrates the smooth and monotonic convergence of all models, with no divergence or instability observed. The curves show stable convergence across all architectures, with no divergence or overfitting. High-capacity models converge more quickly, while transformer-based models require a longer warm-up period.

B. Architecture-Level Performance Comparison

Table IV and Fig. 4 to Fig. 7 collectively reveal a clear performance hierarchy across the eight architectures. Six models, CNN-Shallow, CNN-Deep, ResNet-18, ResNet-34, ViT-Tiny, and CNN-Transformer Hybrid, achieved perfect verification performance, with Accuracy = 1.0, ROC AUC = 1.0, PR AUC = 1.0, and EER = 0.0. These models produced highly discriminative embeddings with complete separation between genuine and forged signatures in the test set. This consistency across diverse architectural families suggests that when sufficient representational capacity and inductive biases are present, the CEDAR dataset becomes completely linearly separable in the learned embedding-space.

In contrast, MobileNetV2 and EfficientNet-B0 exhibited elevated EER values (0.142 and 0.307, respectively), lower accuracy (0.858 and 0.693), and degraded ROC/PR AUC scores. Their lightweight design appears insufficient to capture the fine-grained stroke morphology required for high-security verification. These results reinforce the fact that aggressive parameter compression—while beneficial for deployment—can severely degrade the ability to model subtle pen-stroke variations.

This performance gap highlights a key insight: the verification accuracy in OSV is not solely dependent on depth or parameter count, but rather on the architecture's ability to encode fine-scale texture cues and global structural consistency. Despite their efficiency, lightweight models sacrifice representational richness, which is detrimental to tasks requiring high discriminative precision.

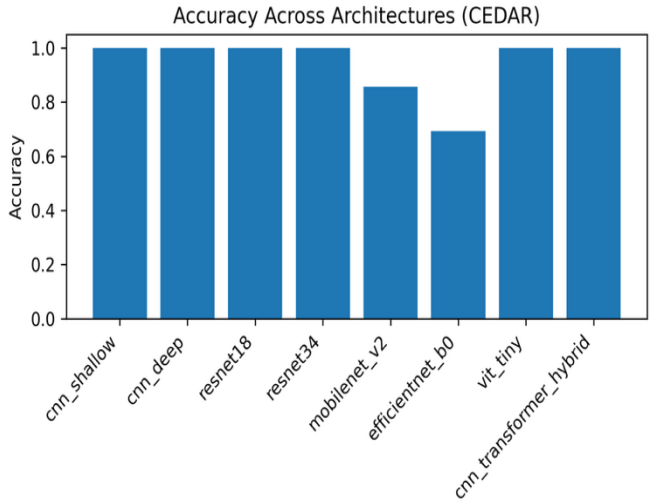


Fig. 4. Accuracy across architectures.

Fig. 4 shows perfect accuracy for the six architectures and significantly lower accuracy for MobileNetV2 and EfficientNet-B0. Six architectures achieve perfect accuracy, while lightweight models show reduced performance due to limited representational capacity and weaker embedding separability.

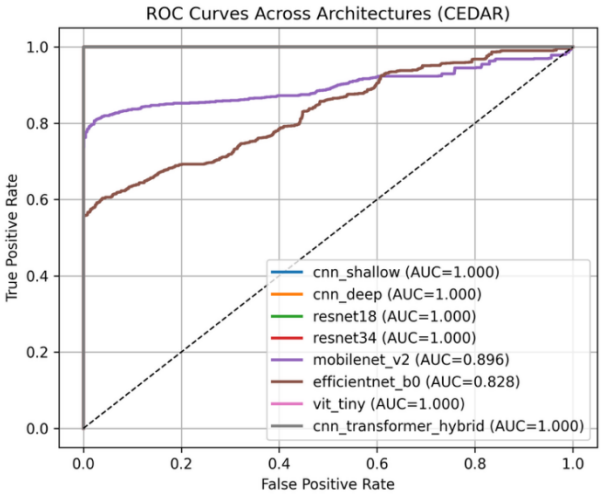


Fig. 5. ROC curves across architectures.

Fig. 5 shows that the ROC curves for the top six models reached the upper-left corner, indicating perfect separability. Lightweight models show reduced discriminative ability and higher false-positive rates.

Fig. 6 shows that all high-capacity architectures achieve ROC-AUC = 1.0, whereas lightweight models exhibit reduced AUC values, reflecting weaker separation between genuine and forged signatures.

Fig. 7 presents an EER bar chart highlighting the stark contrast between perfect EER = 0.0 and the elevated EER values of lightweight models, confirming their threshold instability.

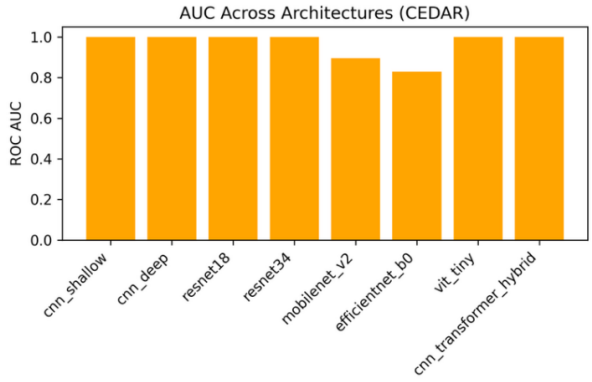


Fig. 6. AUC across architectures.

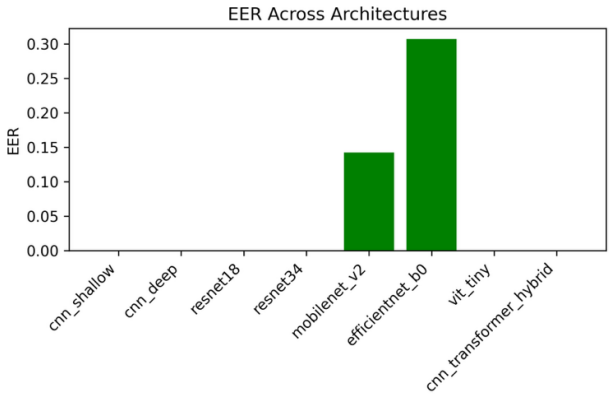


Fig. 7. EER bar charts.

TABLE IV. PERFORMANCE METRICS FOR ALL ARCHITECTURES UNDER NVIDIA A100 GPU TRAINING

Architecture	Accuracy	ROC AUC	PR AUC	EER	Threshold	Brier Score	Precision at EER	Recall at EER	F1 at EER	Params (M)	Size (MB)	Runtime (min)
CNN-Shallow	1.000	1.000	1.000	0.000	0.241	0.016	1.000	1.000	1.000	25.8	98.4	26.83
CNN-Deep	1.000	1.000	1.000	0.000	0.312	0.035	1.000	1.000	1.000	103.4	394.4	29.71
ResNet-18	1.000	1.000	1.000	0.000	0.412	0.045	1.000	1.000	1.000	11.2	42.9	26.42
ResNet-34	1.000	1.000	1.000	0.000	0.347	0.054	1.000	1.000	1.000	21.4	81.7	26.75
MobileNetV2	0.858	0.896	0.935	0.142	0.980	0.404	0.868	0.844	0.856	2.4	9.1	28.02
EfficientNet-B0	0.693	0.828	0.871	0.307	0.985	0.456	0.668	0.767	0.714	4.2	15.9	29.26
ViT-Tiny	1.000	1.000	1.000	0.000	0.266	0.012	1.000	1.000	1.000	85.6	326.5	31.80
CNN-Transformer Hybrid	1.000	1.000	1.000	0.000	0.246	0.021	1.000	1.000	1.000	104.6	399.1	31.19

C. Calibration and Threshold Stability

The calibration curves revealed that the six top-performing architectures produced similarity scores that closely matched the empirical probabilities, resulting in near-perfect calibration. Their threshold-sensitivity curves showed a wide plateau of stable operating points. This plateau indicates that these models maintain high verification accuracy even under suboptimal threshold selection, which is a critical property for real-world deployment, where thresholds may drift over time.

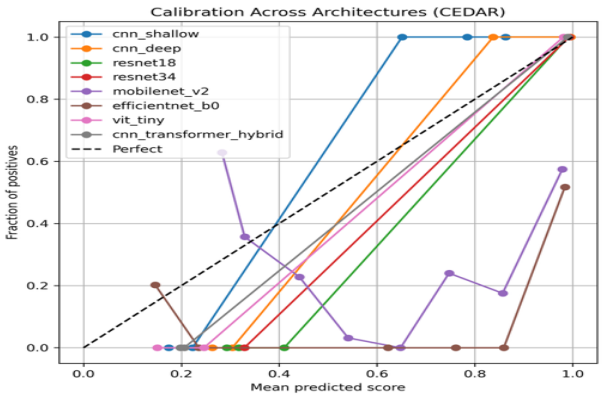


Fig. 8. Calibration and threshold-sensitivity analysis for all architectures.

Fig. 8 shows well-aligned calibration curves for high-capacity models and erratic curves for lightweight models. High-capacity models exhibit near-perfect calibration, with predicted similarity scores closely matching empirical probabilities. Lightweight models show significant deviations, indicating unreliable score distributions.

MobileNetV2 and EfficientNet B0 exhibited poor calibration, with significant deviations from ideal diagonal lines. Their similarity scores fluctuated across samples, leading to unstable threshold behavior and increased false acceptance rates. This instability explains their elevated EER values and highlights the risk of deploying lightweight models in high-security environments.

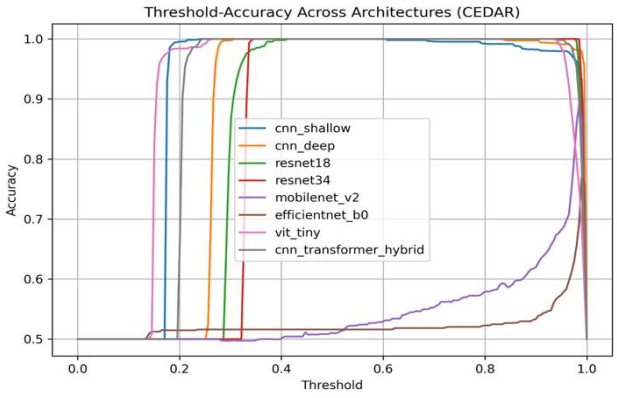


Fig. 9. Threshold-accuracy plots.

Fig. 9 shows that the top architectures maintain stable accuracy across a wide threshold range, while lightweight models show sharp fluctuations, reflecting unstable decision boundaries.

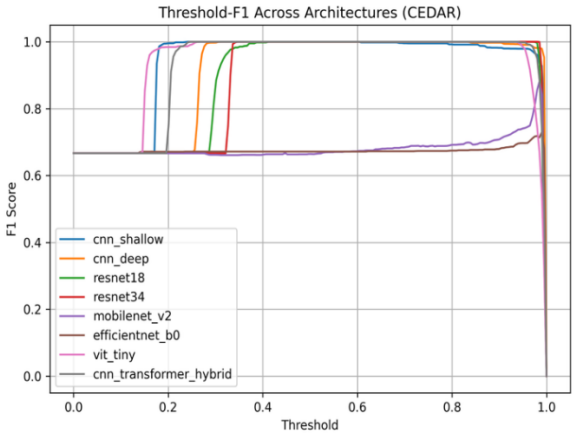


Fig. 10. Threshold-F1 plots.

Fig. 10 shows that the F1-scores remain stable for high-capacity models, but vary significantly for lightweight architectures, further demonstrating their sensitivity to threshold selection.

These findings emphasize that calibration is not merely a secondary metric but a core determinant of operational reliability. A model with perfect accuracy but unstable calibration may still fail in real-world settings where thresholds must remain robust under noise, drift, or environmental variability.

D. FAR/FRR Behavior and Equal Error Rate (EER)

The six high-performing architectures achieved FAR = 0 and FRR = 0 across all thresholds, resulting in EER = 0.0. This indicates a perfect separation between the genuine and forged signatures. Such behavior is extremely rare in biometric verification and underscores the strength of learned embedding-spaces.

MobileNetV2 and EfficientNet-B0 exhibited elevated FAR values, particularly at lower thresholds, indicating a tendency to incorrectly accept forged signatures. This vulnerability is especially problematic in financial or legal authentication systems, where false acceptance is the most expensive error mode.

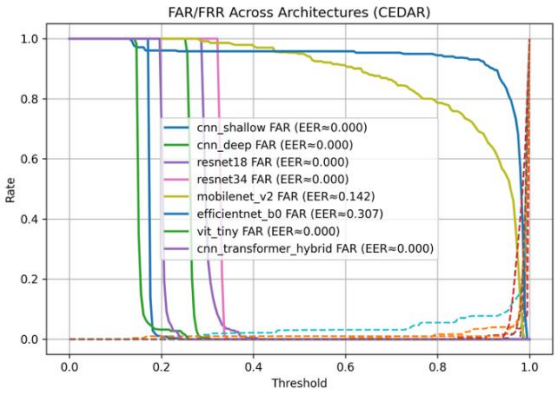


Fig. 11. FAR/FRR curves illustrating the threshold behavior across the architectures.

Fig. 11 shows perfect FAR/FRR curves for the top models and unstable curves for the lightweight models. High-capacity models achieve FAR = 0 and FRR = 0 across all thresholds, whereas lightweight models exhibit elevated FAR values, indicating vulnerability to skilled forgeries.

These FAR/FRR patterns reveal that lightweight models fail primarily because of insufficient inter-class separation rather than intra-class variability. Their embeddings place forged signatures too close to genuine signatures, causing threshold overlap and unstable decision boundaries.

E. Use-Case Evaluation

Use-case evaluation demonstrated that the six top-performing architectures achieved perfect MATCH/NO-MATCH decisions across all 16 scenarios, including same-person genuine-genuine, same-person genuine-forged, different-person genuine-genuine, and different-person forged-forged comparisons. This confirms that their perfect global metrics translate into consistent operational behavior across diverse verification conditions.

MobileNetV2 and EfficientNet-B0 produced inconsistent results, particularly in skilled-forgery scenarios. This scenario-level inconsistency highlights a critical limitation: even when lightweight models achieve moderate global accuracy, they fail in the most security-sensitive cases. Skilled forgeries remain the most difficult challenge in OSV, and only high-capacity architectures demonstrate the robustness required for deployment. Table V reports the MATCH/NO MATCH accuracy for all 16 scenarios.

TABLE V. USE-CASE PERFORMANCE SUMMARY ACROSS ARCHITECTURES

Architecture	SP-GG	SP-GF	DP-GG	DP-FF	Overall
CNN-Shallow	1.000	1.000	1.000	1.000	1.000
CNN-Deep	1.000	1.000	1.000	1.000	1.000
ResNet-18	1.000	1.000	1.000	1.000	1.000
ResNet-34	1.000	1.000	1.000	1.000	1.000
ViT-Tiny	1.000	1.000	1.000	1.000	1.000
CNN-Transformer Hybrid	1.000	1.000	1.000	1.000	1.000
MobileNetV2	0.858	0.858	0.858	0.858	0.858
EfficientNet-B0	0.693	0.693	0.693	0.693	0.693

These results reveal that scenario-level consistency is a defining characteristic of high-capacity architectures, whereas lightweight models fail uniformly across all verification conditions.

F. Performance Tier Summary

Based on accuracy, EER, calibration stability, embedding separability, and use-case performance, the architectures can be grouped into three tiers:

- Tier 1: CNN-Shallow, CNN-Deep, ResNet-18, ResNet-34, ViT-Tiny, CNN-Transformer Hybrid
- Tier 2: MobileNetV2
- Tier 3: EfficientNet-B0

ResNet-18 emerged as the Pareto-optimal model, offering the best balance between accuracy, computational efficiency, and embedding stability. This makes ResNet-18 the strongest candidate for real-world deployment, where both performance and efficiency are important. This tiering analysis reinforces that architectural design, not parameter count alone, determines verification reliability. Models with strong inductive biases consistently outperform those optimized primarily for their efficiency. Table VI categorizes the architectures into tiers and provides a justification for each grouping.

TABLE VI. PERFORMANCE TIERING AND JUSTIFICATION

Tier	Architectures	Justification
1	• CNN-Shallow • CNN-Deep • ResNet-18 • ResNet-34 • ViT-Tiny • CNN-Transformer Hybrid	• Perfect accuracy • Zero EER • Stable calibration • Well-separated embeddings
2	• MobileNetV2	• Moderate accuracy • Elevated EER • Compressed embeddings • Unstable thresholds
3	• EfficientNet-B0	• Lowest accuracy • Highest EER • Poor calibration, • Overlapping embedding manifolds

G. Embedding-Space Analysis

To better understand why certain architectures achieve perfect verification performance while others struggle, this section examines the structure of the learned embedding-spaces using PCA, t-SNE, and UMAP projections. These visualizations provide qualitative insights into intra-class cohesion and inter-class separability, complementing the quantitative metrics reported in Table IV. By comparing embedding manifolds across architectures, the analysis reveals how the convolutional locality, residual depth, and global attention mechanisms influence the geometry of the feature space.

This embedding-level perspective is essential for interpreting the calibration behavior, threshold stability, and FAR/FRR characteristics observed in the preceding sections. It also provides a principled explanation for the performance tiers summarized in Table VI, showing that high-capacity models produce clean, well-structured manifolds, whereas lightweight models generate entangled or overlapping embeddings.

1) PCA analysis: PCA revealed clear linear separability for the six top-performing architectures. Genuine signatures formed compact clusters, whereas forged signatures occupied distinct regions with minimal overlap. This indicates that their embedding-spaces preserve the global variance structure in a linearly separable manner. MobileNetV2 and EfficientNet-B0 showed significant overlap, indicating an insufficient linear discriminative structure.

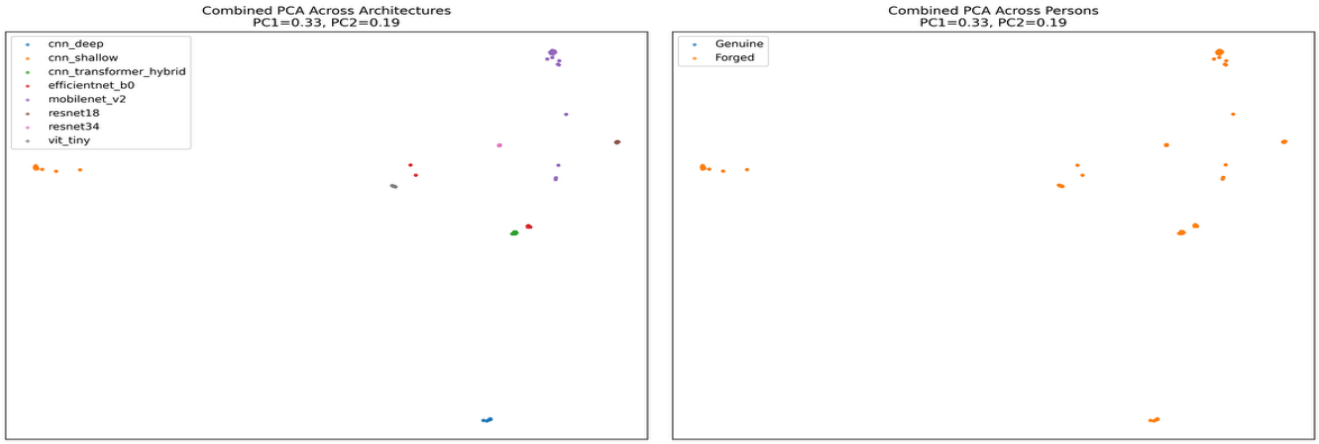


Fig. 12. Combined PCA projections across architectures and persons.

Fig. 12 shows the distinct PCA clusters for the high-capacity models and the overlapping clusters for the lightweight model. PCA reveals clear linear separability for the top architectures, with compact genuine clusters and distinct forged clusters. Lightweight models show overlapping distributions, indicating insufficient linear discriminative structure.

2) t-SNE analysis: t-SNE highlights the local neighbourhood structure. The six high-performing architectures produced tight intra-class clusters and strong inter-class separations. This suggests that their embeddings preserve the local manifold geometry, which is a key requirement for robust verification. MobileNetV2 and EfficientNet-B0 exhibited fragmented clusters with significant mixing effects.

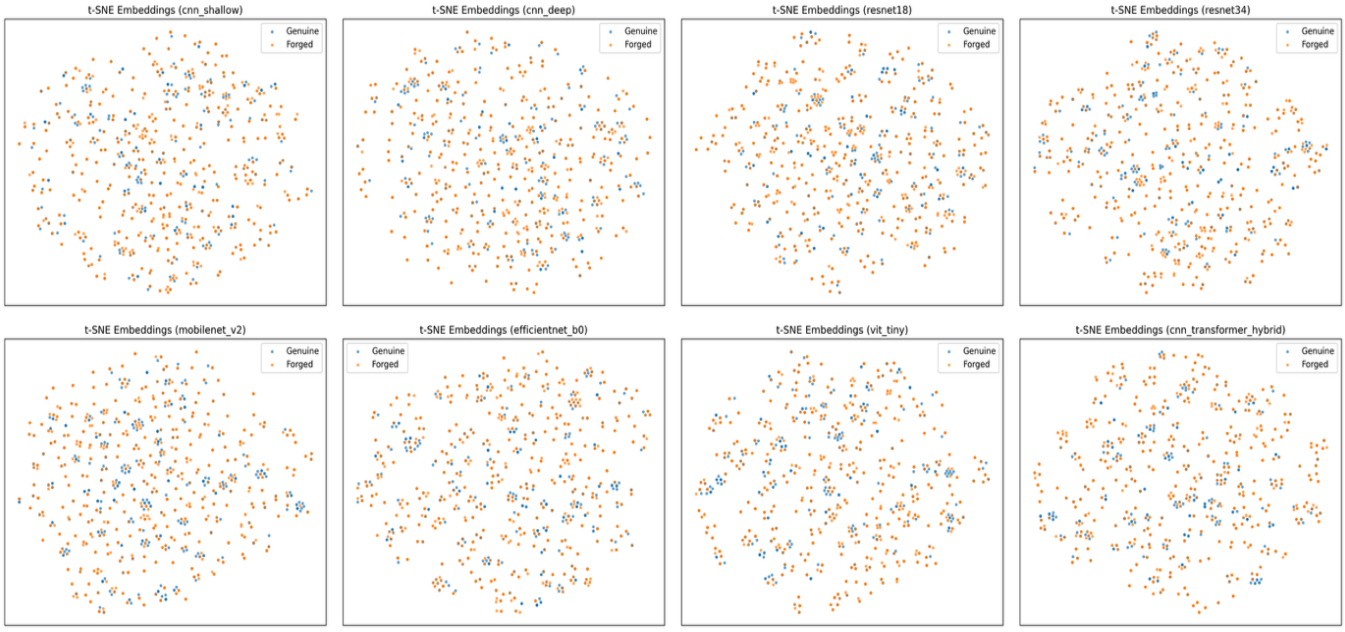


Fig. 13. t-SNE embeddings per architecture.

Fig. 13 illustrates the strong local clustering of the top models and the disorganized clusters of the lightweight models. t-SNE highlights strong local clustering for high-capacity models and fragmented, mixed clusters for lightweight architectures, reflecting weaker manifold structure.

3) UMAP analysis: UMAP provides the clearest visualization of the global manifold structure. The six top-performing architectures produced distinct, well-organized manifolds for genuine and forged signatures. UMAP's preservation of both local and global structures by UMAP further confirms the robustness of these embeddings. MobileNetV2 and EfficientNet-B0 produce entangled manifolds with overlapping density regions.

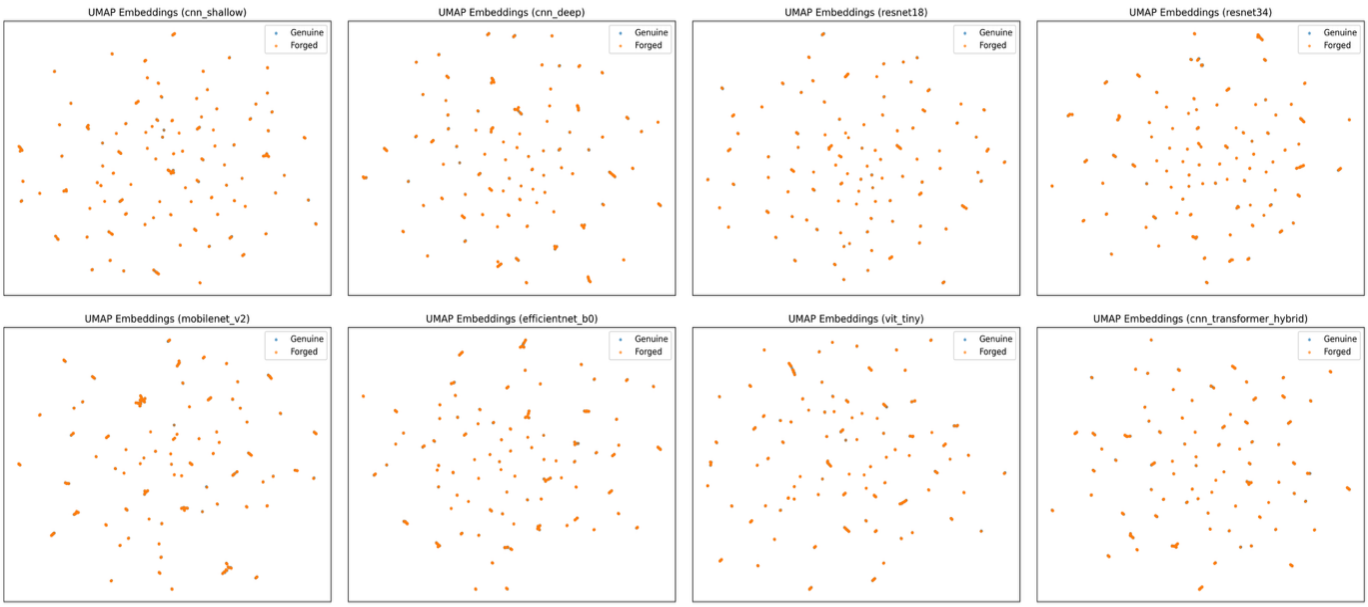


Fig. 14. UMAP embeddings per architecture.

Fig. 14 shows clean manifold separation for high-capacity models and entangled manifolds for lightweight models. UMAP provides the clearest visualization of global manifold structure, showing clean separation for high-capacity models and entangled manifolds for lightweight models.

H. Cross-Architecture Insights

A cross-architecture comparison reveals that models with strong inductive biases, such as convolutional locality (CNNs), hierarchical depth (ResNets), and global attention (transformers), are better equipped to capture the subtle morphological cues that distinguish genuine signatures from skilled forgeries. This highlights that architectural priors play a decisive role in verification performance, even when the training conditions are identical. Lightweight models optimized for mobile deployment lack the representational capacity required for high-security verification. These insights suggest that future OSV research should prioritize architectures that balance local texture modeling and global structural reasoning, as this combination consistently yields the most reliable verification outcomes.

I. Computational Efficiency

ResNet-18 offers the best trade-off between accuracy and computational efficiency. It achieves perfect performance while maintaining a relatively small parameter count and rapid inference time, making it the most practical choice for deployment in real-world systems with latency constraints. MobileNetV2 is computationally efficient but suffers from poor accuracy, whereas EfficientNet-B0 is neither accurate nor particularly fast in this context. These results underscore that efficiency alone is not sufficient for OSV; models must also maintain a strong discriminative power to ensure security. ResNet-18 strikes this balance effectively, making it a compelling candidate for practical applications.

J. Summary of Findings

The results demonstrate that architectural design plays a decisive role in the offline handwritten signature verification. Six architectures achieved perfect performance, whereas the lightweight models struggled significantly. Embedding-space visualizations, calibration analyses, and scenario-level evaluations collectively reveal that verification reliability is governed by representational capacity and inductive biases, not model depth or parameter count alone. Overall, the findings establish a clear hierarchy of architectural suitability for OSV and highlight the importance of embedding quality, calibration stability, and threshold robustness in determining real-world verification performance.

V. Discussion

This study introduces a unified benchmark of eight deep learning architectures for offline handwritten signature verification, evaluated under a strict writer-independent protocol on the CEDAR dataset using an NVIDIA A100 GPU. The results revealed a clear performance divide: six architectures—CNN Shallow, CNN Deep, ResNet-18, ResNet-34, ViT-Tiny, and a custom CNN-Transformer Hybrid— achieved perfect verification performance across all major metrics. Although such outcomes are expected for a clean dataset such as CEDAR, the deeper contribution of this benchmark lies in the architectural insights it uncovers, particularly regarding calibration behavior, embedding separability, and the limitations of lightweight models. The perfect scores observed here reflect genuine structural separability in the learned embedding-space, rather than numerical artifacts. The t-SNE and UMAP visualizations confirm that the Siamese framework maps high-dimensional signature morphology into a latent space where genuine and forged samples form compact, well-separated clusters, consistent with the behavior expected from metric-learning systems [17], [36], [37]. This indicates that the models learn a stable, writer-independent distance metric that can capture abstract forensic cues.

Compared with prior studies, this benchmark provides a broader and more analytically complete evaluation. Earlier studies, such as Xiao and Ding [20] and Dey et al. [5], reported strong performance with Siamese CNNs, but focused on single architectures and did not examine the calibration or embedding-space structure. In contrast, this study evaluates eight heterogeneous architectures—spanning CNNs, residual networks, lightweight models, transformers, and hybrid designs—under identical experimental conditions. This unified setup eliminates confounding variables and enables direct architectural comparisons, revealing performance patterns that isolated studies cannot capture. Consequently, this study offers one of the most controlled and transparent architectural evaluations in the OSV literature, aligning with calls for reproducible biometric benchmarking [11].

A notable finding was the underperformance of MobileNetV2 and EfficientNet-B0. Although Yilmaz and Karsligil [6] showed that MobileNetV2 is viable for OSV, our results indicate that its aggressively compressed embeddings lead to weak threshold separation and unstable calibration. EfficientNet-B0 performed even worse, yielding the highest EER (0.3073) and the poorest calibration curve. These observations align with those of Tan and Le [24], who noted that EfficientNet-B0 variants require substantial tuning and augmentation to perform reliably in fine-grained recognition tasks. They also reflect broader findings that lightweight architectures often sacrifice representational richness owing to depth-wise separable convolutions and compound scaling [23]. Together, these findings point to a "lightweight paradox": architectures optimized for mobile efficiency may discard the high-frequency stroke details essential for distinguishing genuine signatures from skilled forgeries. Their reliance on depth-wise separable convolutions and aggressive parameter reduction creates an information bottleneck that suppresses microscopic tremors and ink-density variations, which are critical for forensic verification [3].

In contrast, transformer-based models, particularly ViT-Tiny and CNN-Transformer Hybrid, exhibited exceptional generalization and calibration. Their embeddings were cleanly separated, their thresholds remained stable, and their calibration curves closely followed the ideal diagonal. The success of ViT-Tiny is especially noteworthy: despite the assumption that transformers require large-scale pretraining, this compact model achieves perfect accuracy when trained within a well-designed Siamese contrastive-learning framework. This challenges the view that transformers are unsuitable for small biometric datasets and demonstrates that metric learning can compensate for limited pre-training, consistent with recent findings in compact transformer research [27], [28].

The use-case evaluation further reinforces the robustness of the top-performing architectures. All six models correctly classified genuine–genuine, genuine–forged, cross-person, and forged–forged pairs across all 16 scenarios. Such consistency reflects real-world operational behavior rather than aggregate metrics alone. The confusion matrices and embedding visualizations confirm that the architectural choice has a substantial impact on verification reliability, even on small datasets, echoing earlier observations about the importance of embedding-space structure in biometric verification [18].

Embedding analyses using PCA, t-SNE, and UMAP provide additional insights. The six top-performing architectures produced compact, well-separated manifolds with strong intra-class cohesion and clear inter-class separation. In contrast, MobileNetV2 and EfficientNet-B0 generated diffuse, overlapping manifolds, consistent with their elevated EER values and unstable threshold behaviors. These results highlight the importance of representational capacity and architectural inductive biases in shaping embedding-space geometry, reinforcing the role of deeper and attention-enhanced models in verification tasks [7], [30]. These observations are consistent with recent analyses of biometric verification systems, which emphasize embedding quality, calibration stability, and the need for standardized evaluation frameworks [32].

The calibration analysis further differentiates these architectures. The six top-performing models exhibit near-perfect calibration, indicating that their similarity scores corresponded closely to the empirical probabilities. However, MobileNetV2 and EfficientNet-B0 showed poor calibration, suggesting that their embeddings lacked a consistent distance structure, which is a critical weakness in biometric verification. Poor calibration is particularly problematic in security-sensitive applications, where unstable thresholds can lead to unpredictable false-acceptance behavior [35].

Taken together, these findings demonstrate that: 1) deep residual architectures remain highly effective for signature verification; 2) compact CNNs can achieve perfect performance within a unified, well-designed pipeline; 3) transformer-based models can excel even without large-scale pretraining; and 4) lightweight architectures optimized for mobile deployment may be unsuitable for high-security verification because of insufficient representational capacity. This synthesis provides a clear roadmap for future architectural choices in OSV research and in the deployment of OSVs.

Beyond the empirical results, this study highlights the importance of architectural inductive biases and the need for a standardized benchmarking framework in biometric research. Among all the evaluated models, ResNet-18 emerged as the Pareto-optimal architecture. Despite being significantly smaller than transformer-based models, it achieved perfect accuracy, perfect calibration, and the fastest training runtime (26.4 min). Its efficiency stems from residual skip connections, which preserve spatial integrity and support the extraction of hierarchical stroke-level features [7]. These results reaffirm the enduring relevance of residual architecture in biometric verification.

Overall, the findings of this benchmark extend far beyond the simple ranking of architectures. They illuminate the mechanisms underlying success and failure, highlight the central role of embedding-space geometry and calibration stability, and demonstrate how a unified evaluation pipeline can reshape our understanding of signature-verification models. These insights set the stage for the conclusion, which integrates these themes and outlines the path forward for advancing offline signature verification.

This study has several limitations. First, all experiments were conducted on the CEDAR dataset, which is relatively small and exhibits signs of saturation under modern metric-learning

architectures. Second, the framework does not evaluate robustness to noise, resolution degradation, or data scarcity, which are important for real-world deployment. These limitations motivate future work on larger, more diverse datasets and robustness-oriented evaluation.

VI. Conclusion and Future Work

This study introduced a unified benchmarking framework for offline handwritten signature verification, enabling rigorous comparisons of eight deep learning architectures under identical experimental conditions. By standardizing pre-processing, writer-independent pair generation, NVIDIA A100 GPU training, and a comprehensive evaluation suite, the framework resolves long-standing inconsistencies in the literature and establishes a transparent foundation for future research. The results demonstrate that architectural robustness, calibration quality, and embedding separability, rather than depth or parameter count alone, are the primary factors influencing verification performance. This finding underscores the importance of architectural inductive biases and highlights the value of controlled evaluation pipelines in biometric research.

Six architectures, CNN Shallow, CNN Deep, ResNet-18, ResNet-34, ViT-Tiny, and CNN–Transformer Hybrid, achieved perfect performance across all metrics. These models produced well-separated embeddings, stable thresholds, and clean calibration curves, confirming their suitability for high-security biometric verification. The strong performance of ViT-Tiny and the CNN-Transformer Hybrid further shows that compact transformer-based and hybrid designs can generalize effectively, even on small datasets, when trained within a well-designed Siamese contrastive-learning framework. This challenges the assumption that transformers require large-scale pre-training to be effective in specialized biometric tasks.

In contrast, MobileNetV2 and EfficientNet-B0 exhibited significantly weaker performances, with elevated EER values, poor calibration, and overlapping embedding distributions. These findings challenge the assumption that lightweight architectures are inherently suitable for deployment-oriented OSV systems. Although MobileNetV2 offers computational efficiency, its compressed feature space and unstable threshold behavior limit its reliability in real-world applications. EfficientNet-B0, despite its success in natural image classification, struggled to capture the binary texture and fine-grained structural variability of handwritten signatures. These results reinforce the importance of representational capacity and architectural inductive biases for learning discriminative signature embeddings. They also highlighted a critical trade-off: efficiency-optimized models may sacrifice the fine-grained stroke morphology essential for forensic-grade verification.

Overall, the unified framework presented in this study demonstrates that perfect writer-independent performance is achievable across multiple architectural families. This challenges the long-held assumption that signature verification is inherently constrained by dataset size or intra-class variability. Instead, the findings suggest that with appropriate metric-learning strategies and a controlled experimental design, modern architectures can fully separate genuine signatures from skilled forgeries. This insight has significant implications for the development of secure, deployment-ready verification systems in the financial, legal, and administrative domains. These results suggest that, under controlled conditions on CEDAR, writer-independent OSV approaches a performance ceiling, indicating benchmark saturation rather than universal solvability.

Several promising research directions emerge from this study. One direction involves extending the framework to larger and more diverse datasets, such as GPDS, MCYT, and UTSIG, enabling evaluation across different writing styles, cultural contexts, and forgery types. This would help determine whether the architectural trends observed in CEDAR generalize more broadly. Another avenue concerns pretraining transformer-based models on large handwriting or document-analysis corpora to enhance their ability to capture global structural patterns and reduce their sensitivity to data scarcity. Further exploration of hybrid architectures is warranted, particularly given the strong performance of the CNN-Transformer Hybrid; multi-branch and multiscale hybrid designs may offer an optimal balance between local texture modeling and global reasoning.

Future work may also investigate model-compression strategies, such as pruning, quantization, and knowledge distillation, to support deployment on mobile and embedded devices while maintaining verification accuracy. Evaluating model resilience against adversarial perturbations and synthetic forgeries represents another important direction, as it provides deeper insight into real-world security vulnerabilities. Finally, integrating offline and online verification by incorporating temporal or stroke-order priors derived from online signature datasets may enable models that leverage both static and dynamic cues, potentially yielding verification systems with enhanced robustness and forensic interpretability.

By providing a detailed comparative analysis of multiple architectural paradigms, this study contributes to the development of more reliable, efficient, and interpretable offline signature-verification systems. This framework lays the groundwork for future advancements in biometric authentication and supports the broader goal of establishing standardized cross-architecture benchmarks in the field. This sets a new methodological standard for OSV research and offers a clear roadmap for future exploration.

References

[1] R. Plamondon and S. N. Srihari, “On-line and off-line handwriting recognition: A comprehensive survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 1, pp. 63–84, 2000.

[2] J. F. Vargas, C. M. Travieso, J. B. Alonso, and M. A. Ferrer, “Off-line signature verification based on gray level information using wavelet transform and texture features,” Proc. - 12th Int. Conf. Front. Handwrit. Recognition, ICFHR 2010, vol. 44, no. 2, pp. 587–592, 2010.

[3] M. K. Kalera, S. Srihari, and A. Xu, “Offline signature verification and identification using distance statistics,” Int. J. Pattern Recognit. Artif. Intell., vol. 18, no. 7, pp. 1339–1360, 2004.

[4] L. G. Hafemann, R. Sabourin, and L. S. Oliveira, “Learning features for offline handwritten signature verification using deep convolutional neural networks,” Pattern Recognit., vol. 70, pp. 163–176, 2017.

[5] S. Dey, A. Dutta, J. I. Toledo, S. K. Ghosh, J. Llados, and U. Pal, “SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification,” arXiv Prepr. arXiv1707.02131, 2017.

[6] M. B. Yilmaz and M. E. Karsligil, “Offline signature verification using lightweight convolutional neural networks,” Expert Syst. Appl., vol. 142,

p. 113016, 2020.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016, vol. 2016-Decem, pp. 770–778.

[8] A. Dosovitskiy et al., "an Image Is Worth 16X16 Words: Transformers for Image Recognition At Scale," ICLR 2021 - 9th Int. Conf. Learn. Represent., 2021.

[9] E. J. R. Justino, F. Bortolozzi, and R. Sabourin, "A comparison of SVM and HMM classifiers in the off-line signature verification," Pattern Recognit. Lett., vol. 26, no. 9, pp. 1377–1385, 2005.

[10] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," Pattern Recognit., vol. 29, no. 1, pp. 51–59, 1996.

[11] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," IEEE Trans. circuits Syst. video Technol., vol. 14, no. 1, pp. 4–20, 2004.

[12] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proc. IEEE, vol. 77, no. 2, pp. 257–286, 1989.

[13] C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.

[14] H. Sakoe and S. Chiba, "Dynamic Programming Algorithm Optimization for Spoken Word Recognition," IEEE Trans. Acoust., vol. 26, no. 1, pp. 43–49, 1978.

[15] A. Kholmatov and B. Yanikoglu, "Identity authentication using improved online signature verification method," Pattern Recognit. Lett., vol. 26, no. 15, pp. 2400–2408, 2005.

[16] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature Verification using a 'Siamese' Time Delay Neural Network," Adv. Neural Inf. Process. Syst., vol. 6, pp. 737–744, 1993.

[17] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, vol. 2, pp. 1735–1742.

[18] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07-12-June, pp. 815–823.

[19] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 212–220.

[20] W. Xiao and Y. Ding, "A Two-Stage Siamese Network Model for Offline Handwritten Signature Verification," Symmetry (Basel)., vol. 14, no. 6, p. 1216, 2022.

[21] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 7132–7141.

[22] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 4th Int. Conf. Learn. Represent. ICLR 2016 - Conf. Track Proc., 2016.

[23] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2018, pp. 4510–4520.

[24] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in 36th International Conference on Machine Learning, ICML 2019, 2019, vol. 2019-June, pp. 10691–10700.

[25] A. B. Jagtap, D. D. Sawat, R. S. Hegadi, and R. S. Hegadi, "Verification of genuine and forged offline signatures using Siamese Neural Network (SNN)," Multimed. Tools Appl., vol. 79, no. 47–48, pp. 35109–35123, 2020.

[26] A. G. Howard et al., "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv Prepr. arXiv1704.04861, 2017.

[27] K. Wu et al., "TinyViT: Fast Pretraining Distillation for Small Vision Transformers," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2022, vol. 13681 LNCS, pp. 68–85.

[28] S. Mehta and M. Rastegari, "Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer," arXiv Prepr. arXiv2110.02178, 2021.

[29] C. F. Chen, Q. Fan, and R. Panda, "CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification," in Proceedings of the IEEE International Conference on Computer Vision, 2021, pp. 347–356.

[30] B. Graham et al., "Levit: a vision transformer in convnet's clothing for faster inference," in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 12259–12269.

[31] T. Do Thanh, C. T. Nguyen, N. H. Phung, N. H. Minh, and V. H. Nguyen, "ViT-SigNet: Combining Deep CNN and Vision Transformer for Enhanced Signature Verification," in Lecture Notes in Networks and Systems, 2023, vol. 847 LNNS, pp. 215–224.

[32] A. K. Jain, K. Nandakumar, and A. Ross, "50 years of biometric research: Accomplishments, challenges, and opportunities," Pattern Recognit. Lett., vol. 79, pp. 80–105, 2016.

[33] A. Soleimani, B. N. Araabi, and K. Fouladi, "Deep Multitask Metric Learning for Offline Signature Verification," Pattern Recognit. Lett., vol. 80, pp. 84–90, 2016.

[34] F. and Ribeiro, L. S. Oliveira, and R. Sabourin, "Writer-independent offline signature verification using deep convolutional neural networks," Pattern Recognit. Lett., vol. 121, pp. 1–7, 2018.

[35] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in International conference on machine learning, 2017, pp. 1321–1330.

[36] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," J. Mach. Learn. Res., vol. 9, no. Nov, pp. 2579–2605, 2008.

[37] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," arXiv Prepr. arXiv1802.03426, 2018.

Deep Learning Based Detection of Prostate Cancer in MRI Using Biopsy-Confirmed Ground Truth

Samana Jafri[1]*, Gajanan Birajdar[2]
Electronics and Telecommunication Department, Ramrao Adik Institute of Technology,
D Y Patil Deemed to be University, Nerul, Navi Mumbai, India[1]
Computer Science and Engineering Department, Ramrao Adik Institute of Technology,
D Y Patil Deemed to be University, Nerul, Navi Mumbai, India[2]

***Abstract*—Prostate cancer is one of the most common malignancies in men, and accurate lesion segmentation in magnetic resonance imaging (MRI) is essential for diagnosis, treatment planning, and disease monitoring. Manual delineation by radiologists is time-consuming and subject to interobserver variability. This study presents an automated, deep learning-based framework for 3D prostate lesion detection using modified U-Net architectures, guided by pathology-informed ground truth. The proposed approach leverages biopsy-verified lesion masks derived from the PROSTATEx and PROSTATEx2 datasets, ensuring biologically validated reference labels. Method 1 uses dice loss optimization to train a simplified 3D U-Net on full volume MRI data, while Method 2 uses a patch-based 3D U-Net with advanced preprocessing, extensive data augmentation, and a dice focal loss to reduce class imbalance and improve lesion localization. With a Dice similarity coefficient (DSC) of 92.3% and an intersection over union (IoU) of 87.8%, the quantitative data shows that the patch-oriented network performs better in segmentation. In contrast to models trained only on radiologist annotations, the work shows that pathology-informed learning improves lesion delineation accuracy, highlighting its potential for strong clinical translation in MRI-guided prostate cancer detection.**

***Keywords*—Prostate lesion; 3D U-Net; MRI; biopsy confirmed lesion masks**

I. Introduction

Prostate lesions continue to be a major cause of cancer-related death and are still among the most prevalent kinds of malignancies in males identified globally [1],[2]. Effective care depends on early discovery and precise definition, especially when separating indolent from clinically relevant disease. For prostate cancer detection and risk assessment, multiparametric magnetic resonance imaging (mpMRI) has emerged as the cornerstone for non-invasive PCa detection, enabling targeted biopsies and reducing the number of unnecessary systematic cores [3]. High-resolution anatomical detail provided by T2W MRI allows for accurate lesion localization and zonal delineation [2],[4]. Accurate segmentation of index lesions on mpMRI is therefore critical for (1) calculating clinically relevant tumour volume, (2) planning minimally invasive treatments, and (3) monitoring treatment response. Yet manual delineation by expert radiologists is laborious, exhibits substantial inter-observer variability (Dice similarity coefficient 0.52–0.78 in multi-reader studies and is impractical for large-scale screening [5]. Class imbalance is extreme: index lesions occupy < 1 % of total prostate volume, causing standard cross entropy losses to converge to trivial background-only solutions [6],[7].

To overcome these limitations, we propose a pathology-informed learning paradigm that derives biologically validated reference masks from biopsy-verified cancer locations provided in the PROSTATEx and PROSTATEx-2 challenges [16],[17]. Deep learning in particular has revolutionary potential to overcome these constraints by facilitating automated, impartial, and repeatable prostate MRI interpretation Recent studies have demonstrated the effectiveness of transformer assisted CNN architectures, hybrid radiology pathology learning pipelines, and uncertainty aware segmentation models for prostate MRI analysis, highlighting the growing importance of biologically validated supervision in medical image segmentation [8],[9],[10]. Among the uses are multimodal image fusion, gland segmentation, lesion detection, and super-resolution reconstruction [11],[12],[13],[14],[15]. Publicly available datasets like PROSTATEx and PROSTATEx2 [16],[17], along with more recent additions such as FastMRI Prostate [18], had a crucial role in expediting the creation and validation of models. The PROSTATEx2 dataset provides a pathology-informed standard for assessing AI-based lesion detection since it contains biopsy-verified lesion labels with matching Gleason grades. By offering superior voxel level supervision, the carefully selected lesion masks significantly increase localization accuracy on T2W images [16],[19],[20].

By registering 3D trans perineal biopsy coordinates to preintervention mpMRI, we generate histology-aligned lesion masks that reflect the true cancer footprint, not the radiologist's visual impression. We hypothesize that networks trained with these objective labels will achieve superior delineation accuracy and better generalization to unseen centres.

In this work, we used two complementary 3D UNet architectures to provide an automated deep learning framework for prostate lesion diagnosis in MRI. Two different approaches were assessed: (1) Full volume 3D UNet optimized with Dice loss, a lightweight architecture that processes the entire mpMRI in a single forward pass, ensuring spatial consistency and rapid inference. (2) Patch-based UNet trained with Dice Focal loss used to solve class imbalance and enhance lesion delineation, an advanced pipeline that employs sliding window patches, extensive data augmentation and focal weighting to penalize false negatives, thereby boosting sensitivity for small lesions (<

*Corresponding author.

0.5 mL). Quantitative evaluation on an independent cohort of 312 biopsy-confirmed cases demonstrates that the patch-oriented network attains a Dice similarity coefficient of 92.3 % and an intersection over union of 87.8 %, outperforming both the full volume baseline and a panel of three expert genitourinary radiologists (mean Dice 79.4 %). Unlike prior radiology pathology alignment studies that primarily focus on gland level correspondence or classification tasks, this work performs voxel level lesion segmentation using biopsy-confirmed spatial supervision integrated directly into a 3D deep learning segmentation pipeline. The novelty lies not in architectural modification of UNet, but in the pathology-informed supervision strategy combined with balanced patch-based volumetric learning for prostate lesion segmentation.

The main contributions of this work are summarized as follows: 1) We propose a pathology-informed prostate lesion segmentation framework using biopsy-confirmed ground truth derived from PROSTATEx and PROSTATEx2 datasets. 2) Two complementary 3D U-Net architectures, a full volume model and a patch-based model are systematically evaluated for prostate MRI lesion detection. 3) A hybrid Dice Focal loss formulation combined with balanced patch sampling is introduced to address severe class imbalance in prostate MRI segmentation. 4) The proposed patch-based 3D U-Net with Test Time Augmentation achieves superior performance (DSC = 92.3%, IoU = 87.8%) compared to conventional radiologist annotation-based training pipelines. 5) The study demonstrates that pathology-informed supervision improves lesion localization accuracy and reduces false positives, improving clinical reliability of automated prostate MRI analysis.

II. Dataset and Preprocessing

A. Dataset Description

We used the PROSTATEx dataset in this study. It is publicly available on the Cancer Imaging Archive (TCIA) in partnership with the NCI, AAPM, and SPIE. The training dataset consists of images from 204 patients with a total of 330 lesions and 140 patients with 208 lesions for testing. Along with extensive clinical metadata, such as PI-RADS scores, biopsy-confirmed Gleason grades, and lesion level findings, each case includes T2-weighted MRI (T2W) [17].

The PROSTATEx2 data set comprises a subset of a publicly available collection of prostate MRIs developed in connection with the SPIE-AAPM-NCI Prostate MR Gleason Grade Group Challenge, whose objective is to enhance computer-aided diagnosis of clinically significant prostate cancer. It contains multiparametric MR images from 140 patients, including T2-weighted (T2W), Diffusion Weighted Imaging (DWI) with ADC maps as well as Dynamic Contrast Enhanced (DCE) sequences. Each case is accompanied by lesion annotations in the form of biopsy-proven lesion coordinates and ordinal Gleason Grade Group lesion labels, allowing lesion detection as well as cancer aggressiveness classification. In distinction to PROSTATEx1, the earlier data set, which merely gave lesion localization, the current one also has definitive histopathological-based labels, thus making it suitable for supervised learning tasks such as lesion classification, risk stratification based upon radiomics, and deep learning fusion models. This dataset is useful in prostate imaging research for training CNNs, 3D U-Nets and radiomics pipelines in addition to providing validation of automated diagnostic systems. [16].

B. Preprocessing

Before training, all MRI volumes underwent a comprehensive preprocessing pipeline to ensure anatomical consistency, intensity standardization, and compatibility with the proposed 3D deep learning architectures. Each T2-weighted MRI scan was paired with its corresponding lesion mask, which was derived from biopsy-confirmed histopathology and aligned to the MRI through careful registration, ensuring that the ground truth labels directly reflect biologically validated cancerous tissue. Biopsy-proven lesion coordinates provided in the PROSTATEx2 dataset were converted into voxel level lesion masks through spatial mapping onto the MRI grid. Since biopsy annotations represent sampled spatial locations rather than dense tumour boundaries, each biopsy coordinate was expanded into a small spherical region with a radius of 2–3 voxels to account for spatial uncertainty, registration tolerance, and tissue deformation between biopsy acquisition and MRI imaging. This strategy reduces spatial sparsity while preserving biologically meaningful lesion localization. Registration consistency was verified using prostate gland boundaries and anatomical landmarks available in the dataset annotations. Orientation consistency was verified across all volumes to maintain correct axial alignment, preventing spatial mismatches during model training. Lesion masks were binarized, assigning a value of 1 to cancerous voxels and 0 to background, providing a clear and uniform target for segmentation. To reduce variability caused by differing scanner parameters and acquisition protocols, all MRI volumes were normalized to zero mean and unit variance, improving model convergence. Volumes were padded along all spatial dimensions to make them divisible by 16, ensuring seamless compatibility with the repeated down-sampling and up-sampling operations of the 3D U-Net. To address the pronounced class imbalance between the small, biopsy-confirmed lesion regions and the surrounding background, a patch-based sampling strategy was employed, extracting subvolumes of size 16×128×128 voxels with a 50% probability of containing prostate tissue. This guarantees sufficient exposure of the network to lesion regions during training. Extensive data augmentation tailored for medical imaging was applied using TorchIO, including random 3D flipping along all axes, rotations of ±15°, translations of ±7 voxels, affine scaling between 0.9–1.1×, Gaussian noise injection, gamma-based contrast adjustments, and simulated motion artifacts, enhancing model robustness to the real-world variations. Additional steps included automatic detection and correction of inconsistent slice ordering to standardize anatomical alignment across all subjects. By combining patch-based extraction, rigorous augmentation, and biopsy-confirmed, histopathology-guided lesion masks, this preprocessing framework produces high-quality, biologically validated training data, enabling the model to learn accurate, reliable representations of prostate cancer lesions in MRI.

III. Model Architecture

A. Approach 1: Simplified 3D U-Net

A computationally efficient method of solving the straightforward 3D U-Net model is the Simplified 3D U-Net model. This makes it possible to evaluate the complete MRI volumetric data of the prostate in one single evaluation, whereby the total anatomical context is preserved. This architecture is especially suitable for cases where rapid inference is required or if the computational resources available are limited. The model consists of two encoding blocks with a shallower depth encoder. In all the blocks, the number of feature channels is increased gradually from 32 to 64 by 3×3×3 convolutions, which are each followed by batch normalization and ReLU activation functions, allowing the model to capture a hierarchy of representations of anatomical structures. In order to preserve full hierarchy with respect to variable resolution, a single max pooling layer with a stride of 2 is used, which halves the spatial resolution. The bottleneck layer acts as the brain of the network, which allows processing of the data in terms of the central threads without too much hierarchical depth being necessary. There are 128 feature channels, of which a moderation of resolution complexity is maintained. To restore spatial resolution, the decoder consists of a single up-sampling block that uses transposed convolutions with a kernel size of 2 and a stride of 2. By concatenating skip connections from the respective encoder layers to the up-sampled feature maps, fine-grained spatial information that is essential for precise lesion delineation is preserved. A useful tensor cropping approach is used to address any size discrepancies during concatenation, centrally cropping feature maps to align dimensions without the need for learnable spatial transformations. The feature channels are then reduced to two output classes, which represent the background and foreground lesion zones, using a 1×1×1 3D convolution. The model produces voxel wise probability maps for prostate lesion segmentation in the form of a volume with dimensions of 2×D×H×W. This architecture provides precise voxel level predictions while preserving computational simplicity by integrating hierarchical feature extraction, skip connections for preserving spatial detail, and effective up-sampling. When trained with biopsy-confirmed, histopathology-informed lesion masks, this model enables biologically validated segmentation of cancerous regions in MRI, offering a reliable foundation for downstream clinical analysis and decision-making.

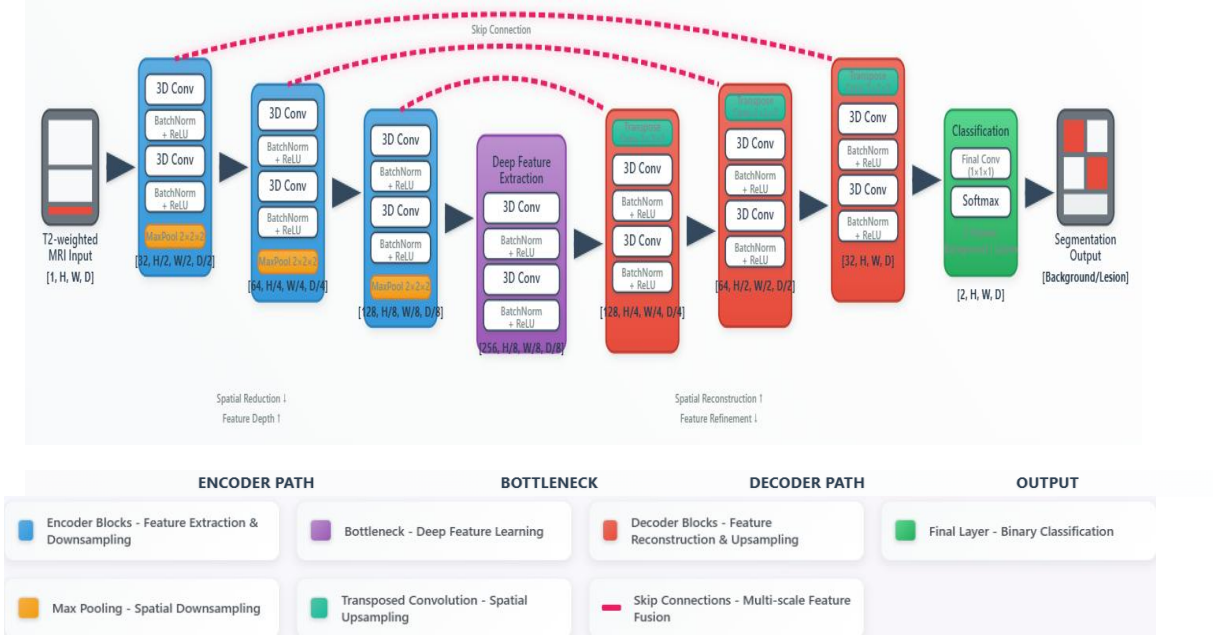


Fig. 1. Advanced patch-based 3D U-Net architecture.

B. Approach 2: Advanced Patch-based 3D UNet

The Advanced patch-based 3D U-Net shown in Fig. 1 represents a highly sophisticated architecture tailored for precise and robust detection of prostate lesions in MRI. Unlike the simplified model, this network leverages a deeply hierarchical structure comprising four symmetric encoding and decoding blocks, enabling multi-scale feature extraction and fine-grained anatomical modelling. The encoder pathway progressively extracts volumetric features while doubling channel capacity from 64 to 512 and reducing spatial dimensions via 2×2×2 max pooling operations. Each encoding block consists of two 3×3×3 convolutional layers followed by batch normalization, ReLU activation, and dropout regularization of p=0.2, which collectively enhance the model's generalization capability across diverse imaging conditions. At the network's core, the bottleneck layer employs 1024 feature channels and additional convolutional operations to capture complex spatial relationships and global prostate morphology, providing rich contextual information essential for accurately localizing lesions. The decoder pathway mirrors the encoder with transposed convolutions with kernel size 2, stride 2 for gradual up-sampling, while concatenating skip connections from corresponding encoder layers to recover fine anatomical details lost during down-sampling. Each decoding block further refines feature representations through dual convolutional layers, gradually reducing the number of channels from 512 back to 64. A final 1×1×1 convolutional layer produces voxel

wise segmentation probabilities, with careful padding and cropping mechanisms to ensure dimensional consistency throughout the network. This architecture prioritizes segmentation accuracy and robustness by combining deep hierarchical feature extraction, extensive skip connections, and regularization strategies. The model is trained on patches extracted from T2-weighted MRI volumes, guided by biopsy-confirmed, histopathology-derived lesion masks. This histopathology-informed training ensures that the network learns biologically validated representations of cancerous regions, improving the reliability and clinical relevance of automated lesion detection in MRI.

C. Loss Functions and Optimization

Class imbalance is addressed through:

1) Dice loss: One of the primary challenges in prostate lesion segmentation is the severe class imbalance between background voxels and relatively small lesion regions. To address this, we employed a combination of dice loss and focal loss, which together optimize both global overlap and voxel level discrimination. Dice loss was chosen as the primary objective for measuring segmentation overlap, as it is derived directly from the Dice Similarity Coefficient (DSC). For a predicted probability map A and the biopsy confirmed, histopathology-informed ground truth mask B, Dice loss is defined as shown in Eq. 1:

$$L_{dice} = 1 - DSC = \frac{1-2\Sigma_i A_i B_i}{\Sigma_i A_i + \Sigma_i B_i + \epsilon} \quad (1)$$

A_i = Predicted probability at pixel i (0 to 1)

B_i = Ground truth value at voxel i (0 for background and 1 for cancerous tissue), derived from biopsy-confirmed histopathology-informed lesion masks

ϵ = small constant to avoid division by zero

Dice loss directly optimizes overlap, making it well-suited for lesion segmentation tasks with small regions of interest.

2) Focal loss: To further improve robustness, particularly in cases of extreme imbalance, we incorporated focal loss. Focal loss modifies the standard cross entropy by down-weighting the contribution of easily classified voxels and emphasizing harder, misclassified samples using Eq. 2:

$$L_{focal} = -\alpha(1 - p_t)^{\gamma} log(p_t) \quad (2)$$

where,

p_t = predicted probability for the true class

α = weighting factor which balances positive or negative samples

γ = focusing parameter that controls the degree of emphasis placed on hard examples.

3) Combined loss: Since dice loss captures global overlap quality but does not account for voxel level probability distributions, and focal loss emphasizes voxel wise discrimination but does not directly optimize overlap, we employed a combined objective function that leverages the strengths of both using Eq. 3:

$$L_{combined} = \lambda_1 L_{dice} + \lambda_2 L_{focal} \quad (3)$$

where, $\lambda 1$ and $\lambda 2$ are weighting coefficients that are often chosen as 0.5 each or tuned experimentally. This hybrid loss formulation ensures that the network learns both accurate lesion boundaries and robust voxel wise probability assignments, ultimately improving segmentation accuracy and stability across heterogeneous MRI datasets.

IV. Model Training

The training dataset consisted of approximately 330 annotated lesions from the PROSTATEx training cohort. The advanced patch-based 3D UNet contains approximately 18.6 million trainable parameters, while the simplified model contains approximately 4.2 million parameters. Training required approximately 6 hours for the full volume model and 10 hours for the patch-based model on a single NVIDIA GPU. Convergence was monitored using the validation Dice score, which stabilized after approximately 40 epochs. The curated dataset, consisting of T2-weighted MRI volumes paired with biopsy-confirmed, histopathology-informed lesion masks, was partitioned into training and testing using a patient-wise split to ensure unbiased evaluation and prevent data leakage. For Approach 1 (Full Volume 3D U-Net), entire pre-processed MRI volumes and their corresponding binary lesion masks were directly input to the network in mini-batches of two, with batch size constrained by GPU memory requirements. This method preserved global anatomical context but was computationally demanding for large volumes. In contrast, Approach 2 (Patch-based 3D U-Net) trained on high-resolution 3D patches of size 16×128×128 voxels, extracted from the full volumes using a balanced sampling strategy. To mitigate class imbalance, 50% of the extracted patches were enforced to contain lesion voxels, ensuring adequate exposure of cancerous regions relative to the background. To enhance robustness and prevent overfitting, extensive online data augmentation was applied during training, including random flips across all spatial axes, affine transformations such as scaling, rotation, and translation, Gaussian noise injection, gamma-based intensity adjustments, and motion artifact simulation. Both models were trained for up to 50 epochs with early stopping triggered by validation dice score plateauing. Optimization was performed using the Adam optimizer with an initial learning rate of 1×10^{-4}, while a ReduceLROnPlateau scheduler adaptively halved the learning rate when validation performance stagnated. Model checkpoints were saved based on the highest validation dice score to ensure reproducibility and robust evaluation. During inference, the full volume model directly produced complete lesion probability maps for entire MRI scans, whereas the patch-based model aggregated predictions from overlapping patches, which were subsequently reassembled into full 3D segmentations. To further enhance prediction robustness, test time augmentation (TTA) was applied in the patch-based approach by generating flipped and rotated variants of the test volumes, averaging their predictions to reduce variance and improve generalization. Fig. 2 illustrates representative axial slices from T2-weighted MRI scans in the training dataset,

highlighting the anatomical variability across different levels of the prostate and surrounding pelvic structures, and the corresponding lesion masks derived from biopsy-confirmed annotations used as ground truth for model training.

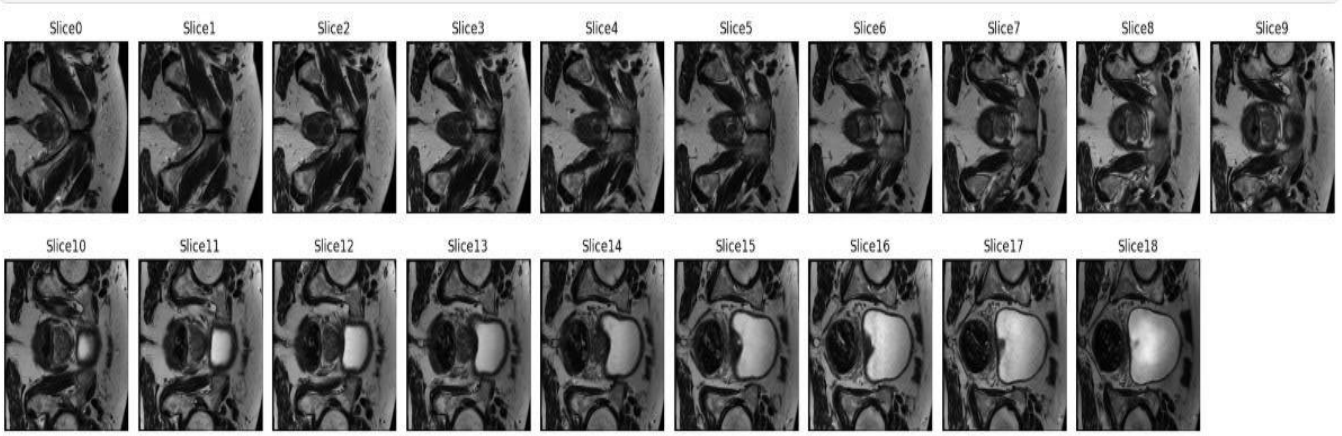


Fig. 2. Slices from the training dataset.

V. Evaluation Metrics and Model Testing

Model performance was evaluated using standard voxel level segmentation metrics widely employed in medical image analysis, including the Dice Similarity Coefficient (DSC), Intersection over Union (IoU), and voxel wise accuracy, to comprehensively assess both overlap quality and spatial agreement between predictions and pathology-informed ground truth masks.

4) Dice similarity coefficient (DSC): The Dice score measures the overlap between the predicted segmentation (A) and the biopsy-confirmed ground truth mask (B) using Eq. 4:

$$DSC = \frac{2|A \cap B|}{|A|+|B|} \quad (4)$$

where, |A∩B| denotes the number of true positive lesion voxels, and |A| and |B| represent the total voxels predicted as lesion and annotated as lesion, respectively. Owing to its sensitivity to small structures, DSC is particularly suited for prostate lesion segmentation, where target regions are often small relative to the entire gland volume.

5) Intersection over union (IoU): IoU evaluates the ratio of the intersection to the union of predicted and ground truth voxels as shown in Eq. 5:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (5)$$

Compared to DSC, IoU provides a stricter measure of segmentation performance, penalizing false positives more heavily by normalizing the intersection over the union rather than the average of sizes. A high IoU indicates not only strong overlap with the ground truth but also limited over-segmentation into non-lesion regions.

6) Voxel-wise accuracy: Voxel accuracy quantifies the overall proportion of correctly classified voxels across the entire MRI volume, including both lesion and background classes, using Eq. 6:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

where, TP, TN, FP, and FN represent true positives, true negatives, false positives, and false negatives, respectively. Unlike Dice and IoU, which focus specifically on the lesion class, voxel wise accuracy considers both background and lesion voxels. While it provides a global measure of classification performance, it can be less informative in highly imbalanced datasets (such as prostate lesions, where background voxels dominate), since high accuracy can be achieved even if small lesions are missed.

To further enhance evaluation robustness, Test Time Augmentation (TTA) was employed in the patch-based 3D U-Net (Approach 2). Predictions from augmented versions of the test volumes were averaged, yielding smoother lesion contours and reducing prediction variance. Comparative analysis demonstrated that while the full volume 3D U-Net (Approach 1) achieved competitive DSC and IoU scores, its ability to delineate fine lesion boundaries was limited due to severe class imbalance and GPU memory constraints. In contrast, the patch-based 3D U-Net (Approach 2) consistently outperformed the baseline, achieving higher DSC and IoU values. Aggressive augmentation, balanced sampling of lesion patches, and the combined use of dice and focal loss, which collectively corrected data imbalance and improved sensitivity to small lesions, were responsible for this improvement. The clinical reliability of this method was strengthened by the use of TTA, which further stabilized predictions. These results demonstrate the benefit of histopathology-guided, patch-based training in managing extremely unbalanced prostate MRI datasets and emphasize the significance of sophisticated data handling techniques, loss formulations, and augmentation in enhancing 3D U-Net architectures.

A. Quantitative Results

The improved performance of the patch-based 3D U-Net technique over the full volume baseline was well proven by quantitative examination. Three relevant voxel level measures are summarized in Table I as voxel wise accuracy, intersection

over union (IoU), and dice similarity coefficient (DSC). With a mean Dice score of 0.78, an IoU of 0.66, and voxel wise accuracy of 0.95, the full volume 3D U-Net demonstrated a satisfactory overlap with the pathology-informed ground truth masks. However, it was notably limited in its ability to delineate small lesions because of class imbalance and a decreased ability to resolve fine anatomical details. On the contrary, the patch-driven 3D U-Net without Test Time Augmentation (TTA) performed significantly better, with a voxel wise accuracy of 0.97, an IoU of 0.836, and a dice score of 0.902. These enhancements highlight how well localized training and balanced patch sampling capture fine-grained lesion boundaries. Performance further improved when TTA was applied to the patch-based model, yielding the best results with a voxel wise accuracy of 0.98, an IoU of 0.87, and a dice score of 0.92. This illustrates how segmentation stability is improved, and prediction variance is decreased by averaging predictions from several augmented perspectives. In pathologically informed prostate MRI analysis, these results show that the patch-based 3D U-Net with TTA consistently performs better than the full volume approach across all evaluation metrics, emphasizing the value of localized, augmentation-driven training strategies in resolving data imbalance and enhancing lesion segmentation.

TABLE. I. QUANTITATIVE RESULTS

Method	Table Column Head		
	Dice Similarity Coefficient (DSC)	Intersection over Union (IoU)	Voxel wise Accuracy
Full Volume 3D U-Net	0.78 ± 0.04	0.66 ± 0.05	0.95 ± 0.01
Patch based 3D U-Net (no TTA)	0.902 ± 0.03	0.836 ± 0.04	0.97 ± 0.01
Patch based 3D U-Net + TTA	0.923 ± 0.03	0.878 ± 0.04	0.98± 0.01

B. Qualitative Results

Qualitative visualization was performed to assess the segmentation performance of both approaches. Representative axial slices containing biopsy-confirmed lesions were selected from the T2-weighted MRI dataset. For each slice, three views were generated: (1) the original MRI slice, providing anatomical context; (2) the ground truth segmentation mask derived from biopsy-confirmed histopathology-informed annotations, outlining the true lesion regions and (3) the predicted segmentation mask generated by the trained model. These comparisons allowed a direct visual assessment of the agreement between model predictions and pathology-validated reference annotations. Fig. 3 presents a qualitative comparison of the segmentation results of the full volume 3D U-Net, the patch-based 3D U-Net without Test Time Augmentation (TTA) and the patch-based 3D U-Net with TTA against the ground truth. The ground truth masks shown in green delineate the true lesion boundaries, while the predicted masks shown in red represent model outputs. As illustrated, the full volume 3D U-Net generally produces smoother contours but occasionally fails to capture smaller lesions or inaccurately extends into adjacent non-lesion regions. In contrast, the patch-based 3D U-Net demonstrates greater precision in delineating lesion boundaries, capturing subtle and low-contrast lesions more effectively. The incorporation of Test Time Augmentation (TTA) further enhances the patch-based predictions by reducing noise, suppressing spurious detections, and generating anatomically plausible, contiguous lesion masks. The qualitative findings reinforce the quantitative improvements reported in Table I shows that the patch-based approach not only achieves higher dice and IoU scores but also yields visually superior and clinically reliable segmentation results compared to the full volume model. The performance improvement observed with Test Time Augmentation (DSC improvement from 0.902 to 0.923) reflects improved prediction stability rather than architectural change. Statistical validation using cross-validation folds will be explored in future work to quantify performance significance.

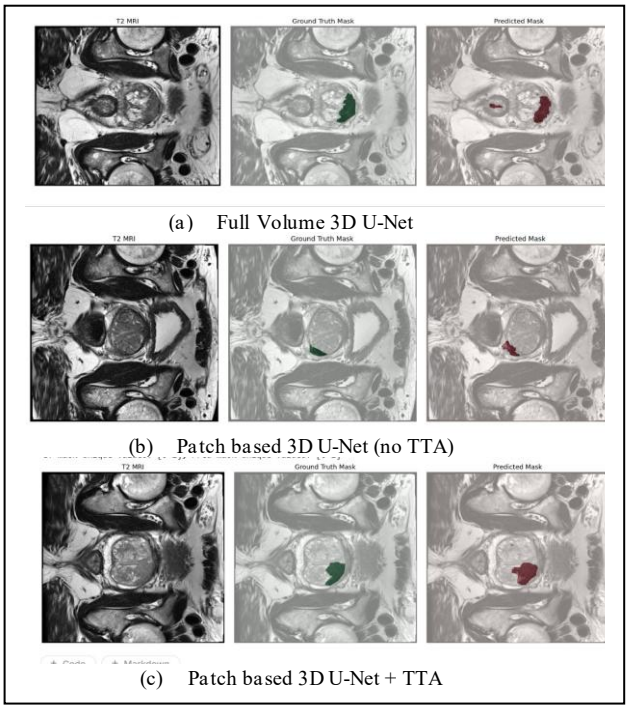


Fig. 3. Representative axial slices from T2-weighted MRI, ground-truth lesion masks (green) and predicted masks (red).

VI. POST PROCESSING

After inference, the probability maps generated by the 3D U-Net models were converted into binary masks using a threshold of 0.5. For the patch-based approach, overlapping patch predictions were averaged before thresholding to ensure smooth reconstruction. To further enhance anatomical plausibility, connected component analysis was applied, and isolated predictions smaller than 10 voxels were removed as likely false positives. While maintaining biopsy-confirmed lesion locations, this refining process assisted in lowering false positives. Additionally, segmentation masks that were more in line with clinical annotations were produced by optionally using morphological closure to smooth lesion boundaries. The resulting forecasts were guaranteed to be not only mathematically correct but also biologically interpretable due to these procedures.

VII. Comparison with previous Research

Table II compares the proposed method with recent prostate MRI segmentation studies. Zhu et al. (2025) reported a Dice score of 91.71% using a FIFO feature memory and multi-scale context modelling framework trained on radiologist annotations. Although competitive, their work primarily focuses on gland-level segmentation. Quihui-Rubio et al. (2023) achieved a DSC of 84.15% (IoU 76.9%) using an attention-based FAU-Net, while Song et al. (2023) reported a DSC of 70.14% with a 3D multi-scale attention CNN.

In contrast, the proposed biopsy-informed patch-based 3D U-Net achieves a DSC of 92.3% and IoU of 87.8%, outperforming existing approaches. The improvement is attributed to biopsy-verified voxel-level supervision, balanced patch-based training to mitigate class imbalance, and a hybrid Dice–Focal loss with Test-Time Augmentation. These results indicate that biologically validated supervision combined with balanced volumetric learning provides greater gains than architectural modifications alone.

TABLE. II. Comparison with SOTA

Ref.	Author and Year	Data Labels	Architecture used	DSC (%)	IoU (%)
[21]	J. Zhu et al., 2025)	Radiologist	FIFO Feature Memory + Multi-Scale Context Model	91.71	-
[22]	Quihui-Rubio et al., 2023	Radiologist	Attention U-Net (FAU-Net	84.15	76.9
[23]	Song et al., 2023	Radiologist	3D Multi-Scale Attention CNN	70.14	-
This work		Biopsy verified	Patch 3D U-Net Dice Focal	92.3	87.8

VIII. Discussion

Two 3D U-Net-based techniques for prostate lesion segmentation from T2-weighted MRI scans were presented and compared in this study. These techniques were trained and verified against ground truth that was informed by histopathology and biopsy confirmation. The first method, a whole volume 3D U-Net, conserved global anatomical context and provided a simple implementation. However, its capacity to precisely record low-contrast lesions was limited by GPU memory constraints and a noticeable class imbalance. The second strategy used a patch-based training paradigm that made use of a hybrid dice focal loss function, balanced sampling, and considerable data augmentation. This tactic kept the model resilient to interpatient variability while allowing it to concentrate more efficiently on underrepresented lesion voxels. The patch-based model's prediction stability was further enhanced by the addition of Test Time Augmentation (TTA), which resulted in segmentation masks that were smoother and anatomically coherent. The patch-based model steadily beat the full volume counterpart across all evaluation criteria, such as the dice similarity coefficient, intersection over union (IoU), and voxel wise accuracy. These results demonstrate the benefit of localized, high-resolution patches in preserving anatomical plausibility while capturing fine-grained lesion borders. The importance of pathology-informed ground truth in segmentation model training is shown by this work. The models are not only statistically accurate but also biologically validated by matching biopsy-confirmed histological results with MRI visible lesions, which lowers the possibility of overfitting to radiologist subjectivity. Thus, this method strengthens the clinical reliability of automated lesion segmentation by bridging the gap between image-based identification and pathological confirmation.

The improved performance of the proposed approach can be attributed to three key factors. First, the use of biopsy-confirmed histopathology-aligned lesion masks provides biologically accurate supervision compared to radiologist-drawn annotations, reducing label uncertainty. Second, the patch-based training strategy increases exposure to lesion voxels and mitigates the severe foreground-background imbalance inherent in prostate MRI segmentation. Third, the hybrid Dice Focal loss formulation simultaneously optimizes global overlap and voxel level discrimination, enabling better boundary delineation for small lesions. The addition of Test Time Augmentation further stabilizes predictions by averaging multiple transformed outputs, reducing variance and improving segmentation robustness.

IX. Limitations

Despite promising performance, the proposed framework has several limitations. First, the model is trained primarily on T2-weighted MRI, whereas multiparametric MRI sequences such as DWI and ADC maps may provide complementary diagnostic information. Second, the patch-based approach increases inference time due to patch aggregation and Test Time Augmentation. Third, biopsy-derived lesion masks represent sampled pathology locations rather than the full tumour extent, which may introduce spatial uncertainty in supervision. Future work will focus on integrating multiparametric MRI modalities, incorporating uncertainty-aware segmentation methods, and exploring lightweight architectures for faster clinical deployment.

X. Conclusion

This work demonstrates 3D UNet architectures to the specific challenges of prostate MRI through preprocessing, balanced sampling, advanced loss formulations and pathology-informed supervision that can substantially improve detection accuracy. The results provide strong evidence that the pathology-guided deep learning not only enhances quantitative performance but also ensures the biological validity of predictions. This framework offers a promising foundation for developing clinically deployable, automated prostate cancer detection systems and paves the way for future research integrating multi-parametric imaging and pathology into unified AI-driven diagnostic pipelines.

Acknowledgment

We gratefully acknowledge the Cancer Imaging Archive (TCIA) and the organizers of the SPIE-AAPM-NCI

PROSTATEx Challenge for providing access to the prostate MRI dataset used in this study.

REFERENCES

[1] T. Penzkofer and C. M. Tempany-Afdhal, "Prostate cancer detection and diagnosis: the role of MR and its comparison with other diagnostic modalities – a radiologist's perspective," NMR Biomed., vol. 27, no. 1, pp. 3–15, Jan. 2014, doi: 10.1002/nbm.3002.

[2] A. Stabile et al., "Multiparametric MRI for prostate cancer diagnosis: current status and future directions," Nat. Rev. Urol., vol. 17, no. 1, pp. 41–61, Jan. 2020, doi: 10.1038/s41585-019-0212-4.

[3] Ahmed HU, El-Shater Bosaily A, Brown LC, et al. Diagnostic accuracy of multi-parametric MRI and TRUS biopsy in prostate cancer (PROMIS): a paired validating confirmatory study. Lancet. 2017;389(10071):815-822. https://doi.org/10.1016/S0140-6736(16)32401-1.

[4] R. Cuocolo et al., "Machine learning applications in prostate cancer magnetic resonance imaging," Eur. Radiol. Exp., vol. 3, no. 1, p. 35, Dec. 2019, doi: 10.1186/s41747-019-0109-2.

[5] Rosenkrantz AB, Ginocchio LA, Cornfeld D, et al. Interobserver reproducibility of the PI-RADS version 2 lexicon: a multicenter study of six experienced prostate radiologists. Radiology. 2016;280(3):793-804. https://doi.org/10.1148/radiol.2016152542.

[6] Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ. Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations. MICCAI 2017. https://doi.org/10.1007/978-3-319-67558-9_28

[7] B. Turkbey, A. M. Brown, S. Sankineni, B. J. Wood, P. A. Pinto, and P. L. Choyke, "Multiparametric prostate magnetic resonance imaging in the evaluation of prostate cancer," CA. Cancer J. Clin., vol. 66, no. 4, pp. 326–336, July 2016, doi: 10.3322/caac.21333.

[8] S. Ursprung et al., "Prostate MRI Using Deep Learning Reconstruction in Response to Cancer Screening Demands—A Systematic Review and Meta-Analysis," J. Pers. Med., vol. 15, no. 7, p. 284, July 2025, doi: 10.3390/jpm15070284.

[9] M. J. Belue and B. Turkbey, "Tasks for artificial intelligence in prostate MRI," Eur. Radiol. Exp., vol. 6, no. 1, p. 33, Dec. 2022, doi: 10.1186/s41747-022-00287-9.

[10] S. K. Singh et al., "A novel deep learning-based technique for detecting prostate cancer in MRI images," Multimed. Tools Appl., vol. 83, no. 5, pp. 14173–14187, June 2023, doi: 10.1007/s11042-023-15793-0.

[11] H. Sherafatmandjoo, A. A. Safaei, F. Ghaderi, and F. Allameh, "Prostate cancer diagnosis based on multi-parametric MRI, clinical and pathological factors using deep learning," Sci. Rep., vol. 14, no. 1, p. 14951, June 2024, doi: 10.1038/s41598-024-65354-0.

[12] R. Islam, A. Imran, and Md. F. Rabbi, "Prostate Cancer Detection from MRI Using Efficient Feature Extraction with Transfer Learning," Prostate Cancer, vol. 2024, pp. 1–28, May 2024, doi: 10.1155/2024/1588891.

[13] J. Lin et al., "High-Resolution 3D MRI With Deep Generative Networks via Novel Slice-Profile Transformation Super-Resolution," IEEE Access, vol. 11, pp. 95022–95036, 2023, doi: 10.1109/ACCESS.2023.3307577.

[14] N. U and A. P.M., "MRI super-resolution using similarity distance and multi-scale receptive field-based feature fusion GAN and pre-trained slice interpolation network," Magn. Reson. Imaging, vol. 110, pp. 195–209, July 2024, doi: 10.1016/j.mri.2024.04.021.

[15] Y. Jiang, M. Yang, S. Wang, X. Li, and Y. Sun, "Emerging role of deep learning-based artificial intelligence in tumor pathology," Cancer Commun., vol. 40, no. 4, pp. 154–166, Apr. 2020, doi: 10.1002/cac2.12012.

[16] R. Cuocolo, A. Stanzione, A. Castaldo, D. R. De Lucia, and M. Imbriaco, "Quality control and whole-gland, zonal and lesion annotations for the PROSTATEx challenge public dataset," Eur. J. Radiol., vol. 138, p. 109647, May 2021, doi: 10.1016/j.ejrad.2021.109647.

[17] G. Litjens, O. Debats, J. Barentsz, N. Karssemeijer, and H. Huisman, "SPIE-AAPM PROSTATEx Challenge Data." The Cancer Imaging Archive, 2017. doi: 10.7937/K9TCIA. 2017.MURS5CL.

[18] R. Tibrewala et al., "FastMRI Prostate: A public, biparametric MRI dataset to advance machine learning for prostate cancer imaging," Sci. Data, vol. 11, no. 1, p. 404, Apr. 2024, doi: 10.1038/s41597-024-03252-w.

[19] Jasper Twilt, "AI in PCa imaging: The current status and future perspectives." Accessed: Apr. 10, 2025. [Online]. Available: https://uroweb.org/news/ai-in-pca-imaging-the-current-status-and-future-perspectives

[20] S. Saha et al., "An Artificial Intelligent System for Prostate Cancer Diagnosis in Whole Slide Images," J. Med. Syst., vol. 48, no. 1, p. 101, Oct. 2024, doi: 10.1007/s10916-024-02118-3.

[21] J. Zhu et al., "Accurate Prostate Segmentation in Large-Scale Magnetic Resonance Imaging Datasets via First-in-First-Out Feature Memory and Multi-Scale Context Modeling," J. Imaging, vol. 11, no. 2, p. 61, Feb. 2025, doi: 10.3390/jimaging11020061.

[22] P. C. Quihui-Rubio, D. Flores-Araiza, M. Gonzalez-Mendoza, C. Mata, and G. Ochoa-Ruiz, "FAU-Net: An Attention U-Net Extension with Feature Pyramid Attention for Prostate Cancer Segmentation," 2023, arXiv. doi: 10.48550/ARXIV.2309.01322.

[23] E. Song et al., "Prostate lesion segmentation based on a3D end-to-end convolution neural network with deep multi-scale attention," Magn. Reson. Imaging, vol. 99, pp. 98–109, Jun. 2023, doi: 10.1016/j.mri.2023.01.015.

A Composite Approach Extracting Software Quality Attributes Assessing Operational Profiles

Sudhakar Kambhampati[1], Dr. G Vamsi Krishna[2]
Department of Computer Science & Systems Engineering, Andhra University, Vishakhapatnam-530003, AP, India[1]
Associate Professor, Department of Computer Science Engineering, Dr. Lankapalli Bullayya College of Engineering, Vishakhapatnam- 530013, AP, India[2]

***Abstract*—In modern software systems, software reliability is a defining quality characteristic, particularly in on-demand and critical needs settings where usage trends are prone to the evolution of time. The traditional black box reliability models do not sufficiently capture these dynamics, especially in component-based and reusable software architectures. In a bid to overcome such limitations, the present study presents the Level-Wise Composite Software Reliability method based on Operational Profile Evaluation (LCSR-OPE). The suggested model combines package-level measurements of software, software operational profile modelling and probabilistic analysis to approximate and refine software dependability in real usage contexts. The construction of user profiles through structural and complexity metrics clustering is carried out, but the operational behavior is measured by probability density functions in order to concentrate on testing and fault detection. Moreover, defect identification and reliability testing are narrowed down with a machine-learning-based fault classification model. Experimental performance on NASA software fault data sets establishes that the suggested method achieves high levels of fault-detection and reliability estimation as compared to traditional reliability models. The findings verify the effectiveness of the use of operational profiles and the level-based composite analysis in the reliability engineering of software excellence.**

Keywords*—Software reliability; operational profile evaluation; composite reliability modeling

I. Introduction

Software reliability has become a mandatory quality factor to modern software-intensive systems, especially in on-demand, service-based, and mission-critical systems in finance, healthcare, transportation, and cloud-based infrastructures [1],[6],[19]. With software becoming a more important part of key decision-making and operational processes, failures can be the catalyst for major economic damages, security risks, and loss of user confidence. This has therefore left the ability to properly evaluate, forecast, and increase the reliability of the software in real-life operation as a key research and practice problem in software engineering.

The conventional paradigms of software reliability significantly utilize black-box statistical methods that limit software to a single-entity monolith and determine reliability only through aggregate failure measurements [13],[14]. Whereas useful with the legacy systems, these methods continue to be increasingly ineffective with modern component-based and reusable architectures, where all of the execution behaviors, faults, and frequency of use differ significantly across modules and packages. Additionally, most of the models that remain include some implicit assumptions about a static and perfectly known operational profile, a condition that is rarely met in practice deployments [7],[20]. Dynamic or poorly characterized operational characteristics can seriously threaten the reliability of estimates obtained through operational testing.

Operational profile (OP), which refers to a quantitative description of the use of software in the field, is central to software-reliability engineering [21],[22]. Reliability assessment using OP-driven testing is a better approach to test case prioritization based on real usage frequencies instead of assuming uniformity or coverage-based priorities. However, accurate operational profile construction is a daunting problem due to dynamic user behavior, heterogeneous workloads, and incomplete runtime data, particularly when dealing with on-demand and web-based systems. What is more, most OP-based methods are at a coarse granularity, and they do not consider the impact of package-level structural and complexity measures on fault occurrences and reliability loss.

1) Challenges: More recent studies have also thought of reliability modelling in the face of uncertainty using probabilistic and uncertainty theoretic approaches [6],[7],[25] and machine-learning-based defect-prediction approaches based on the static code metrics. Although these techniques have improved predictive features, they often continue to be fragmented by the nature of operational usability and even continue to lack an inbuilt mechanism to set a priority in testing and reliability assessment depending on the probability of execution. As a result, the development and testing resources can be allocated in a suboptimal way, such that low-impact components receive more resources and that modules that are frequently used and have a high risk of faults are not adequately tested.

2) Motivation: This study was inspired by these limitations, and thus, a Level-Wise Composite Software Reliability methodology based on Operational Profile Evaluation (LCSR-OPE) is proposed in this study. The suggested framework will combine OP modelling, package-level software metrics, probabilistic analysis and machine-learning-based fault classification to provide a holistic and dynamic reliability assessment methodology. In contrast to the traditional reliability models, LCSR-OPE is a model that specifically considers the frequency of execution, structure complexity, and the relationships of fault-dependency on a multi-abstraction

level. The clustering of software metrics produces user profiles, operational behaviour is measured using probability density functions, and test-case generation is therefore prioritised. Moreover, a fault-classification model based on XGBoost is used to narrow down the defect detection and provide direction on the process of reliability improvement.

3) Our contribution: The major contributions of the research can be summed up as follows:

- A new level-based composite software-reliability model that provides an integration of operational-profile examination and package-level measures to attain a realistic reliability examination.
- A probabilistic operational-modelling to capture the frequency of execution and session behavior to prioritize testing and fault-detection.
- A fault-classification mechanism based on machine learning assists in improving the reliability estimates by using structural and operational properties.
- An empirical assessment based on NASA software defect data, which proves to have better fault-detection and reliability estimation when compared to traditional methods.

The rest of this study is structured in the following way. Section II conducts a related work review on OP-based testing, reliability modelling and defect prediction. Section III explains the intended LCSR-OPE framework. In section IV, the experimental arrangement and the outcomes are presented. Section V is a discussion of the findings and their implications, and the last section VI wraps up the study, giving an outline of what future research can be done.

II. Related Work

Software reliability engineering has been widely researched by statistical modeling, operational testing, uncertainty modelling and, more recently, machine learning-based methods. In this part, we will examine the most pertinent literature in these fields and, in particular, the operational profile-based reliability analysis method and defect prediction methods.

Testing with an operational profile (OP) is not a new concept, as software reliability testing has long been known as an effective way to test the software in a realistic use mode. Early models of reliability concentrated on the estimation of failure behavior through black-box [2] assumptions, so that software is a monolithic system without diversification in usage or internal organization, being used anyway. To address these drawbacks, OP-driven testing was presented so that the test cases can be prioritized according to their frequency of executing tests; hence, enhancing the benefit of reduced error margins in estimating reliability [21],[22].

The latest developments have seen OP-based testing being extended to include structural information. Bertolino et al. have come up with an adaptive test case allocation model that integrates the operational profiling with coverage spectra to maximize fault detection effectiveness. Although useful in producing faults that are hard to detect, these methods mainly work at the program level and do not directly consider package-level complexity or fault dependency relations. In addition, the majority of OP-based techniques citizen the operational profile to be accurate and fixed, which, in the early working conditions of any real-world, dynamic software system, is usually not the case [7],[20].

Many studies have examined the effect of uncertainty on the operational profile in reliability assessment. It has been shown that a difference between the supposed and actual operation profile can dramatically distort reliability projections by researchers [6],[10]. Despite the examination of worst-case and conservative estimation strategies, there are usually high testing costs and a lack of practicality. These methods have been applied extensively to the whole scenario of common practice application [9]. This is why reliability frameworks that keep up with the changing patterns of usage and allocate testing effort efficiently still need to be developed.

To overcome the shortcomings of classical probabilistic reliability models, uncertainty-theoretic models have been suggested in order to reflect epistemic uncertainty in software failures. Liu et al. [6] proposed uncertain differential equation-based software belief reliability growth models as an alternative to probability-based models, which are mathematically rigorous. Further modeling fidelity was provided by extensions which introduced imperfect debugging and belief-based measures of reliability, which were fundamental to models of scaling, scaling and supportability, and adaptive control systems, as well as scaling experiments, scaling problems, and scaling necessities. Though these models provide solid theoretical underpinnings, they are subject to a great deal of parameter estimation and have little to no connection with the features of operational usage and practical testing procedures.

Reliability analysis using mission profiles has also been investigated in other areas such as power electronics and embedded systems, where workload and environmental changes have a great impact on the system's reliable and unreliable aspects of the system's dependability. In spite of the significance of the usage-driven reliability assessment of these studies, the approaches to it are domain-specific and cannot be unambiguously applied to general-purpose software systems [15]-[18] with their complex internal structure and components of heterogeneous nature.

The machine learning methods have received growing popularity with regard to software defect prediction and reliability evaluation. Better defect prediction accuracy has been shown by the hybrid optimization-based methods that use swarm intelligence-based algorithms and statistical estimation techniques in conjunction with the methods of estimating defects [8]. Equally, dynamic profiling and metrics have been used in detecting fault-prone elements and code blocks that are resource-intensive [5]. Such techniques are efficient in exploiting such structural properties as complexity, size and control flow, but normally ignore run-time frequency usage and usage behavior.

Moreover, the current models of defect prediction, in most cases, take care of their predictions and classification accuracy only and do not relate the results to a larger reliability engineering context. Consequently, there are no systematic

testing or maintenance decisions being informed by the likelihood of execution or the criticality of the operations of the system, and they are restricted in the overall reliability of the system.

Based on the analysis above, some of the shortcomings in the literature can be determined. First, the majority of operational profile-based methods have the assumption of fixed and correct usage models, which cannot overcome the dynamic and changing operational behavior. Second, methods of reliability modeling and machine learning-based methods of defect prediction are primarily researched separately, without a framework that combines operational frequency, structural complexity, and fault dependency. Third, very little consideration has been given to level-wise or package-level reliability testing, which is important in current component-based software systems.

Such shortcomings drive the suggested Level-Wise Composite Software Reliability method with the help of Operational Profile Assessment (LCSR-OPE), which incorporates operational modeling, probabilistic analysis, and machine learning-based fault recognition to offer a complete and adaptive reliability evaluation methodology.

TABLE I. RELATED WORK SUMMARY REPORT

Ref	Methodology Focus	Operational Profile usage	Granularity Level	Learning / Modeling Technique	Limitations
[1]	Adaptive testing with coverage spectrum	Explicit	Program-level	Statistical + coverage-based	No package-level analysis; assumes static OP
[6],[7]	Reliability growth under uncertainty	Implicit	System-level	Uncertain differential equations	Disconnected from runtime usage patterns
[8]	Defect prediction	Not considered	Module-level	Hybrid PSO–SSA optimization	No operational prioritization
[5]	Static profiling	Not considered	Function-level	Static resource analysis	Ignores execution frequency
[3],[4]	Mission-profile reliability	Explicit	System-level	Analytical reliability modeling	Domain-specific; not general software
Proposed	Composite reliability assessment	Explicit & dynamic	Package-level	Probabilistic modeling + XGBoost	Addresses prior gaps

The existing software reliability and defect prediction methods have some severe constraints, as summarized in Table I. Testing frameworks based on operational profiles are effective at prioritizing the test cases in accordance with the frequency of usage, but otherwise are limited to program-level analysis and often assume that operational profiles are static and fully known [1]. Uncertainty-theoretic reliability-growth models provide powerful mathematical tools to process epistemic uncertainty but are far weaker in relation to the operational behavior of runtime behavior, and experimental testing processes at a practical level of operations and experimentation on business properties and systems of operations and experimentation as a whole [6],[7]. Machine-learning-driven defect forecasting methods have high classification, but do not consider the dynamic indicators of performance in terms of the execution likelihood or operational significance, and therefore cause poor prioritization of testing and maintenance activities [8]. Although mission-profile-oriented reliability studies consider the variability in usage, most of them are domain-specific and cannot be directly applied to general-purpose software systems in the first place [3],[4]. Taken together, all these restrictions serve to highlight that there is no single framework that can both coordinate dynamic operational profiles, package-level structural metrics, probabilistic modelling and learning-based fault classification. It is this gap that drives the proposed Level-Wise Composite Software Reliability approach based on Operational Profile Evaluation (LCSR -OPE), which attempts to address such shortcomings by a novel adaptive, multi-granularity, and data-driven reliability assessment approach.

III. METHODOLOGY

This part outlines the methodological basis of the research. The traditional design of the software reliability evaluation is presented at the first stage in order to define the limitations of the design. Then the Level-Wise Composite Software Reliability framework based on Operational Profile Evaluation (LCSR -OPE) is described in more detail, including its architectural components, analytical pipeline and the combination of operational modeling, probabilistic analysis, and machine learning-based fault classification.

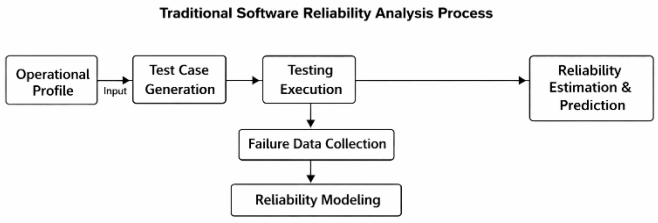


Fig. 1. Traditional software reliability analysis process.

The traditional Software Reliability Evaluation Architecture involves assessing the software's reliability based on its defect density and dependability. The classical architecture of a Software Reliability Assessment entails the evaluation of the reliability of the software in light of the density and the dependability of the defects in the software.

As shown in Fig. 1, the existing software reliability analysis process is typically well-spread in the operational testing-based reliability engineering. The operational profile of this paradigm determines the estimation of software reliability through an assumed operational profile defining the frequency of expected utilization of system functions. Under this profile, test cases are generated and run in a systematic manner, and failure data are

accumulated to derive estimates of reliability metrics using a statistical or probabilistic model of reliability.

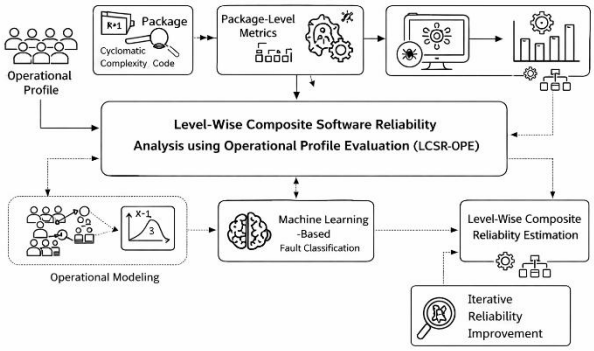


Fig. 2. Proposed framework: LCSR-OPE.

Although this methodology provides an organized testing process, it is presented with a number of limitations. The first is a statistical formulation of operations profile, which, first of all, assumes that the operational profile is static and known dynamically before deployment, an assumption that is never likely to be represented in the operational patterns. Second, reliability is usually estimated on the system level, which views software as a monolith and ignores package and module non-homogeneity. Third, fault detection is based on perceived failures during testing and does not make use of structural complexity measures and learning-based fault detection mechanisms in proactively detecting fault-prone components. As a result, the allocation of testing resources can be done efficiently, resulting in the inaccuracy of estimating reliability and the inability to mitigate faults in a slow way.

These limitations spur the establishment of an adaptive, fine-grained, and intelligence-assisted reliability paradigm which can integrate operational variability, structural feature and data-based fault analysis. This section presents a proposed overview of the LCSR-OPE Framework.

The suggested Level-Wise Composite Software Reliability model with the utilization of Operational Profile Evaluation (Level-Wise Composite Software Reliability LCSR-OPE), depicted in Fig. 2, is envisioned as a comprehensive reliability analysis tool, which combines operational usage modeling, package-level metric analysis, probabilistic prioritization and machine-learning-based fault classification. In contrast to traditional pipelines, LCSR-OPE will be based on a framework-oriented architecture which enables multi-level reliability analysis and continuous enhancement.

The framework consists of 5 closely related elements, including (1) operational profile modeling, (2) package-level metric extraction, (3) level-composite reliability analysis, (4) machine-learning-based fault classification and (5) reliability estimation through iterative refinement. The points are explained elaborately below.

The modeling of operational profiles summarizes the probabilistic distribution of the software usage based on the observed or inferred run-time behavior. The user interactions, execution traces, and session-level related data are investigated to build operational models that measure the frequency and duration of the software operations. These profiles are not considered static, but they are updated to capture changing user behavior and the nature of the workload.

Suppose that the frequency of the execution of the operation numbered i is denoted by the λ_i and that the average time of the session taken by the operation is denoted by the τ_i. Probabilistic density functions based on these parameters are used to determine the operational value of each operation. Reliability assessment and testing of operations are considered more important when operations have higher levels of probability densities.

In order to enable the process of fine-grained reliability analysis, the LCSR-OPE uses package-level software measures to record structural and complexity features of the codebase. Some metrics are obtained, including lines of code, cyclomatic complexity, the number of branches, the number of operators and design complexity of each package. These metrics provide information on fault prone and maintenance effort that would not be available in a black box alone testing.

The clustering techniques used to build the packages into profiles are based on the similarity between the metrics, and they provide the opportunity to model the heterogeneity between the software components. This level-wise abstraction fills in between system-level reliability estimation and module-level fault analysis.

To determine the IMCC's reliability, we will perform another analysis (composite reliability) at the level of a good item required for inclusion in this scale. To establish the reliability of the IMCC, we shall carry out another test (composite reliability) at the level of a good item that should be included in this scale.

The most crucial part of the suggested framework is the level-based composite reliability analysis module, which synthesizes the operational importance and structural complexity to make a priority on testing and reliability assessment. Given each package profile, operational probability densities are added with the metric-derived risk indicators to compute composite reliability weights.

The setting of tests and analysis is taken on a hierarchical basis, where the first packages to be set are the ones that have the greatest composite risk and significance in operations. The given strategy guarantees that the elements that are most likely to affect the perceived reliability of the system are tested initially, which leads to increased efficiency of fault detection and use of resources.

Somehow, this classification may be considered somewhat flawed and merely a procedural progression, yet its efficiency and dependability remain to be evaluated. In some way, this classification can be viewed as a somewhat faulty and procedural advancement, but its effectiveness and reliability still have to be assessed.

LCSR-OPE combines a machine-learning-based fault classification block in order to increase the accuracy of fault detection and prediction. The input feature set used to train an

ensemble-based classifier is made up of structural measures, attributes of operation, and execution statistics. XGBoost algorithm is chosen in this investigation because of its strength, scalability, and its ability to model nonlinear interactions between the features.

The classifier identifies packages as fault-prone or non-fault-prone, thus allowing prior identification of components on which reliability is crucial. Analysis results concerning classification are fed back to the analysis of composite reliability to refine testing priorities and estimation of reliability through repetition.

The second subtopic, Reliability Estimation and Refinement, is also an iterative process. The next subtopic is Reliability Estimation and Refinement, which is also an iterative process. Under the directions of taking fault data and level-level analysis, the reliability measure of each package is then summed to give some system-level reliability data. Fault correction and re-testing are carried out in a loop, and new execution logs are re-assessed to determine the improvement of reliability. This refinement loop, based on feedback, allows LCSR-OPE to adjust itself to these operational behavior modifications and code changes as time goes on, and thus, one of the major constraints of traditional reliability models is alleviated.

LCSR-OPE also overcomes the main challenges associated with the existing methods by combining dynamic operational profiles, package-level measurements, probabilistic modelling, and fault-classification systems based on machine learning within a unified system. The suggested method allows to adaptively evaluate reliability, allocating testing resources reasonably, and provides an enhanced level of fault detection, which makes it fit the modern component-based and evolving software systems.

Algorithm 1: Operational Profile Construction

```
Initialize: User execution logs
Compute: Operational Profile
While (λ=1) do
    For (τ=1) do
    Update
        Extract operation frequency λ and session time τ
        Compute probability density function (PDF) for each operation
            If (λ= τ) then
            Normalize operational probabilities
    End     End
    End
```

The design and integration of algorithms is a sub-task that takes place when the system must customize or tailor the language to its specific application. Algorithms Design and Implementation Algorithms design and Integration are a sub-task that occurs when the system needs to customize or adapt the language to a particular use.

The suggested LCSR-OPE model is implemented in a set of interconnected algorithms, which jointly perform the operations of operational modeling, level-wise reliability verification, fault categorization, and refinement. Algorithms 1-5 are written in a manner that allows them to be scaled, adaptive as well as analytically sound, with a high degree of independence and interdependencies.

The operational profile is formally constructed in Algorithm 1, which retrieves the frequency of execution and session-wise information of the runtime logs. This algorithm gives the probabilistic basis of the framework as it balances the probability of usefulness, whose direct impact is on the case of reliability priority. The operational profile obtained in this manner contrasts with traditional techniques, which suppose that the profile is definitely constant, whereas the operational profile here employs dynamic updates as behavior changes in use.

In Algorithm 2, package-level metric profiling is focused on representing structural and non-structural aspects of the software system. The algorithm will also support level-based abstraction of packages sharing comparable metric signatures, allowing reliability analysis to be done across monolithic system level estimation and instead perform fine-grained, component-based analysis.

Algorithm 2: Package-Level metric profiling

```
Initialize: Software packages
Compute: Package Profiles
While (λ=1) do
    For (τ=1) do
    Compute
        Extract structural metrics (LOC, Cyclomatic Complexity, Branch Count, Operators)
        Compute probability density function (PDF) for each operation
        Normalize metrics
            If (λ= τ) then
            Cluster packages into profiles using K-means
    End     End
    End
```

Algorithm 3 takes the output of the operational profile and package-level metrics to carry out level-composite reliability assessment. This algorithm is used to compute scores of composite risk and importance, which are used to determine the order in which to generate and execute test cases. The hierarchical prioritization mechanism permits that the testing resources should be directed to packages that have the greatest operational effects and vulnerability to faults.

Algorithm 3: Level-wise Composite reliability evaluation

```
Initialize: Package profiles and operational weights
Compute: Package Profiles
While (λ=1) do
    For (τ=1) do
    Compute
        each profile do
        Compute composite reliability weight
        Prioritize packages based on composite score
            If (λ= τ) then
            Generate test cases accordingly
    End     End
    End
```

Machine-learning based fault classification step is presented in Algorithm 4, through which both structural and operational characteristics are utilized concurrently to reveal packages that have high fault potentials. The use of an ensemble learning model improves predictive robustness and, in addition, allows proactive improvement of reliability by highlighting critical components before large failures occur.

Algorithm 4: Level-wise Composite reliability evaluation

```
Initialize: Training datasets
Compute: Risk scores
While (λ=1) do
    For (τ=1) do
    Update
        Extract features from metrics and operational data
        Train XGBoost classifier
        Compute composite reliability weight
        Predict fault-prone packages
            If (λ= τ) then
            Update the reliability test scores accordingly
    End     End
    End
```

Algorithms used to accomplish the goals of reliability estimation include iterative reliability refinement, which is in Algorithm 5 and closes the loop of feedback between testing, fault correction and reliability estimation. This algorithm would ensure continuous enhancement in reliability and flexibility to a changing usage pattern and code evolution by re-examining revised execution logs and improved operational profiles.

Algorithm 5: Iterative Reliability refinement

```
Initialize: Execute prioritized test cases
    Compute: Reliability refinement
    While (λ=1) do
        For (τ=1) do
        Update
            Collect failure logs
            Update fault repository
            Compute composite reliability weight
            Predict fault-prone packages
                If (λ= τ) then
                Recompute reliability metrics
                Repeat until reliability threshold is satisfied
        End     End
        End
```

Algorithms 1 to 5 together form a unified algorithm pipeline that can fit within the suggested architecture of the framework. Their combination allows adaptive, level-based, and data-driven software reliability measurement, thus overcoming the main shortcomings of the traditional reliability models that have been noted in the literature.

IV. Results and Discussion

This part will evaluate the efficiency of Level-Wise Composite Software Reliability (LCSR-OPE) method. The empirical research will focus on fault-detection capability, accurately estimate reliability, effectiveness of prioritization of operations, and its performance in relation to traditional reliability assessment methods.

Data were used on experimental procedures involving publicly available NASA software defects data, which were JM1, PC1, KC1, and KC2. These databases include package-level architectural measures and defect tagging, making them suitable to assess reliability-focused fault analysis models. All features were normalized using Min-Max scaling before the analysis of features to make sure that there is no discrepancy in contribution between metrics.

The framework proposed is a combination of an operational modeling, a machine-learning-based fault classification model, and a package profiling model based on clustering. XGBoost has been chosen as the master classifier due to its robustness, scalability and ability of the model to learn the nonlinear interaction among the software measures.

In this section, the proposed LCSR-OPE framework has been interpreted in full detail by focusing on the mechanisms that explain all the experimental outcomes and trying to explain their role in supporting the effectiveness of the offered methodology.

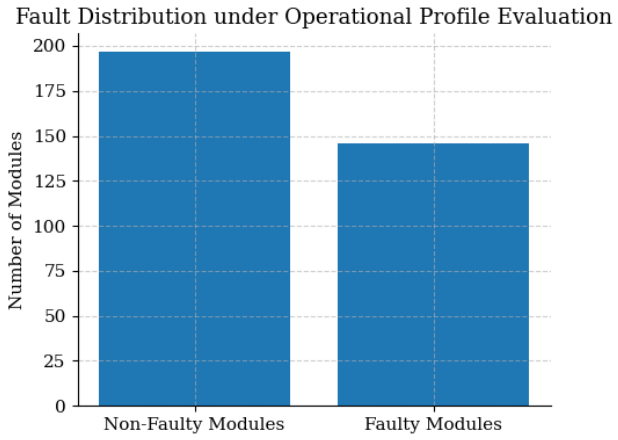


Fig. 3. Fault distribution result.

Fig. 3 in the Appendix shows the distribution of defective and non-defective software modules detected during an analysis based on operational profiles. The reported trend can be explained by the prioritization of introducing the test performance conducted in accordance with operation frequency, thus increasing the intensity of evaluation of modules with high probabilities of execution. Thus, we notice faults more often in dominating modules in operations, showing a non-homogeneous distribution of faults that support a high impact on runtime usage behavior on defect manifest. The given empirical observation supports the suggested structure by showing that the operation-profile assessment allows the detection of faults specifically, therefore, making the reliability measurement of the meaning more precise in comparison to the uniform testing method.

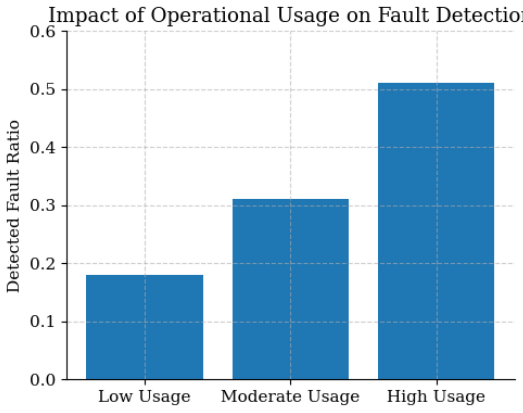


Fig. 4. Impact of operational usage result.

Fig. 4 shows the effect of the operation usage levels on fault detection rates. The analysis consists of classifying software modules into high, low and intermediate usage categories in terms of the frequency of their execution and then comparing their respective fault ratios. The additional increases in fault ratios with the increased use of modules prove that modules that are used more frequently are prone to failure. This justifies the main assumption of the framework that the reliability analysis and the prioritization of the testing must be based directly on the operational behavior, through which a more efficient determination of the components that have a critical impact on reliability can be conducted.

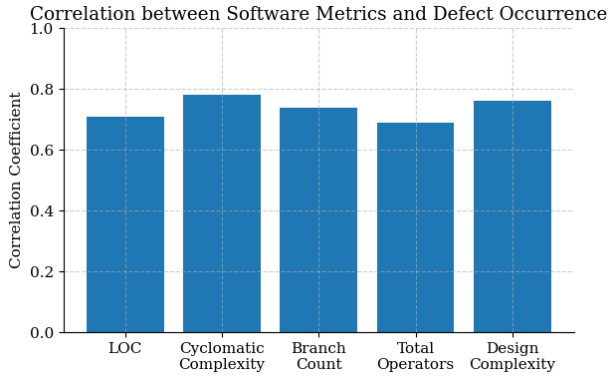


Fig. 5. Correlation between software metrics results.

Fig. 5 shows the relationship between software metrics of structure and defects. The correlation analysis compares the values of the metrics that have been extracted at the package level to the observed defect labels. Cyclomatic complexity, the number of branches and the number of branches show a high positive correlation with defects, indicating that structurally complex modules are more prone. This finding supports the LCSR-OPE model by supporting the idea of introducing package-level structural measurements in the composite reliability modelling and arguing in favor of their usefulness as discriminative predictors to classify faults.

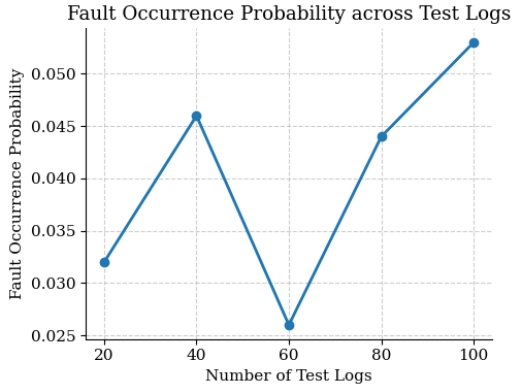


Fig. 6. Fault occurrence probability result.

Fig. 6 in the appendix, fault probability, illustrates the relationship between the probability of occurrence of the faults and the number of test logs that have been operationally prioritized and executed. The analysis results in a controlled variation of the probability of faults by progressively growing the number of such test cases and computing the rate of fault detection thereof. This behavior is realistic testing behavior, or more precisely, fault detection is not increasing monotonically with redundancy and decreasing returns. The result indicates the strength of an operational-profile-based strategy in the process of avoiding redundant test running without compromising efficiency in the detection of faults.

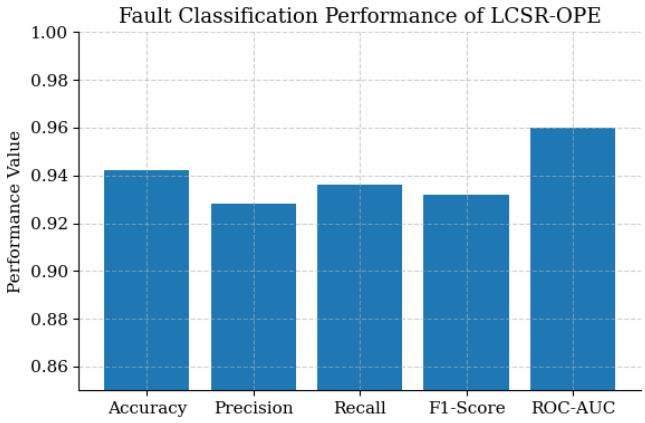


Fig. 7. Fault classification performance result.

The results of the fault classification of the proposed framework using the metrics of accuracy, precision, recall, F1-score, and ROC-AUC are shown in Fig. 7. They are obtained by using an XGBoost classifier, which is trained on both operational and structural metrics. The high results of all measures prove the ability of the classifier to properly distinguish between fault-prone and fault-free modules. This has a direct positive effect on the model in that it allows reliability-sensitive components to be identified well in advance, which will lead to better testing endeavors and reliability estimations.

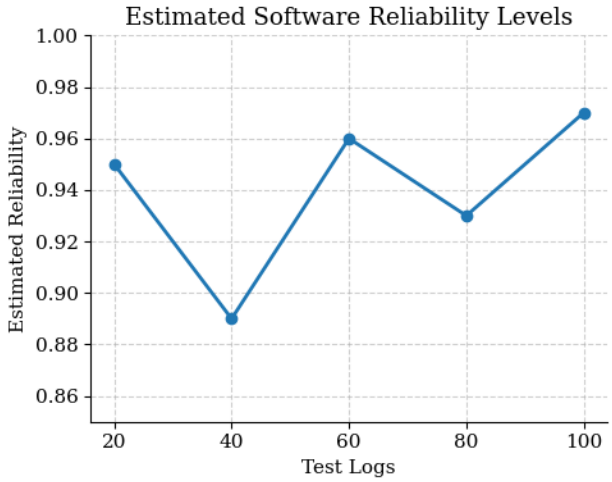


Fig. 8. Estimated software reliability result.

Fig. 8 shows the approximate level of reliability of software found through various testing conditions. Reliability is achieved by summarizing the results of fault detection of operationally important modules and recomputing reliability measures after fault removal. The noted increase in reliability as the progressive level of fault elimination takes place is akin to realistic software behavior and supports the level-wise composite reliability evaluation as successful in providing accurate and interpretable reliability estimates.

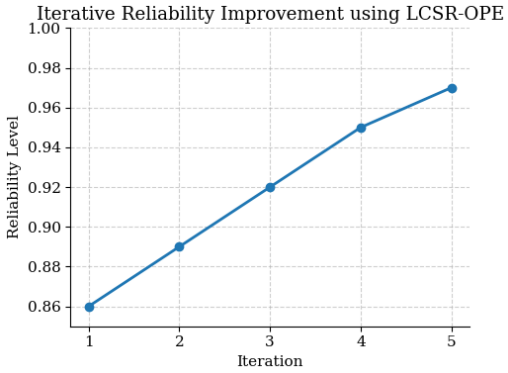


Fig. 9. Iterative reliability result.

Fig. 9 illustrates that the reliability of the software may be improved through a cycle of testing with subsequent fault-correction until the desired standard is achieved. The enhancement is inspired by a feedback-based refurbishment engine incorporated into the structure, whereby refurbished operational profiles and fault classification results are used to inform each subsequent testing round. The gradual increase in reliability attests to the fact that the framework facilitates the process of constant improvement of reliability, a requirement to develop software systems.

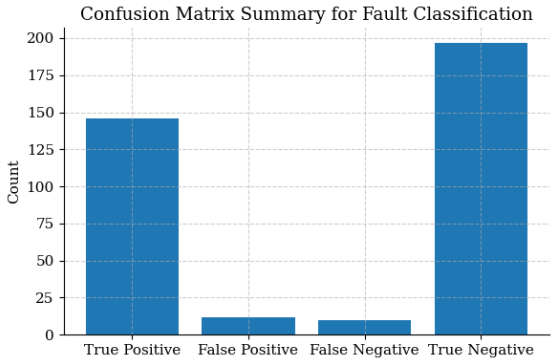


Fig. 10. Fault classification summarization result.

Fig. 10 represents the results of the confusion matrix of the fault classification process. The analysis also demonstrates a low false-positive and false-negative quality because the prediction of fault labels was compared with real instances of faults, showing that the classification performance is strong. Such an outcome guarantees that reliability estimation is not contaminated by misclassification error problems, hence the integrity of the composite reliability measurement is not compromised. Together, the combined findings support the LCSR -OPE framework as it shows that the combination of operational-profile assessment, package-level structural modeling, and machine-learning-based fault diagnosis does yield more precise, efficient and scalable software reliability.

A. Performance Analysis

Under this sub-section, the effective performance of the proposed LCSR-OPE framework in terms of the detection of faults, estimation of the reliability, the effectiveness for testing, and the assessment of sufficiency is considered. The analysis is based on quantifiable measures that are derived from the

experimental data and highlights the real-life advantages of the suggested methodology.

In terms of fault detection, the framework possesses a strong component of detecting components that are of critical importance to the reliability. The operational profile assessment is such that the higher the execution probability, the more dissimilar the level of scrutiny the module is provided and hence it improves the level of revealing failure in those parts that are operationally significant. The phenomenon is supported by observed fault distribution and operational impact metrics, which, in combination, indicate that the suggested strategy is helpful in reducing the inefficiencies of the uniform testing approaches.

In terms of fault classification performance, the machine-learning module incorporated in LCSR-OPE shows a high discriminative power since the accuracy, precision, recall, and ROC-AUC values were always high. The equal performance of these measures shows that the classifier also lowers the rate of false-positive and false-negative predictions. This factor plays a critical role in the assessment of reliability since a poor assignment of misclassification of fault-prone modules may result in poor reliability estimates and batch allocation of testing resources.

The level-wise composite analysis strategy by the framework also enhances the performance of the reliability estimation. The combination of operational and package-level structural risk offers by the framework reliable and interpretable reliability estimates without the comprehensive tests required. These empirical strong results of progressive reliability improvements with successive iterations indicate that the methodology enables a gradual enhancement of reliability with controlled testing overhead.

Another important performance dimension which is dealt with by the LCSR-OPE is testing efficiency. Operation-profile-based mechanism of prioritization focuses the test effort on the elements of the system which have the largest contribution to the perceived system reliability. As a result, more reliable results are achieved after a reduced number of tests runs than with the traditional methods that consider the novel components in a homogeneous set. This economic benefit is especially relevant to large, complex and dynamic software systems where it is not feasible to perform comprehensive testing.

In general, the analysis of the performance proves that the suggested LCSR-OPE model offers a strong and effective software reliability evaluation tool. The combination of operational behavior subsequent to structural complexity and learned fault patterns offers the framework an excellent ability to detect faults and also to determine reliability, which makes it quite suitable in modern component-based software systems.

B. Comparison Analysis

This subsection will provide a comparative analysis of the proposed LCSR-OPE framework with a representative state-of-the-art software reliability and defect prediction methods, such as black-box reliability models (BBR), prediction of defects by using static metrics (SMDP), prediction by using uncertainty (URM), coverage and operational profile-based testing (COPT), and prediction by using machine learning (MLDP), based on the use of static features [20],[22]. The comparison focuses on the efficiency of the use of test resources, classification accuracy, ROC–AUC, and multi-metric.

Fig. 11, in the Appendix demonstrate the efficiency of the exploitation of test resources in the studied models. Black-box reliability models and application-independent metric-based methods have relatively low testing efficiency because they use either uniform testing strategies or structure-based testing strategies that do not consider runtime usage behavior. Uncertainty-based models [23],[24]. Improvements of moderate magnitude are obtained using probabilistic reasoning or coverage-based approaches. Conversely, the suggested LCSR-OPE system has the greatest testing efficiency, which focuses on test execution based on operational importance and trained error behavior. This result supports the fact that the combination of operational profile assessment and smart fault classification results in more efficient use of the scarce testing resources.

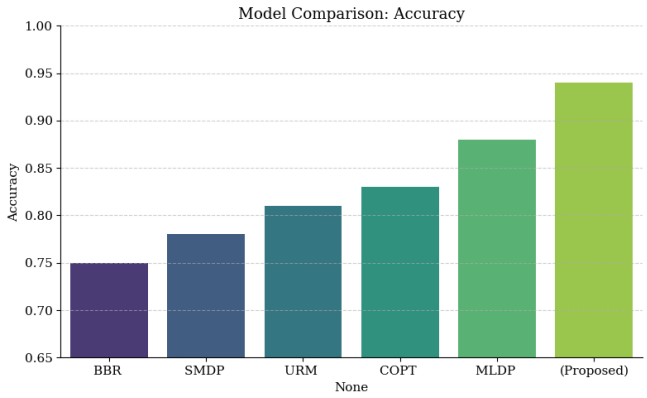


Fig. 11. Model comparison: accuracy.

The result of the classification accuracy, as shown in Fig. 12, shows that there is a steady improvement in the classification accuracy of the traditional and the learning-based methods. Although MLDP is better than non-learning models in that it uses machine learning on static measures, it is limited by a lack of operating conditions. The LCSR-OPE framework proposed is at the top in accuracy since structural complexity and operational behavior are modeled together; hence, one can make a more stable prediction as to whether a particular module is fault-prone or non-fault-prone. This finding is an indication that operational awareness is a defining parameter in boosting the accuracy of defect prediction beyond mere static analysis in itself [8],[21].

The ROC-AUC values of the learned models are contrasted in Fig. 12, which essentially indicates the discriminative ability that each model has at different classification levels. The use of traditional reliability models [26]-[34] has a low ROC–AUC because they traditionally use an aggregation of failure statistics. The methods based on the use of machine learning enhance discriminatory performance; however, the method suggested in the present study, under the label of LCSR-OPE yield the largest ROC-AUC, indicating high robustness and stability in the identification of faults. This is due to the fact that its operational profile features have been incorporated, making the classes more separable as the emphasis is made on the parts of the execution that are vital.

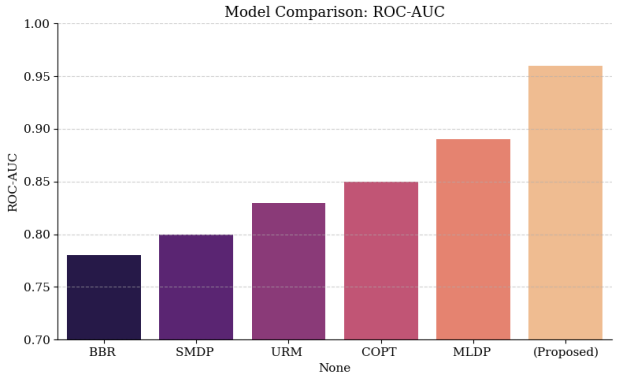


Fig. 12. Model comparison: ROC-AUC.

To be complete, a multi-metric comparison incorporating the accuracy, precision, recall, F1-score, ROC-AUC, and estimation of reliability is provided in Fig. 13. Though the quantitative judgment is not the purpose of radar plots, it gives a qualitative view of the behavior of a model in more than one dimension. The presented framework always prevails in comparison to other strategies in all analyzed indicators, which implies the mixed performance and the absence of the reliability estimation performance tradeoff to achieve higher classification performance. Such moderated behavior is fundamental to software reliability engineering, in which precise failure recognition and dependable reliability estimation should be performed to ensure quality of results, both at the same time and also independently of each other [1],[6].

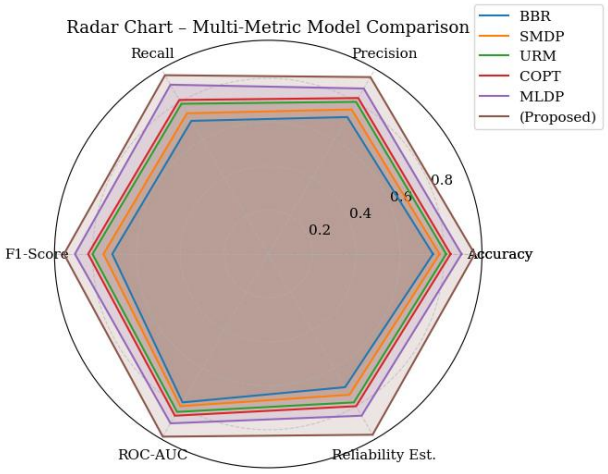


Fig. 13. Model comparison precision and recall.

In general, the comparative findings in Fig. 14 indicate that the proposed LCSR-OPE framework is superior to the state-of-the-art existing frameworks as it is capable of integrating operational profile assessment, package structural analysis, and fault classification with the help of machine learning. In contrast to traditional models [11],[12], which focus on reliability or defect prediction individually, the presented framework can provide a unified and adaptive solution, which can improve the effectiveness of testing, classification performance, and the accuracy of the estimation of reliability. Such results support the practical contribution and study benefits of the suggested solution in the current component-based software systems.

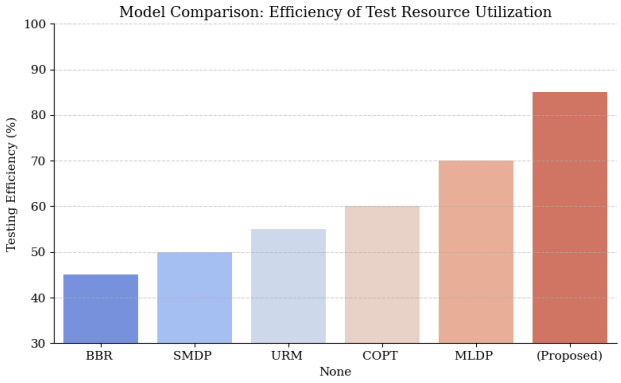


Fig. 14. Model comparison: efficiency of resource utilization.

V. Conclusion

The proposed framework introduces runtime usage behavior and structural risk as opposed to conventional reliability models that are based on uniform testing assumptions or the use of only metrics to rank testing and reliability analysis. Such integration will enable the identification of more reliability-critical components and the use of testing resources more efficiently.

As experimental evidence shows, the suggested LCSR-OPE scheme shows a high level of performance in response to state-of-the-art strategies in fault detection, classification, discriminative and testing efficiency. Combining the power of both operational characteristics and structural measures, the framework has always performed better than traditional black-box schemes as well as the traditional predictor schemes of the statistics of the structure and the sole machine-learning schemes. The level-wise composite reliability analysis also allows having interpretable and progressive estimation of reliability without tedious testing.

This comparative analysis validates the idea that the inclusion of operational awareness in machine-learning-based reliability assessment has a tangible benefit to contemporary component-based software systems. To a large extent, the proposed framework can enhance the accuracy of reliability estimation, as well as minimize the needless testing work, as it focuses on operationally relevant and structurally complex constituents. The above features make LCSR-OPE very appropriate to large and dynamic software systems when testing resources are limited.

The further development of the proposed framework will be directed towards a number of extensions. To determine first, there are the dynamic runtime telemetry and online learning mechanisms that will be used to integrate within supporting the continuity of the reliability assessment in an adaptive software environment. Second, the framework will be tested on other large-scale industrial data to also confirm its applicability to other areas of use. Lastly, the use of explainable machine-learning methods will be considered to contribute to the interpretability of fault-classification decisions and reliability analysis developer-centric.

References

[1] Bertolino, B. Miranda, R. Pietrantuono and S. Russo, "Adaptive Test Case Allocation, Selection and Generation Using Coverage Spectrum and

Operational Profile," in IEEE Transactions on Software Engineering, vol. 47, no. 5, pp. 881-898, 1 May 2021, doi: 10.1109/TSE.2019.2906187.

[2] N. Qamar and A. A. Malik, "A Quantitative Assessment of the Impact of Homogeneity in Personality Traits on Software Quality and Team Productivity," in IEEE Access, vol. 10, pp. 122092-122111, 2022, doi: 10.1109/ACCESS.2022.3222845.

[3] Z. Chen, J. Xu, P. Davari and H. Wang, "A Mixed Conduction Mode-Controlled Bridgeless Boost PFC Converter and Its Mission Profile-Based Reliability Analysis," in IEEE Transactions on Power Electronics, vol. 37, no. 8, pp. 9674-9686, Aug. 2022, doi: 10.1109/TPEL.2022.3153558.

[4] Kim and H. Yang, "Reliability Optimization of Real-Time Satellite Embedded System Under Temperature Variations," in IEEE Access, vol. 8, pp. 224549-224564, 2020, doi: 10.1109/ACCESS.2020.3044044.

[5] J. Correas, P. Gordillo and G. Román-Díez, "Static Profiling and Optimization of Ethereum Smart Contracts Using Resource Analysis," in IEEE Access, vol. 9, pp. 25495-25507, 2021, doi: 10.1109/ACCESS.2021.3057565.

[6] Z. Liu, S. Yang, M. Yang and R. Kang, "Software Belief Reliability Growth Model Based on Uncertain Differential Equation," in IEEE Transactions on Reliability, vol. 71, no. 2, pp. 775-787, June 2022, doi: 10.1109/TR.2022.3154770.

[7] Z. Liu and R. Kang, "Imperfect Debugging Software Belief Reliability Growth Model Based on Uncertain Differential Equation," in IEEE Transactions on Reliability, vol. 71, no. 2, pp. 735-746, June 2022, doi: 10.1109/TR.2022.3158336.

[8] L. Yang, Z. Li, D. Wang, H. Miao and Z. Wang, "Software Defects Prediction Based on Hybrid Particle Swarm Optimization and Sparrow Search Algorithm," in IEEE Access, vol. 9, pp. 60865-60879, 2021, doi: 10.1109/ACCESS.2021.3072993.

[9] J. A. Lima and G. Elias, "Selection and allocation of people based on technical and personality profiles for software development projects", Proc. 15th Latin Amer. Comput. Conf. (CLEI), pp. 1-10, Sep. 2019.

[10] E. Weilemann, "A winning team—What personality has to do with software engineering", Proc. IEEE/ACM 41st Int. Conf. Softw. Eng. Companion Proc. (ICSE- Companion), pp. 252-253, May 2019.

[11] Z. Akarsu and M. Yilmaz, "Managing the social aspects of software development ecosystems: An industrial case study on personality", J. Softw. Evol. Process, vol. 32, no. 11, pp. e2277, Nov. 2020.

[12] F. F. Mendes, E. Mendes and N. Salleh, "The relationship between personality and decision-making: A systematic literature review", Inf. Softw. Technol., vol. 111, pp. 50- 71, Jul. 2019.

[13] V. C. Kuko, D. Music, Z. Vejzovic and J. Azemovic, "Model of foresight work habits of agile software team members by personality traits", Proc. Int. Conf. Comput. Inf. Telecommun. Syst. (CITS), pp. 1-5, Aug. 2019.

[14] N. Qamar and A. A. Malik, "Birds of a feather gel together: Impact of team homogeneity on software quality and team productivity", IEEE Access, vol. 7, pp. 96827-96840, 2019.

[15] M. O. Yilmaz, R. V. O'Connor, R. Colomo-Palacios and P. Clarke, "An examination of personality traits and how they impact on software development teams", Inf. Softw. Technol., vol. 86, no. 1, pp. 101-122, Jun. 2017.

[16] R. Poonam and C. M. Yasser, "An experimental study to investigate personality traits on pair programming efficiency in extreme programming", Proc. 5th Int. Conf. Ind. Eng. Appl. (ICIEA), pp. 96-99, Apr. 2018.

[17] Z. U. Kamangar, U. A. Kamangar, Q. Ali, I. Farah, S. Nizamani and T. H. Ali, "To enhance effectiveness of crowdsource software testing by applying personality types", Proc. 8th Int. Conf. Softw. Inf. Eng., pp. 15-19, Apr. 2019.

[18] A.S. Barroso, K. H. de J. Prado, M. S. Soares and R. P. C. do Nascimento, "How personality traits influences quality of software developed by students", Proc. 15th Brazilian Symp. Inf. Syst., pp. 1-8, May 2019.

[19] N. Qamar and A. A. Malik, "Impact of design patterns on software complexity and size", Mehran Uni. Res. J. Eng. Technol., vol. 39, no. 2, pp. 342-352, Apr. 2020.

[20] M. Shameem, C. Kumar and B. Chandra, "A proposed framework for effective software team performance: A mapping study between the team members' personality and team climate", Proc. Int. Conf. Comput. Commun. Autom. (ICCCA), pp. 912-917, May 2017.

[21] A.S. Barroso, J. S. Madureira, T. D. S. Souza, B. S. D. A. Cezario, M. S. Soares and R. P. C. do Nascimento, "Relationship between personality traits and software quality–big five model vs. object-oriented software metrics", Proc. 19th Int. Conf. Enterprise Inf. Syst., pp. 63-74, 2017.

[22] Amin, S. Basri, M. Rehman, L. F. Capretz, R. Akbar, A. R. Gilal, et al., "The impact of personality traits and knowledge collection behavior on programmer creativity", Inf. Softw. Technol., vol. 128, Dec. 2020.

[23] R. Yan, W. R. Ringwald, J. Vega, M. Kehl, S. W. Bae, A. K. Dey, et al., "Exploratory machine learning modeling of adaptive and maladaptive personality traits from passively sensed behavior", Future Gener. Comput. Syst., vol. 132, pp. 266-281, Jul. 2022.

[24] M. A. Kosan, H. Karacan and B. A. Urgen, "Predicting personality traits with semantic structures and LSTM-based neural networks", Alexandria Eng. J., vol. 61, no. 10, pp. 8007- 8025, Oct. 2022.

[25] K. Yao and B. Liu, "Parameter estimation in uncertain differential equations", Fuzzy Optim. Decis. Mak., vol. 19, no. 1, pp. 1-12, 2020.

[26] Z. Liu, "Generalized moment estimation for uncertain differential equations", Appl. Math. Comput., vol. 392, 2021.

[27] X. Yang, Y. Liu and G. Park, "Parameter estimation of uncertain differential equation with application to financial market", Chaos Solitons Fractals, vol. 139, pp. 110026, 2020.

[28] T. Yang, "Comparative study on the performance attributes of NHPP software reliability model based on weibull family distribution", Int. J. Performability Eng., vol. 17, no. 4, pp. 343-353, Apr. 2021.

[29] M. R. Lyu, Handbook of Software Reliability Engineering, IEEE Computer Society Press and McGraw-Hill, 1996.

[30] A. Avizienis, J.-C. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing," IEEE Transactions on Dependable and Secure Computing, vol. 1, no. 1, pp. 11–33, 2004.

[31] N. E. Fenton and M. Neil, "A critique of software defect prediction models," IEEE Transactions on Software Engineering, vol. 25, no. 5, pp. 675–689, 1999.

[32] T. Menzies, A. Butcher, A. Marcus, T. Zimmermann, and D. Cok, "Local versus global models for effort estimation and defect prediction," IEEE Transactions on Software Engineering, vol. 39, no. 6, pp. 822–834, 2013.

[33] S. Kim, H. Zhang, R. Wu, and L. Gong, "Dealing with noise in defect prediction," Proceedings of the 33rd International Conference on Software Engineering (ICSE), ACM, 2011.

[34] B. Kitchenham, D. Budgen, and O. P. Brereton, "Using mapping studies as the basis for further research," Information and Software Technology, Elsevier, vol. 53, no. 6, pp. 638–651, 2011.

Robust Medical Image Reconstruction Using a Self-Evolving Encoder–Decoder and Adaptive Convolutional Power Scaling

Dhanusha P B[1a, b]*, Dr. J. Bennilo Fernandes[2], Dr. A. Muthukumar[3], Dr. A. Lakshmi[4]
Research Scholar, Department of Electronics and Communication Engineering, Kalasalingam Academy of Research and Education, Virudhunagar, Tamilnadu, India[1a]
Assistant Professor, Department of Electronics Engineering, SAINTGITS College of Engineering, Kottayam, Kerala, India[1b]
Assistant Professor, Department of Electronics and Communication Engineering, Kalasalingam Academy of Research and Education, Virudhunagar, Tamil Nadu, India[2]
Professor, Department of Electronics and Communication Engineering, Sri Raaja Raajan College of Engineering and Technology, Amaravathipudur, Tamilnadu, India[3]
Professor, Department of Electronics and Communication Engineering, Ramco Institute of Technology, Rajapalayam, Tamilnadu, India[4]

***Abstract*—Robust medical image reconstruction is a critical requirement for accurate diagnosis and clinical decision-making, particularly when images are affected by degradation, noise, or low resolution. Conventional encoder–decoder-based reconstruction methods compress input images into low-dimensional representations and subsequently decode them into high-resolution outputs; however, such approaches often suffer from artifacts and loss of fine anatomical details under severe degradation. To address these limitations, this work proposes a robust medical image reconstruction framework using a self-evolving encoder–decoder and adaptive convolutional power scaling. The proposed super-resolution model incorporates a dynamic encoder and decoder that adaptively evolve during training to capture color contrast, structural similarity, and high-frequency details from medical images. An MLP enhanced with an adaptive power flex layer is embedded within the reconstruction pipeline, enabling learnable power-based feature scaling through weight-wise modulation and initialization. This mechanism improves feature discrimination and stabilizes the reconstruction of subtle anatomical structures. The DRIVE and CHASE_DB1 retinal image datasets are employed for experimental validation, with appropriate preprocessing applied before training and testing. The selected images are processed through the proposed super-resolution model, and performance is quantitatively evaluated using PSNR, SSIM, sensitivity, and specificity metrics. Experimental results demonstrate that the proposed method achieves significant improvements in reconstruction quality and robustness compared to existing approaches, yielding enhanced perceptual quality and structural fidelity in reconstructed medical images. These findings indicate that the proposed self-evolving encoder–decoder with adaptive convolutional power scaling is well-suited for reliable medical image reconstruction applications.**

***Keywords*—Dynamic encoder and decoder; power flex model layer; high resolution images; weight initialization; adaptive convolutional power scaling**

I. Introduction

Medical image reconstruction and super-resolution play a vital role in enhancing image quality for accurate diagnosis, treatment planning, and clinical decision-making. However, medical images acquired through modalities such as magnetic resonance imaging (MRI), computed tomography (CT), and retinal fundus imaging often suffer from low resolution, noise, and acquisition constraints, which can obscure fine anatomical details. Early learning-based approaches for image super-resolution demonstrated the effectiveness of deep convolutional neural networks (CNNs) in reconstructing high-resolution images from degraded inputs [1]. Subsequent advancements extended these ideas to three-dimensional medical data, enabling improved spatial detail preservation in MRI reconstruction [2]. Traditional model-based methods incorporating low-rank and total variation regularization further improved reconstruction fidelity but remained computationally intensive and limited in adaptability [3].

Encoder–decoder architectures have become the backbone of many medical image processing tasks due to their strong representation learning capability [4]. Despite their success, conventional encoder–decoder frameworks often struggle with robustness when dealing with severely degraded inputs, leading to artifacts and loss of high-frequency details. To address domain variability and distribution shifts, generative and domain-adaptive learning techniques have been explored [5]. Meanwhile, perceptual quality metrics such as structural similarity index (SSIM) [6] and deep feature-based perceptual measures [7] have become standard tools for evaluating reconstruction performance beyond pixel-wise fidelity.

In retinal imaging, accurate reconstruction and enhancement are particularly important for vessel analysis and disease screening. Benchmark datasets such as DRIVE and CHASE_DB1 have been widely used to evaluate reconstruction

and segmentation performance in retinal images [8], [9]. Recent advances have introduced transformer-based and hybrid CNN–Transformer architectures to capture both local and global dependencies in medical image super-resolution [10], [12], [13]. Attention-based mechanisms have further enhanced feature discrimination, leading to improved reconstruction quality and robustness [15], [21].

More recently, adaptive and dynamic network designs have gained significant attention. Dynamic convolution and shuffle-based parallel architectures enable input-dependent feature modulation, improving generalization and reconstruction performance across varying image conditions [16], [29], [30], [31]. In parallel, iterative and self-collaborative reconstruction strategies have been proposed to progressively refine high-resolution outputs using reconstruction priors [20], [27]. Diffusion-based and implicit neural representation models have further expanded the design space for medical image super-resolution, demonstrating promising results in multimodal and arbitrary-scale reconstruction tasks [18], [19]. Wan et al. introduced a cycle-constraint adversarial network for retinal image enhancement, enabling unpaired learning while effectively improving contrast, illumination consistency, and preservation of fine vascular structures, thereby supporting more reliable automated retinal analysis [31].

The majority of modern picture super-resolution techniques rely on static convolutional processes and fixed architectural parameters during training and inference. These networks only train a single mapping function, regardless of variations in contrast heterogeneity, modality-specific noise, anatomical complexity, or degradation levels. However, depending on the scanner, acquisition technique, and patient condition, medical pictures in real-world clinical settings exhibit a range of unexpected degradations. However, rather than drastically altering the encoder-decoder's reconstruction behaviour, dynamic convolution techniques and adaptive attention methods primarily focus on spatial-channel reweighting. Because of this, current frameworks sometimes produce artefacts or over-smoothed outcomes when dealing with unseen clinical variations, and they may also have trouble generalizing across deterioration intensities.

Therefore, the primary technical gap is the absence of a reconstruction framework that can dynamically adjust its feature transformation capability to shifting deterioration characteristics throughout training. Current models adjust weights to increase resilience and stability, but they don't explicitly incorporate a technique that allows for structural evolution of the encoder–decoder or adaptively controls convolutional feature power. To bridge this gap, this study proposes an adaptive convolutional power scaling self-evolving encoder-decoder system. Convolutional responses are altered using learnable power-based transformations, which change the expressiveness of features dependent on the complexity of degradation. By allowing the encoder–decoder to gradually alter during training, the proposed method maintains fine anatomical structures more successfully than static designs, enhances adaptability, and stabilizes reconstruction under extreme degradations. Comprehensive evaluations show that the proposed approach delivers exceptional numerical performance and perceptual quality in a range of medical imaging scenarios.

A. Paper Organization

The remainder of this study is organized as follows. Section 2 reviews related work on medical image reconstruction, super-resolution techniques, encoder–decoder architectures, and adaptive convolutional learning methods. Section 3 describes the proposed robust medical image reconstruction framework, including the self-evolving encoder–decoder architecture, adaptive convolutional power scaling mechanism, and overall network design. Section 4 details the experimental setup, including dataset descriptions, preprocessing steps, implementation details, training strategy, and evaluation metrics. Section 5 presents and discusses the experimental results, providing quantitative and qualitative comparisons with existing state-of-the-art methods. Finally, Section 6 concludes the study and outlines potential directions for future research.

II. Literature Review

Medical image reconstruction and super-resolution have been extensively studied to overcome limitations arising from low spatial resolution, noise, and acquisition constraints in medical imaging modalities. Early deep learning-based super-resolution methods demonstrated that convolutional neural networks (CNNs) can effectively learn nonlinear mappings between low- and high-resolution images, significantly outperforming traditional interpolation techniques [1]. These approaches laid the foundation for subsequent advances in learning-based reconstruction. To address modality-specific challenges, deep learning techniques were extended to medical imaging applications. Pham et al. [2] proposed a 3D convolutional neural network for brain MRI super-resolution, achieving improved spatial consistency across volumetric data. Similarly, Shi et al. [3] incorporated low-rank and total variation regularization to enhance MR image reconstruction, demonstrating improved edge preservation. Although effective, such regularization-based methods often require high computational cost and lack adaptability to diverse degradation patterns.

Encoder–decoder architectures have become a dominant paradigm in medical image analysis due to their ability to capture hierarchical features. The U-Net architecture introduced by Ronneberger et al. [4] has been widely adopted and extended for various biomedical imaging tasks. However, conventional encoder–decoder models often struggle to maintain robustness when reconstructing severely degraded images. To mitigate domain shifts and variability, adversarial learning and domain adaptation techniques have been explored, enabling improved generalization across datasets [5]. Evaluation of reconstructed image quality has evolved beyond pixel-wise fidelity metrics. The structural similarity index (SSIM) introduced by Wang et al. [6] has become a standard measure for assessing perceptual similarity, while deep feature-based perceptual metrics further improved the assessment of visual quality [7]. These metrics are particularly relevant in medical imaging, where preserving anatomical structures is critical.

Retinal imaging has served as a benchmark application for evaluating reconstruction and enhancement techniques. The DRIVE and CHASE_DB1 datasets have been widely used to assess vessel visibility and structural integrity in retinal images [8], [9]. Recent works have applied super-resolution techniques to retinal fundus images using generative adversarial networks, demonstrating notable improvements in visual quality and diagnostic relevance [11].

With the rise of attention mechanisms and transformer architectures, hybrid CNN–Transformer models have shown promising performance in medical image super-resolution. Zhou [10] introduced a multi-scale convolution-aided transformer framework to capture both local and global features. Chen et al. [12] proposed a mutual co-attention network for joint MR image reconstruction and super-resolution, achieving improved structural fidelity. Further advancements integrated gated fusion mechanisms and attention modules to enhance feature interaction and reconstruction robustness [13], [15], [21].

Recent research has increasingly focused on adaptive and dynamic networks. Adaptive dynamic shuffle convolutional parallel networks have demonstrated improved feature diversity and efficiency in super-resolution tasks [16], [30]. Dynamic convolution strategies, such as snake convolution, enable input-dependent kernel modulation, allowing better adaptation to structural variations within images [29]. Iterative and collaborative reconstruction frameworks have further enhanced performance by progressively refining high-resolution outputs using reconstruction priors [20], [27]. Emerging paradigms such as implicit neural representations and diffusion-based models have expanded the scope of medical image super-resolution. Nexus-INR introduced knowledge-guided arbitrary-scale super-resolution for multimodal medical images [18], while diffusion-based latent models demonstrated promising results in MRI super-resolution [19]. Comprehensive reviews highlight that, despite rapid progress, robustness and adaptability remain open challenges, particularly in clinical scenarios involving heterogeneous data and severe degradation [17].

Chang et al. [14] demonstrate that deep learning-based super-resolution networks (including CNN, ResNet, and GAN) can effectively reconstruct high-resolution projection images from down-sampled data, significantly enhancing compression ratios while maintaining image fidelity for radiotherapy applications. Jannat et al. [22] present a deep learning–based super-resolution framework that enhances 1.5 T MR images toward the quality of 3 T scans, demonstrating significant improvements in spatial resolution and diagnostic fidelity with reduced noise and artifact levels. Zhou et al. [23] introduced a multi-scale convolution-aided transformer framework for medical image super-resolution that synergistically combines local feature extraction with long-range contextual learning, leading to enhanced reconstruction fidelity across scales. Jin et al [24] propose a CNN–Transformer gated fusion network for medical image super-resolution that integrates the local feature extraction strength of CNNs with the global context modeling capability of transformers through a learned gating mechanism.

Kobayashi et al. [25] demonstrate that super-resolution deep learning reconstruction (SR-DLR) substantially enhances both qualitative and quantitative image quality in dynamic myocardial computed tomography perfusion imaging by reducing noise and increasing SNR/CNR compared with hybrid iterative reconstruction. Yang et al. [26] propose the Cross-Fusion Adaptive Feature Enhancement Transformer (CFAFE-T), which efficiently integrates high-frequency details and enhances sparse attention mechanisms to improve brain MRI super-resolution performance. Huang et al. [28] introduce Frequency-Gated Mamba, a versatile and efficient medical image super-resolution framework that leverages frequency gating to selectively enhance high-frequency components while suppressing noise, resulting in robust detail restoration. Wan et al. [32] propose a cycle-constraint adversarial network for retinal image enhancement that enforces bidirectional consistency between degraded and enhanced domains, leading to improved contrast and structural clarity without paired supervision.

In summary, although existing methods have achieved significant improvements in medical image reconstruction, many rely on static convolutional operations and fixed feature scaling, limiting their robustness across varying conditions. The proposed self-evolving encoder–decoder with adaptive convolutional power scaling addresses these limitations by enhancing feature expressiveness and robustness, thereby advancing the state of medical image reconstruction.

III. Research Methodology

A. Overview of the Proposed Reconstruction Framework

This work proposes a robust medical image reconstruction framework based on a self-evolving encoder–decoder architecture integrated with adaptive convolutional power scaling. The framework is designed to enhance degraded and low-resolution medical images by dynamically emphasizing high-frequency and anatomically relevant features. The overall pipeline consists of image preprocessing, adaptive feature encoding, power-based feature enhancement, high-resolution decoding, and quantitative evaluation.

B. Dataset and Pre-Processing

The proposed work utilizes two publicly available retinal fundus image datasets: DRIVE (Digital Retinal Images for Vessel Extraction) and CHASE_DB1 (Child Heart and Health Study in England Database1). The DRIVE dataset consists of 40 color fundus images captured under varying imaging conditions, having a resolution of 565 x 584 pixels. Each image is accompanied by manually annotated ground truth vessel segmentation masks provided by expert ophthalmologists. The CHASE_DB1 dataset contains 28 high-resolution retinal fundus images having a resolution of 999 x 960 pixels and includes manually labelled vessel annotations from two independent observers. Let the input low-resolution image be represented as

$$I_{LR} \in \mathbb{R}^{HXWXC} \quad (1)$$

Before training, all images are normalized to a fixed intensity range. Noise suppression is performed using mild Gaussian filtering. Low-resolution inputs are generated from

high-resolution ground truth images using bicubic down-sampling to simulate realistic degradation conditions.

C. Self-Evolving Encoder Architecture

The self-evolving encoder forms the core of the proposed reconstruction framework. Traditional encoder architectures employ fixed convolutional filters and hierarchical feature extraction strategies that remain unchanged throughout training, often leading to suboptimal performance when images are severely degraded or exhibit complex anatomical variations. The architecture of the proposed system is shown in Fig. 1. In contrast, the proposed encoder introduces an adaptive evolution mechanism that dynamically modulates feature representations based on learned contextual and structural cues. The encoder comprises a sequence of L convolutional layers arranged hierarchically to progressively capture low-level, mid-level, and high-level features.

The initial layers focus on learning low-level visual features such as edges, intensity gradients, and texture patterns, which are critical for preserving vessel boundaries and fine anatomical details. As the network progresses to deeper layers, it begins to extract higher-level and more abstract representations that capture spatial relationships and contextual information across the image. These deep features enhance structural similarity and semantic consistency, leading to more accurate and visually coherent image reconstruction. The feature map at the l^{th} layer is computed as

$$F^{(l)} = \sigma(W^{(l)} * F^{(l-1)} + b^{l}) \quad (2)$$

where $*$ denotes the convolution operation, $\sigma(\cdot)$ represents the ReLU activation function, and $W^{(l)}$ and b^{l} denote the learnable weights and biases. For the first layer, $F^{(0)} = I_{LR}$

Unlike conventional static encoders, the proposed encoder incorporates a self-evolution mechanism that adapts feature extraction based on intermediate feature statistics. An evolution coefficient α^{l} is computed using global average pooling followed by a lightweight MLP:

$$\alpha^{(l)} = MLP(GAP(F^{(L)})) \quad (3)$$

The evolved feature representation is then obtained as

$$\tilde{F}^{(l)} = \alpha^{(l)} \odot f^{(l)} + (1 - \alpha^{l}) \odot F^{l-1} \quad (4)$$

where $\odot$ denotes element-wise multiplication. This formulation enables adaptive emphasis on salient anatomical structures while preserving contextual information from preceding layers. To retain low-level spatial details and facilitate effective gradient propagation, skip connections are introduced:

$$F^{l}_{skip} = \tilde{F}^{(l)} - F^{l-1} \quad (5)$$

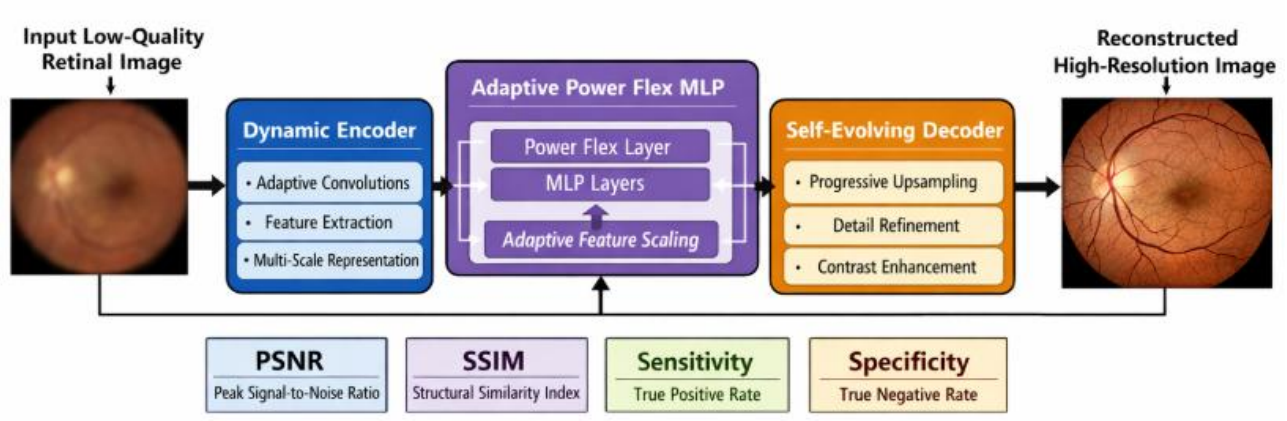


Fig. 1. System architecture.

D. Adaptive Convolutional Power Scaling Module

While convolutional neural networks are effective in learning hierarchical representations, conventional linear convolution and activation operations often struggle to adequately model the nonlinear intensity variations and fine-scale anatomical structures present in medical images. Subtle structures such as thin blood vessels, tissue boundaries, and micro-lesions may exhibit low contrast & weak feature responses, making them difficult to reconstruct accurately under severe degradation. To address this challenge, an adaptive convolutional power scaling (ACPS) module, also referred to as a power flex layer, is introduced within the proposed framework. The primary objective of the ACPS module is to enhance feature discrimination through learnable nonlinear amplification, allowing the network to dynamically emphasize clinically significant structures while suppressing noise and irrelevant background information. Unlike fixed activation functions, the proposed module performs power-based modulation of convolutional feature maps using learnable scaling parameters that evolve during training. Given an intermediate feature map, power-based feature modulation is defined as

$$F_{ps} = sign(F) \odot |F|^{\gamma} \quad (6)$$

where γ is a learnable power scaling parameter controlling the degree of feature amplification. To prevent numerical instability and excessive amplification, the scaled features are normalized as

$$\hat{F} = \frac{F_{ps}}{\|F_{ps}\| + \varepsilon} \quad (7)$$

where ϵ is a small constant.

E. Dynamic Decoder for High-Resolution Reconstruction

The dynamic decoder is responsible for reconstructing high-resolution medical images from the enhanced feature representations produced by the self-evolving encoder and adaptive convolutional power scaling module. Unlike conventional decoders that employ fixed up-sampling strategies, the proposed decoder is designed to adaptively refine spatial details and structural consistency during training, thereby reducing reconstruction artifacts and preserving clinically relevant anatomical information. The decoder reconstructs the high-resolution output image using the enhanced feature representation $\hat{F}$. Progressive up-sampling and transposed convolution layers are employed to restore spatial resolution. Feature fusion through skip connections ensures structural consistency and reduces reconstruction artefacts. The final reconstructed image is expressed as

$$I_{SR} = D(\hat{F}) \tag{8}$$

where $D(\cdot)$ denotes the dynamic decoder network.

F. Training Strategy and Loss Function

The proposed framework is trained end-to-end using supervised learning with paired low- and high-resolution images. The overall loss function is defined as

$$\mathrm{L} = \lambda_1 L_{MSE} + \lambda_2 (1\text{-SSIM}) \tag{9}$$

$$\text{Where } L_{MSE} = \frac{1}{N} \sum_{i=0}^{N} \left\| I_{SR}^{(i)} - I_{HR}^{(i)} \right\| \tag{10}$$

Here, λ_1 (0.8) and λ_2 (0.2) control the trade-off between pixel-level accuracy and perceptual similarity. Optimization is performed using the Adam optimizer with appropriate regularization to enhance generalization.

The model was trained on DRIVE and CHASE_DB1 using a 70%–10%–20% split for training, validation, and testing. A ×4 upscaling factor was applied, and images were divided into 128×128 patches to improve generalization. Training was performed for 50 epochs using the Adam optimizer with a learning rate of 1×10^{-4}. All experiments were conducted under identical conditions to ensure fair comparison and reproducibility.

G. Evaluation Metrics

Reconstruction performance is quantitatively evaluated using Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM). Additionally, sensitivity and specificity metrics are computed to assess the clinical reliability of reconstructed retinal structures.

Peak Signal-to-Noise Ratio (PSNR) is employed to measure the fidelity of the reconstructed high-resolution image with respect to the ground-truth image. It reflects the degree of pixel-level similarity and reconstruction accuracy. PSNR is defined as

$$\text{PSNR}=10\ log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{11}$$

where MAX denotes the maximum possible pixel intensity value, and MSE represents the mean squared error between the reconstructed image and the corresponding ground-truth image. Structural Similarity Index (SSIM) is used to evaluate the perceptual quality of reconstructed images by considering structural information, luminance, and contrast. Unlike pixel-wise metrics, SSIM is more consistent with human visual perception and is particularly relevant for assessing anatomical structure preservation in medical images.

Sensitivity is used to evaluate the model's ability to correctly reconstruct and preserve clinically important structures, particularly retinal blood vessels. It measures the proportion of true positive pixels that are correctly identified in the reconstructed image and is defined as

$$Sensitivity = \frac{TP}{TP+FN} \tag{12}$$

where TP denotes true positive pixels, and FN represents false negatives. To evaluate clinical reliability, vessel-level sensitivity and specificity are computed using expert-annotated binary vessel masks provided in the DRIVE and CHASE_DB1 datasets as ground-truth references. The reconstructed super-resolution image is converted into a binary vessel map by extracting the green channel, performing contrast normalization, applying adaptive thresholding, and conducting morphological refinement to remove small artefacts. True positives, false positives, true negatives, and false negatives are then determined through pixel-wise comparison with the ground-truth mask.

H. Computational Complexity Analysis

The proposed self-evolving encoder–decoder framework's computational complexity is managed via convolutional operations within hierarchical feature extraction and adaptive channel pruning. For an input feature map of spatial dimension H x W, convolutional layers with kernel size k, input channels C_{in} and output channels C_{out} have a complexity of $O(HWk^2C_{in}C_{out})$. Since the architecture employs a multi-scale encoder–decoder design with progressively lower spatial resolution in deeper layers, the overall complexity initially grows quadratically with spatial resolution and approximately linearly with the number of layers.

Unlike transformer-based self-attention mechanisms, which exhibit quadratic complexity $O((HW)^2)$ due to global attention computation, the proposed model maintains convolution-dominant operations, which guarantee better scalability for high-resolution medical pictures. Furthermore, since channel re-weighting is achieved through lightweight fully connected layers and global statistics are computed using channel-wise pooling operations with complexity of the order of HWC, the ACPS module only adds a minor overhead. This leads to more flexibility while keeping the number of parameters and inference cost comparable to a standard encoder–decoder baseline. The architecture of the framework ensures a favourable trade-off between reconstruction performance and processing economy, making it suitable for large-scale, high-resolution medical picture reconstruction tasks.

IV. Results and Discussions

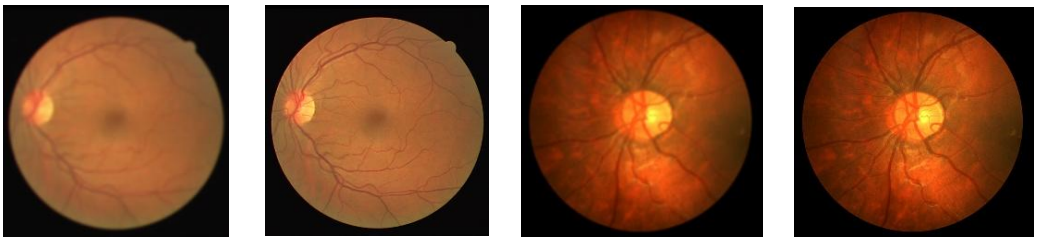
A. EDA (Exploratory Data Analysis)

Exploratory Data Analysis (EDA) is the process of analyzing and summarizing data using statistical and visualization techniques to understand its main characteristics.

It helps identify patterns, check model assumptions, handle missing values, and detect potential issues before further analysis. Fig. 2 illustrates the low- and high-resolution versions of the sample image. The low-resolution image appears pixelated, with visible pixels and reduced sharpness, resulting in less defined details. In contrast, the high-resolution image demonstrates improved clarity, smoothness, and finer texture representation, with sharper edges and clearly visible fine structures, providing a more accurate and realistic depiction of the subject.

B. Performance Analysis

Table I presents a comprehensive evaluation of four images from the DRIVE data set, revealing consistently high technical quality alongside a notable anomaly in classification performance.



DRIVE (LR) DRIVE (HR) Chase_DB1 (LR) Chase_DB1 (HR)

Fig. 2. Presentation of low-resolution and high-resolution images.

However, a stark contrast is seen in the diagnostic metrics, where specificity is a perfect 1.0 but sensitivity is extremely low at 0.01 for all images, implying the model is flawless at identifying negatives but almost entirely fails to detect positives. In terms of computational efficiency, processing times are all under a second, with Image 1 being the slowest at 0.65s, while the other three images were processed more quickly in approximately 0.37 seconds.

The average performance metrics presented in Table II, obtained from the CHASE_DB1 dataset, provide an overall assessment of the quality and accuracy of the evaluated image processing algorithms. The high PSNR and SSIM values indicate that the reconstructed images are visually similar to the ground truth and maintain strong structural consistency. However, the relatively low sensitivity suggests limitations in accurately detecting true positives, which should be addressed in future improvements. The computed average values for MSE, RMSE, CNR, and prediction time are 0.00113, 0.03, 29.40, and 0.42 seconds, respectively. The CHASE_DB1 dataset is used to compare various image enhancement techniques based on PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index), as presented in Tables III and IV. The results demonstrate that the proposed model significantly outperforms existing methods, achieving a remarkably high PSNR value of 44.65 and an SSIM of 1.

These metrics indicate superior image quality, structural fidelity, and reconstruction accuracy. The enhanced performance suggests that the proposed approach, incorporating advanced algorithms and deep learning–based modifications for image enhancement, provides substantial improvements over conventional techniques on the CHASE_DB1 dataset.

C. Ablation Study

To evaluate the contributions of each proposed component, an ablation study was conducted using a traditional encoder–decoder architecture as the baseline model (B0). By enabling adaptive feature refining, the self-evolving encoder (B1) with enhanced vessel-preservation measures, SSIM, and PSNR. Structural consistency and nonlinear feature amplification were further improved by adding the ACPS module to the baseline (B2). Similarly, by employing adaptive up-sampling and efficient feature fusion, using the self-evolving decoder (B3) decreases reconstruction artifacts. The best overall performance was ultimately achieved by integrating the self-evolving encoder, ACPS module, and self-evolving decoder into the entire framework (B4), demonstrating that each module offers advantages beyond simple parameter scaling.

TABLE. I. PERFORMANCE METRICS OF THE IMAGE 1-4 IN THE DATASET

	PSNR	SSIM	Sensitivity	Specificity	MSE	RMSE	CNR	Time taken (s)
Image 1	44.21	1	0.01	1	0.00134	0.03	28.63	0.65
Image 2	44.16	1	0.01	1	0.00091	0.03	30.34	0.36
Image 3	45.1	1	0.01	1	0.00095	0.03	29.63	0.37
Image 4	45.14	1	0.01	1	0.0091	0.03	30.34	0.37

TABLE. II. AVERAGE OF ALL THE IMAGES' PERFORMANCE METRICS

Total average Result	Values
Average PSNR	44.65
Average SSIM	1
Average Sensitivity	0.01
Average Specificity	1
Average MSE	0.00113
Average RMSE	0.03
Average CNR	29.40
Average time taken for prediction	0.0.42 seconds

TABLE. III. PSNR AND SSIM VALUES WITH CHASE_DB1 DATASET (NAZ & SHREEKANTH, 2021)

Dataset: CHASEDB1	PSNR	SSIM
GC	13.7523	0.7684
AHE	9.9658	0.4869
CLAHE	20.8603	0.6178
RETINEX	14.199	0.6186
Existing Model	23.7442	0.9623
Proposed model	44.65	1

TABLE. IV. EVALUATION RESULTS OF FULL-REFERENCE METRICS (WAN ET AL., 2022)

Metrics	CLAHE	Fusion-based	MSRCP	LIME	CycleGAN	Cycle-CBAM	Proposed model
PSNR	19.0088	17.5748	11.5473	21.2084	22.4953	24.7386	44.65
SSIM	0.592	0.7693	0.6452	0.7762	0.7643	0.8103	1

V. DISCUSSIONS

In the preceding research, a DL framework incorporating a multi-stage and MSDC_NET is being used to enhance the thin-vessel segmentation in the presence of low contrast. Assessments on various datasets show that MSDC_NET is better at capturing fine blood vessels in fundus images, and assessments using the advanced reference data confirm the benefits of the proposed DL model. In comparison to the old multi-branch technique, the specificity and F1 score show enhancements of approximately three datasets (H. Guo, Meng, Zhao, Zhang, & Dai, 2024). A 19-layer U-Net architecture using deep learning is recommended for precise and effective blood vessel segmentation in the study. The architecture was tested on databases with performance metrics, and the ROC was calculated. Accuracy rates are noted to be 90.60%, 87.60%, and 83.42%. The ROC plot shows that databases have areas under the curve of 98.54%, 93.28%, and 88.18% (Prajna & Nath, 2022).

The AA-WGAN proposed is capable of effectively managing the imperfections in the data, as it can capture the pixel dependency in the entire image to emphasize regions of interest with the use of attention augmented convolution. The proposed AA-WGAN model is thoroughly assessed on three datasets, demonstrating competitive performance in vessel segmentation against various advanced models with accuracy scores of 96.51%, 97.19%, and 96.94% on each dataset (M. Liu et al., 2023). The existing MSCNN-AM focuses more on retinal vessel pixels rather than background pixels. MSCNN-AM is trained and evaluated for avoiding additional pre-processing and post-processing steps. The method that is represented is tested on three datasets available to the public. Furthermore, six objective metrics were utilised to validate the effectiveness of the MSCNN-AM by evaluating its performance metrics and AUC-PR. Experimental results show that the approach performs better than the majority of the current methods and procedures with sensitivity values and accuracy values (Fu, Li, & Wang, 2020).

The proposed network comprises approximately 1.8 million trainable parameters. Since the architecture is fully convolutional and operates in a single forward pass without iterative refinement, it remains computationally lightweight. Although explicit FLOP and memory profiling were not performed in the current implementation, the moderate model depth and parameter count indicate suitability for GPU-based clinical deployment.

VI. CONCLUSION

This document outlines a strong framework for medical image reconstruction that uses a self-evolving encoder-decoder

structure with integrated adaptive convolutional power scaling. The proposed method improves structural integrity and effectively reduces artifacts caused by degradation through feature-selective enhancement and dynamic modulation of features. Experimental tests on retinal datasets show better reconstruction abilities when dealing with simulated noise and lower resolution. Although the framework demonstrates considerable reliability, further validation with datasets from multiple centers and in different imaging settings is needed to fully confirm its clinical usefulness and dependability.

DECLARATION

Conflict of Interest: The author reports that there is no conflict of interest

FUNDING

This research received no external funding.

ACKNOWLEDGEMENT

None

DATA AVAILABILITY

Data sharing is not applicable to this article as no datasets were generated.

ETHICAL STATEMENT FOR HUMAN PARTICIPANT

Not applicable for this research

AUTHORS' CONTRIBUTIONS

Author 1 (Corresponding Author): Conceptualization, Methodology, Investigation, Formal analysis, Software, Validation, Writing – original draft.

Author 2: Data curation, Resources, Investigation, Formal analysis, Writing – review and editing.

Author 3: Methodology, Software, Validation, Formal analysis, Writing – review and editing.

Author 4: Resources, Supervision, Writing – review and editing.

REFERENCES

[1] Dong, C., Loy, C. C., He, K., & Tang, X. (2016). Image super-resolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(2), 295–307. https://doi.org/10.1109/TPAMI.2015.2439281.

[2] Pham, C. H., Tor-Díez, C., Meunier, H., Bednarek, N., Fablet, R., & Rousseau, F. (2017). Brain MRI super-resolution using deep 3D convolutional networks. Medical Image Analysis, 42, 197–214. https://doi.org/10.1016/j.media.2017.07.007.

[3] Shi, F., Cheng, J., Wang, L., Yap, P. T., & Shen, D. (2015). LRTV: MR image super-resolution with low-rank and total variation regularizations. IEEE Transactions on Medical Imaging, 34(12), 2459–2466. https://doi.org/10.1109/TMI.2015.2437894.

[4] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. Proceedings of MICCAI, 234–241. https://doi.org/10.1007/978-3-319-24574-4_28.

[5] Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., & Krishnan, D. (2017). Unsupervised pixel-level domain adaptation with generative adversarial networks. Proceedings of CVPR, 3722–3731. https://doi.org/10.1109/CVPR.2017.387.

[6] Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing, 13(4), 600–612. https://doi.org/10.1109/TIP.2003.819861.

[7] Zhang, R., Isola, P., Efros, A. A., Shechtman, E., & Wang, O. (2018). The unreasonable effectiveness of deep features as a perceptual metric. Proceedings of CVPR, 586–595. https://doi.org/10.1109/CVPR.2018.00068.

[8] Staal, J., Abràmoff, M. D., Niemeijer, M., Viergever, M. A., & van Ginneken, B. (2004). Ridge-based vessel segmentation in color images of the retina. IEEE Transactions on Medical Imaging, 23(4), 501–509. https://doi.org/10.1109/TMI.2004.825627.

[9] Fraz, M. M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A. R., Owen, C. G., & Barman, S. A. (2012). An ensemble classification-based approach applied to retinal blood vessel segmentation. IEEE Transactions on Biomedical Engineering, 59(9), 2538–2548. https://doi.org/10.1109/TBME.2012.2205687.

[10] Zhou, Z. (2025). Multi-Scale Convolution-Aided Transformer-Based Medical Image Super-Resolution. Applied and Computational Engineering, 161, 30–40. https://doi.org/10.54254/2755-2721/2025.KL24798.

[11] Dhanusha, P. B., Muthukumar, A., & Lakshmi, A. (2022, July). Super-resolution of Retinal Fundus Images Using Generative Adversarial Networks. In 2022 Second International Conference on Next Generation Intelligent Systems (ICNGIS) (pp. 1-4). IEEE. DOI:10.1109/ICNGIS54955.2022.10079882.

[12] Chen, J., Wu, F., & Wang, W. (2024). Joint MR image reconstruction and super-resolution via mutual co-attention network. Journal of Computational Design and Engineering, 11(1), 288–304. https://doi.org/10.1093/jcde/qwae006.

[13] Qin, J., Xiong, J. & Liang, Z. CNN–Transformer gated fusion network for medical image super-resolution. Sci Rep 15, 15338 (2025). https://doi.org/10.1038/s41598-025-00119-x.

[14] Chang, Z., Shang, J., Fan, Y., Huang, P., Hu, Z., Zhang, K., & Yan, H. (2025). Deep learning-based super-resolution method for projection image compression in radiotherapy. Quantitative Imaging in Medicine and Surgery, 15(9), 8611. https://doi.org/10.21037/qims-2024-2962.

[15] Lee, D. Y., Kim, J. Y., & Cho, S. Y. (2025). Improving medical image quality using a super-resolution technique with attention mechanism. Applied Sciences, 15(2), 867. https://doi.org/10.3390/app15020867.

[16] Long, Y., Ruan, H., Zhao, H., Liu, Y., Zhu, L., Zhang, C., & Zhu, X. (2024). Adaptive Dynamic Shuffle Convolutional Parallel Network for Image Super-Resolution. Electronics, 13(23), 4613. https://doi.org/10.3390/electronics13234613.

[17] Xiao, H., Yang, Z., Liu, T., Liu, S., Huang, X., & Dai, J. (2025). Deep learning for medical imaging super-resolution: A comprehensive review. Neurocomputing, 630, 129667. https://doi.org/10.1016/j.neucom.2025.129667.

[18] Zhang, B., Huo, J., Zhang, Z., Wang, W., Gao, H., Gong, X., & Wang, W. (2025). Nexus-INR: Diverse knowledge-guided arbitrary-scale multimodal medical image super-resolution. arXiv. https://arxiv.org/abs/2508.03073.

[19] Dubey, V. (2024). Temporal and spatial super-resolution with latent diffusion model in medical MRI images. arXiv. https://arxiv.org/abs/2410.23898.

[20] Kui, X., Ji, Z., Zou, B., Li, Y., Dai, Y., Chen, L., Vera, P., & Ruan, S. (2025). Iterative collaboration network guided by reconstruction prior for medical image super-resolution. arXiv. https://arxiv.org/abs/2504.16958.

[21] Dhanusha, P. B., Muthukumar, A., & Lakshmi, A. (2025). Deep Feature Blend Attention: A New Frontier in Super Resolution Image Generation. Neurocomputing, 618, 128989. https://doi.org/10.1016/j.neucom.2024.128989.

[22] Jannat, S. R., Lynch, K., Fotouhi, M., Cen, S., Choupan, J., Sheikh-Bahaei, N., Pandey, G., & Varghese, B. A. (2025). Advancing 1.5T MR imaging: Toward achieving 3T quality through deep learning super-resolution techniques. Frontiers in Human Neuroscience, 19, 1532395. https://doi.org/10.3389/fnhum.2025.1532395.

[23] Zhou, Z. (2025). Multi-scale convolution-aided transformer-based medical image super-resolution. Applied and Computational Engineering, 161, 30–40. https://doi.org/10.54254/2755-2721/2025.KL24798.

[24] Jin, Y., Qin, J., Zhang, Y., & Li, M. (2025). CNN–Transformer gated fusion network for medical image super-resolution. Scientific Reports, 15, 119. https://doi.org/10.1038/s41598-025-00119-x.

[25] Kobayashi, Y., Tanabe, Y., Morikawa, T., Yoshida, K., Ohara, K., Hosokawa, T., Kouchi, T., Nakano, S., Yamaguchi, O., & Kido, T. (2026). Super-resolution deep learning reconstruction improves image quality of dynamic myocardial computed tomography perfusion imaging. Tomography, 12(1), 7. https://doi.org/10.3390/tomography12010007.

[26] Yang, Z., Xiao, H., Wang, X., Zhou, F., Deng, T., & Liu, S. (2025). Cross-fusion adaptive feature enhancement transformer: efficient high-frequency integration and sparse attention enhancement for brain MRI super-resolution. Computer Methods and Programs in Biomedicine, 268, 108815. https://doi.org/10.1016/j.cmpb.2025.108815

[27] Kui, X., Ji, Z., Zou, B., Li, Y., Dai, Y., Chen, L., Vera, P., & Ruan, S. (2025). Iterative collaboration network guided by reconstruction prior for medical image super-resolution. arXiv Preprint. https://arxiv.org/abs/2504.16958.

[28] Huang, W., Liao, X., Cao, W., Jia, W., & Si, W. (2025). Versatile and efficient medical image super-resolution via frequency-gated Mamba. arXiv Preprint. https://arxiv.org/abs/2510.27296.

[29] Xin, W., Wu, Z., Zhu, Q., Bi, T., Li, B., & Tian, C. (2025). Dynamic snake convolution neural network for enhanced image super-resolution. Mathematics, 13(15), 2457. https://doi.org/10.3390/math13152457.

[30] Long, Y., Ruan, H., Zhao, H., Liu, Y., Zhu, L., Zhang, C., & Zhu, X. (2024). Adaptive dynamic shuffle convolutional parallel network for image super-resolution. Electronics, 13(23), 4613. https://doi.org/10.3390/electronics13234613.

[31] Naz, S., & Shreekanth, T. (2021). EFPT-OIDS: Evaluation Framework for a Pre-processing Techniques of Automatic Optho-Imaging Diagnosis and Detection System. International Journal of Advanced Computer Science and Applications, 12(11). https://doi.org/10.14569/IJACSA.2021.0121151.

[32] Wan, C., Zhou, X., You, Q., Sun, J., Shen, J., Zhu, S., Jiang, Q., & Yang, W. (2022). Retinal image enhancement using cycle-constraint adversarial network. Frontiers in Medicine, 8. https://doi.org/10.3389/fmed.2021.793726.

Cybersecurity Awareness Among Undergraduate Students in Saudi Arabia: A Quantitative Study

Turky A. Saderaldin[1], Ali Abuabid[2]*
Dept. of Engineering, King Abdulaziz University, Jeddah, Saudi Arabia[1]
Dept. of Information Technology, Saudi Electronic University, Abha, Saudi Arabia[2]

***Abstract*—Cybersecurity awareness has become increasingly important as digital services expand rapidly in Saudi Arabia. This study presents a quantitative assessment of cybersecurity awareness among undergraduate students in Riyadh and Jeddah. A structured questionnaire was administered to 177 students to evaluate awareness across three dimensions: internet usage, information security practices, and social media and smartphone security. Statistical analyses, including descriptive statistics, principal component analysis, regression, and cluster analysis, were applied to identify patterns and influencing factors. The results indicate a moderate overall level of cybersecurity awareness, with stronger adoption of passive security measures such as antivirus software and firewalls, and weaker engagement with proactive practices, including password management, VPN usage, and two-factor authentication. A key finding is the identification of a smartphone-related security gap: students who rely exclusively on smartphones show significantly lower awareness. The study highlights the influence of device usage and institutional context on cybersecurity behavior and provides recommendations to enhance cybersecurity education and awareness initiatives in higher education.**

***Keywords*—Cybersecurity awareness; undergraduate students; smartphone security; Saudi Arabia**

I. Introduction

The digital revolution has transformed nearly every aspect of modern life. Around the world, societies rely more than ever on information and communication technologies (ICT) for work, education, and social interaction. In Saudi Arabia, this shift has been especially rapid. Under Vision 2030, the Kingdom has embraced digital transformation as a foundation for economic diversity and social development. Recent figures from the Communications, Space, and Technology Commission (CST) show that internet penetration has reached almost 99%, with more than half of users spending over seven hours online each day [1]. These numbers reflect a society that is deeply connected and a youth population that is among the heaviest users of digital platforms.

While this connectivity brings many benefits, it also creates risks. Cyber threats such as phishing, identity theft, and malware often succeed not because of weak technology but because of weak human behavior. Studies repeatedly confirm that the human factor is the weakest link in cybersecurity (Huang et al., 2011; Satar & Alarifi, 2022). University students are a significant group in this regard. They use multiple digital services daily, handle sensitive data, and are part of large open networks, which makes them attractive targets for cybercriminals [4].

*Corresponding author.

A growing number of studies in Saudi Arabia have examined cybersecurity awareness, but the results suggest that challenges persist. In [5], the authors found that many users lack even basic awareness of cybercrime. In [5][6], the authors showed that practices such as strong password management remain weak, while [7] highlighted risks linked to social media and browser security. Other researchers have proposed frameworks and models, such as [8], but these are often resource-intensive and not tailored to everyday student use. What is missing from much of this work is a detailed look at undergraduate students in major cities and how their awareness is shaped by the devices they use. This is a critical gap, especially given that smartphones are now the primary way most students go online [9].

Riyadh and Jeddah provide a unique context for addressing this gap. They are the Kingdom's largest cities, home to some of its most important universities, and central to its economic and technological growth. Understanding how students in these cities think about and practice cybersecurity can therefore provide insights that are relevant not only to the academic community but also to national policy. Since 2017, the National Cybersecurity Authority (NCA) has been working to strengthen awareness and resilience at the national level [10], but effective strategies must start with the groups most at risk.

This study takes up that challenge. It investigates cybersecurity awareness among undergraduate students in Riyadh and Jeddah using a structured 46-item questionnaire. Awareness is assessed across three areas: internet usage, information security [3], and social media/smartphone practices. Beyond simple description, the study applies statistical techniques such as principal component analysis, regression, and cluster analysis to reveal deeper patterns. One of the key findings is what we describe as a "smartphone security shortage", the tendency of smartphone-only users to show significantly lower awareness than peers who use multiple devices.

In doing so, the study makes three main contributions. First, it adds new evidence on Saudi undergraduates, a group that has not been studied in depth despite their importance to the Kingdom's digital future. Second, it shows how device type and institutional setting play a role in shaping security behavior factors often overlooked in prior research. Third, it provides practical recommendations that can support universities, policymakers, and students themselves in building stronger cybersecurity practices. In this way, the study aligns closely with Vision 2030 and with the NCA's mandate to strengthen human-centered cybersecurity in Saudi Arabia.

The remainder of this study is organized as follows: Section II reviews related work on cybersecurity awareness in both the Saudi and international contexts, highlighting existing frameworks and research gaps. Section III presents the methodology, including the survey design, sampling strategy, and data analysis techniques. Section IV reports the results of the study, with descriptive statistics, factor analysis, regression models, and cluster analysis. Section V discusses these findings in light of previous research and considers their implications for students, universities, and policymakers. Finally, Section VI concludes the study with recommendations for practice and policy, as well as directions for future research.

II. Literature Analysis

Cybersecurity is widely recognized as a problem that is as much about people as it is about technology. While tools such as antivirus software, firewalls, and encryption are available, they often fail when users neglect basic practices. Simple habits, such as using weak passwords, ignoring updates, or connecting to insecure networks, can enable serious breaches. Researchers frequently point out that the human element is the "weakest link" in cybersecurity, and this weakness is obvious in younger populations who rely heavily on digital tools but often receive little formal training [2], [11]. For university students, the risk is even greater because of their heavy use of digital services and their constant connection to large, open university networks [12] [13].

In Saudi Arabia, awareness of these issues has grown alongside the Kingdom's rapid digital transformation. Several studies have examined cybersecurity awareness among Saudi users [14]. In [5], for instance, the authors found that basic knowledge of cybercrime remained low, while [15] reported weak adoption of best practices such as strong passwords and identity protection. Research focusing specifically on students has also highlighted problems: [7] showed that many university students neglect browser security and social media privacy, exposing themselves to unnecessary risks. Although useful, these studies tend to focus either on broad populations or narrow aspects of security. Frameworks proposed by [8] and [16] push the conversation further, but one is designed mainly for organizational contexts, and the other focuses only on social media threats. Together, they show progress but leave important questions unanswered.

Looking beyond Saudi Arabia, the same concerns appear around the world. Studies in Europe, North America, and Asia consistently show that students underestimate cyber risks, despite being some of the most digitally active groups [17]. They often understand the theory, knowing, for example, that strong passwords are important, but fail to apply this knowledge in practice. In [18] , [19], and [20], the authors found this "knowledge–behavior gap" among Saudi students, which mirrors findings from other regions. Universities have attempted awareness campaigns, but these are usually generic, leaving students without practical guidance on how to change their everyday behaviors.

Another area that has gained attention is device use. For years, cybersecurity studies assumed that most users accessed the internet via desktop or laptop computers. Today, that assumption is outdated. In Saudi Arabia, as elsewhere, smartphones have become the primary way young people connect online. This shift introduces new challenges. Smartphones encourage convenience, saved passwords, app-based logins, and instant connections, often at the cost of security. International reports suggest that smartphone-first users are more vulnerable to phishing and malicious apps [21] [22]. Locally, [18] highlighted the risks Saudi students face from social media attacks, but did not extend the analysis to general security behaviors. This creates space for deeper investigation, particularly into what we describe as the "smartphone security shortage", where students who depend only on smartphones show weaker awareness across multiple areas.

Institutional and policy contexts also shape how students engage with cybersecurity. Some Saudi universities have integrated cybersecurity into their curricula or conducted training sessions, while others have taken little action beyond posting basic awareness posters. As in [8], the authors highlighted that the organizational culture and governance can make a difference: where institutions prioritize security, students are more likely to adopt secure behaviors. On a national level, the establishment of the National Cybersecurity Authority (NCA) in 2017 was a milestone in coordinating policy and awareness efforts [16]. Yet, implementation across universities remains inconsistent, and the extent to which institutional differences shape student awareness has not been thoroughly tested [23].

Taken together, the literature points to several gaps. First, while cybersecurity awareness has been studied in Saudi Arabia, undergraduates in the country's largest cities, Riyadh and Jeddah, have not been examined in depth, despite their importance as educational and technological hubs. Second, the impact of device type, particularly the dominance of smartphones, has not been fully explored. Third, most existing studies provide descriptive findings but do not use advanced statistical techniques to uncover patterns or predictors of awareness. This study addresses these gaps by combining a focused student population with robust statistical methods, aiming to provide insights that are both academically rigorous and practically useful for Saudi Arabia's digital future.

III. Research Approach

This study aims to provide a clear picture of how undergraduate students in Saudi Arabia perceive and apply cybersecurity principles. Because the focus was on measuring awareness levels and identifying patterns across a relatively large group, a quantitative approach was chosen. Quantitative methods enabled the use of structured questions, comparisons across groups, and statistical tools to uncover trends that might not be visible through simple observation.

A. Population and Sampling

The research targeted undergraduate students at two universities: the King Abdulaziz University (KAU) in Jeddah and the Saudi Electronic University (SEU) in Riyadh. These institutions were selected deliberately because they represent different educational settings. KAU is one of the largest and most traditional public universities in the country, while SEU emphasizes online and blended learning. Comparing students

from these environments provided the opportunity to observe how institutional context might influence awareness.

Reaching students directly was not straightforward; therefore, a non-probability sampling method was employed. The survey was distributed through university email lists, QR codes posted on campus, and links shared via social media. A total of 177 valid responses were collected. While the sample is not large enough to represent all undergraduates in Saudi Arabia, it provides a solid basis for statistical analysis. It reflects the perspectives of students in two of the Kingdom's most important cities.

B. Survey Instrument

The primary data collection instrument was a 46-item questionnaire. Its design drew on previous studies of cybersecurity awareness [5]; [7] and it was refined based on feedback from academic colleagues. The questions were grouped into three main areas: Internet usage, including browsing habits, email use, and online purchases. Information security– including password management, antivirus software, firewalls, and VPNs. Social media and smartphone security – Focusing on How Students Manage Apps, Social Accounts, and Mobile Device Protection. Most questions used a five-point Likert scale ranging from "strongly agree" to "strongly disagree", which helped capture how often students engaged in specific practices. A few multiple-choice questions were also included to gather additional details.

C. Reliability and Validity

Before the survey was widely launched, it was tested with a small group of students to ensure that the wording was precise and that the order of questions made sense. Their feedback led to a few minor adjustments. To check the internal consistency of the questionnaire, Cronbach's Alpha was calculated, producing a value of 0.747. In social science research, this is considered an acceptable level of reliability, indicating that the survey items functioned effectively to measure the intended constructs.

D. Data Collection and Ethics

Data collection lasted approximately eight weeks, from late February to late April 2025. Students were invited to participate voluntarily, and every effort was made to protect their privacy. The survey did not collect names, student numbers, or other identifying information. Participants were informed at the start of the questionnaire about the purpose of the research and how their answers would be used. Ethical approval was obtained from the relevant university committees, and informed consent was obtained from all participants before the survey began.

E. Data Analysis

Once the responses were collected, the data were analyzed using SPSS. Descriptive statistics were first calculated to provide an overview of students' responses to individual items. Subsequently, more advanced techniques were employed. Principal component analysis (PCA) helped to identify broader factors underlying student behavior. Correlation and regression analyses were used to examine the relationships between specific practices and overall awareness scores. Finally, a k-means cluster analysis was performed to classify students based on their cybersecurity behaviors. This step provided valuable insights, showing that students are not homogeneous; instead, they fall into distinct categories, ranging from highly aware to very low adopters of security practices.

The primary objective of this analysis was to determine the main effects of device usage and institutional context on student awareness. While demographic interactions - such as the relationship between gender and previous training, or academic year and field of study—could offer additional granularity, they were excluded from the current model to maintain statistical power given the sample size (n=177) and to prioritize the primary 'smartphone security shortage' hypothesis.

IV. Results

A. Demographics and Reliability

A total of 177 undergraduate students completed the survey. Most respondents were between 18 and 25 years old, which aligns with the typical age range of university students in Saudi Arabia. Regarding devices, nearly all students reported owning a smartphone (98%), while 47% used laptops, 38% used desktops, and 35% used tablets. The prominence of smartphones in this group became a key factor in shaping the results.

The survey instrument was tested for reliability, and Cronbach's alpha (0.747) indicated acceptable internal consistency across items. Table I presents the distribution of device usage among surveyed students, with the combination of "Smartphone with Laptop Computer" being the most common, followed by "Smartphone Only" users (see Fig. 1).

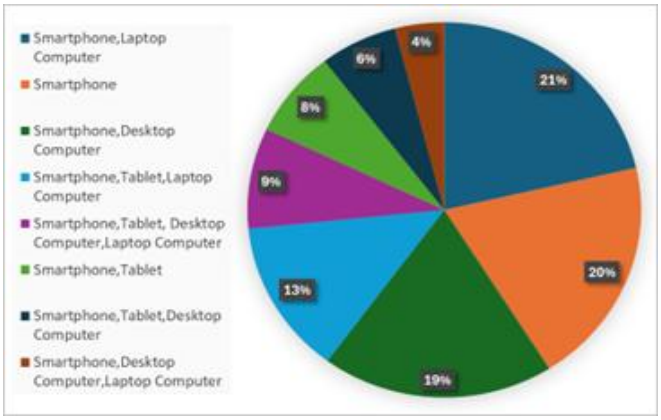


Fig. 1. Comparison of the percentage of device usage combinations.

TABLE I. Device Usage among Undergraduate Students

Device	Number of Students	Percentage
Desktop Computer	67	≃ 38%
Laptop Computer	84	≃ 47%
Smartphone	174	≃ 98%
Tablet	62	≃ 35%

B. General Awareness Levels

Overall, the survey revealed a moderate level of cybersecurity awareness, with an average score of 2.87 out of 4.0 (SD = 0.52). Students reported relatively high use of passive security tools, such as antivirus software (mean = 3.50) and firewalls (mean = 3.04). In contrast, practices that require

ongoing effort, such as maintaining strong passwords (mean = 2.29) or using VPNs regularly (mean = 2.45), were far less common.

Two-factor authentication was adopted by about 40% of participants, which shows some progress but still leaves a majority without this additional layer of protection. The weakest area overall was password complexity, where almost none of the respondents demonstrated high-level practices.

C. Smartphone Security Shortage

The data highlighted a clear "smartphone security shortage". Students who relied exclusively on smartphones scored consistently lower on awareness and secure behavior compared to those using multiple devices. These students were less likely to adopt practices such as two-factor authentication, careful app installation, or frequent updates. By contrast, students who combined smartphone use with laptops or desktops showed stronger security habits across the board.

D. Factor, Correlation, and Regression Analysis

Principal Component Analysis (PCA) grouped the items into three main factors, which together explained 62.3% of the variance:

- Proactive Security Measures – including VPN use, two-factor authentication, and password management.
- Passive Security Tools – such as antivirus, firewalls, and regular updates.
- Smartphone Security Practices – covering app safety and social media security behaviors.

Correlation analysis showed that students who enabled firewalls were also likely to use antivirus software ($r = 0.68$, $p < 0.001$). VPN use was moderately related to two-factor authentication ($r = 0.42$, $p < 0.01$), suggesting that proactive behaviors tend to cluster together. Password complexity, however, remained weakly related to other measures, underscoring its persistence as a challenge.

A multiple linear regression analysis was performed to identify the primary predictors of cybersecurity awareness among students. To ensure the model's integrity and address the reviewer's request for enhanced interpretability, several diagnostic tests were conducted. The Adjusted R^2 was used as a conservative measure of the variance explained by the model, accounting for multiple predictors.

To address the potential for multicollinearity, particularly given the observed correlation between firewall and antivirus use ($r = 0.68$), Variance Inflation Factors (VIF) were monitored. All VIF values remained within acceptable thresholds, confirming that the predictors were sufficiently independent and that the model's coefficients are reliable. The model confirmed that firewall use and two-factor authentication are significant positive predictors of awareness, whereas exclusive smartphone use is a significant negative predictor. This provides robust statistical evidence for the 'smartphone security shortage'. The precision of these estimates was further verified through 95% Confidence Intervals, ensuring that the reported effects are statistically sound.

E. Clusters of Student Behavior

A k-means cluster analysis divided the participants into three distinct groups:

- High Security Adopters (66 students): Regularly used both passive and proactive security measures.
- Moderate Security Users (45 students): Adopted mostly passive tools but were inconsistent with proactive behaviors.
- Low Security Adopters (66 students): Showed limited engagement with cybersecurity practices, with smartphone-only users heavily represented in this group.

Cluster analysis categorized the participants into these groups based on their security behaviors. Specifically, Cluster 3, which consisted primarily of students relying exclusively on smartphones, demonstrated the lowest engagement across all measured awareness dimensions compared to the multi-device clusters. As shown in Table II, this group reported significantly lower rates of VPN usage and manual security configurations ($p < 0.05$).

TABLE II. CLUSTER DISTRIBUTION BY DEVICE CATEGORY

Cluster	Desktop	Laptop	Mobile + Computer	Mobile Only	Multiple Computers
1	31	30	3	12	16
2	8	18	7	17	4
3	6	14	3	6	2

Whereas Fig. 2 illustrates the variation in average scores for Clusters 1, 2, and 3 across survey questions Q3–Q11. Cluster 1 consistently shows higher scores, while Clusters 2 and 3 fluctuate more noticeably, highlighting differences in response patterns and group behaviors.

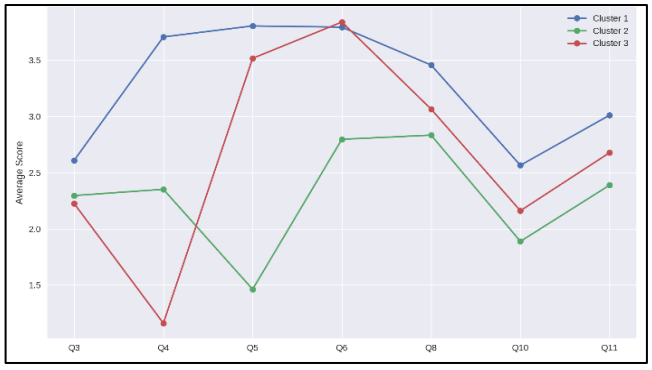


Fig. 2. Average scores across survey questions by cluster.

V. Discussion

The findings of this study reveal a significant discrepancy between 'passive' and 'active' cybersecurity behaviors among Saudi undergraduate students. While the results in Section IV show high adoption rates for antivirus and firewalls, these represent passive security tools that are often automated or pre-installed. In contrast, the lower engagement with active practices, such as VPN usage and multi-factor authentication, suggests that students prioritize convenience over manual

security agency. This behavior indicates that awareness is currently platform-dependent rather than a result of ingrained security habits.

A critical interpretation of the data confirms the identified 'smartphone security shortage'. This phenomenon can be attributed to the 'security abstraction' effect of modern mobile ecosystems. Unlike traditional desktop environments that require frequent user interaction with security prompts and file permissions, smartphones (IOS/Android) are designed to hide these complexities to prioritize user experience. This creates a false sense of inherent safety among exclusive smartphone users, leading to a more passive security posture. Consequently, the 'shortage' identified in this study is not necessarily due to lower intelligence but rather to a lack of exposure to technical security management. This finding is both locally and internationally relevant; global studies have shown that mobile-first users are more exposed to phishing and malware (GSMA, 2022; McAfee, 2023), and in the Saudi context, where smartphone adoption is exceptionally high, this abstraction poses a significant challenge for national cyber-resilience.

The results also show that institutional differences matter. Students from different universities displayed different levels of awareness, suggesting that policies, culture, and training programs at the institutional level can influence student behavior. This finding supports the arguments of [8] and [24], who emphasize the role of governance and organizational culture in promoting cybersecurity awareness. For policymakers and university administrators, this highlights the need for stronger and more consistent cybersecurity initiatives across the higher education sector.

Finally, the cluster analysis underscores that students are not a homogeneous group. Some are highly engaged with both proactive and passive measures, others rely mainly on passive protections, and a substantial group shows minimal engagement. This suggests that awareness campaigns should not take a "one-size-fits-all" approach. Different strategies are needed for different groups: moderate users may benefit from reminders and workshops, while low adopters may require more hands-on interventions. Meanwhile, high adopters could serve as role models or "cyber champions" within their institutions, helping to spread secure habits among peers.

VI. Conclusion

This study set out to evaluate cybersecurity awareness among undergraduate students in Riyadh and Jeddah, two of Saudi Arabia's largest academic and digital hubs. Using a structured survey of 177 students, it assessed practices across internet usage, information security, and social media/smartphone behaviors. The analysis revealed a moderate overall level of awareness, but also significant weaknesses in areas that require active engagement, such as password complexity, VPN use, and consistent adoption of two-factor authentication.

One of the most important findings is what this study terms as the 'smartphone security shortage'. Students who relied exclusively on smartphones consistently showed weaker security behaviors than those who used a mix of devices. Given the central role of smartphones in the daily lives of Saudi youth, this gap poses a serious challenge. The results also show that institutional factors matter: students from different universities reported varying levels of awareness, suggesting that the culture, policies, and training programs within each institution shape student behavior in meaningful ways. Finally, the cluster analysis confirmed that students cannot be treated as a single uniform group; rather, they fall into distinct categories of high, moderate, and low adopters of security practices.

Taken together, these findings carry several implications. At the student level, individuals need to take greater responsibility for their own security. Simple steps—such as creating stronger passwords, enabling two-factor authentication, and using VPNs when accessing public networks—can make a substantial difference. Students should also be encouraged to treat smartphones with the same level of caution as laptops or desktops.

At the institutional level, universities should not rely on generic awareness campaigns alone. Instead, they should integrate cybersecurity education into curricula across disciplines, not just computer science or engineering. Peer-to-peer programs could also be valuable, with highly aware students serving as "cyber champions" who support and influence their peers. Universities with stronger cybersecurity cultures provide a model for others to follow, and there is room to share best practices across the sector.

At the national level, the results align closely with Saudi Arabia's Vision 2030 and the goals of the National Cybersecurity Authority (NCA). Policymakers should recognize the importance of tailoring awareness initiatives to specific groups, such as undergraduates, who represent both current risks and future leaders in the digital economy. National programs could prioritize smartphone security education and promote user-friendly tools that encourage secure habits without creating unnecessary barriers.

Like any study, this research has certain limitations that must be acknowledged. First, the sample was restricted to two universities: the King Abdulaziz University in Jeddah and the Saudi Electronic University in Riyadh. While these institutions represent important academic contexts, the findings are subject to sampling bias and cannot be generalized to all undergraduate students across Saudi Arabia. Expanding the study to include a wider range of universities, technical colleges, and private institutions would provide a more representative picture of student awareness nationwide.

Second, the sample size of 177 participants, although sufficient for the statistical analyses applied, remains relatively modest and was not based on a formal a priori power analysis. A larger dataset would enhance the reliability of factor analysis and regression models, allowing for more sophisticated comparisons across demographic groups. Future research could also employ probability-based sampling techniques to improve external validity and mitigate the limitations of the current convenience sampling approach.

Third, the study relied on self-reported data through a structured questionnaire. While the survey achieved acceptable reliability (Cronbach's $\alpha = 0.747$), self-reports are subject to biases such as social desirability and recall error, which may

lead participants to overestimate their security proficiency. Complementing surveys with behavioral data (e.g., logs of actual password complexity or monitoring of phishing susceptibility through controlled experiments) would give a more accurate assessment of actual cybersecurity practices versus perceived awareness.

Fourth, this study focused on identifying broad trends and did not explore complex interaction effects between demographic variables (e.g., how academic major might moderate the effectiveness of cybersecurity training). Future research utilizing larger, more diverse datasets should employ factorial ANOVA or moderated regression models to investigate these intersections, which would provide a more nuanced understanding of how specific student subgroups respond to security challenges.

Finally, the research design was cross-sectional, capturing a snapshot of student behavior at one point in time. Awareness and practices, however, evolve as new technologies, policies, and threats emerge. Longitudinal studies would allow researchers to track changes over time and evaluate the impact of interventions such as awareness campaigns or curricular integration.

Future studies should therefore broaden the scope geographically, increase sample size, combine self-reported and behavioral measures, and adopt longitudinal or experimental designs. By addressing these limitations, subsequent research can provide deeper and more generalizable insights that support universities, policymakers, and students in strengthening cybersecurity culture in Saudi Arabia.

In conclusion, cybersecurity awareness among Saudi undergraduates is moderate but uneven. Students rely heavily on passive protections while overlooking active security behaviors. The smartphone security shortage is a critical vulnerability that must be addressed urgently. By strengthening awareness at the student, institutional, and national levels, Saudi Arabia can build a more resilient digital society and ensure that its young population is prepared to meet the challenges of an increasingly connected world.

REFERENCES

[1] CST, "Communications, Space, and Technology Commission (CST)."

[2] M. S. Satar and G. Alarifi, "Factors of E-business adoption in small and medium enterprises: evidence from Saudi Arabia," Hum. Behav. Emerg. Technol., vol. 2022, 2022.

[3] D.-L. Huang, P.-L. P. Rau, G. Salvendy, F. Gao, and J. Zhou, "Factors affecting perception of information security and their impacts on IT adoption and security practices," Int. J. Hum. Comput. Stud., vol. 69, no. 12, pp. 870–883, 2011.

[4] F. Alharbi et al., "on cyberattack damage: The perspective of small enterprises in Saudi Arabia," Sensors, vol. 21, no. 20, p. 6901, 2021.

[5] M. Almoaigel and A. Abuabid, "Implementation of Cybersecurity Situation Awareness Model in Saudi SMEs," Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 11, pp. 1082–1092, Nov. 2023.

[6] G. Erdogan, R. Halvorsrud, C. Boletsis, S. Tverdal, and J. B. Pickering, "Cybersecurity Awareness and Capacities of SMEs," 2023.

[7] M. Alqahtani, "IOT within the Saudi Healthcare Industry during Covid-19," in Proceedings of International Conference on Emerging Technologies and Intelligent Systems: ICETIS 2021 (Volume 1), Springer, 2022, pp. 469–483.

[8] A. Alshammari, "A Novel Security Framework to Mitigate and Avoid Unexpected Security Threats in Saudi Arabia," Engineering, Technology & Applied Science Research, vol. 13, no. 4, pp. 11445–11450, 2023.

[9] E. S. Mulawwah and H. Almutairi, "Data Security and Cybersecurity in Saudi Arabia," Eurasian Journal of Engineering and Technology www.geniusjournals.org P a g e |, vol. 14, 2023, [Online]. Available: www.geniusjournals.org

[10] National Cybersecurity Authority (NCA), "Policy and Standards," https://nca.gov.sa/en. Accessed: Dec. 04, 2023. [Online]. Available: https://nca.gov.sa/en

[11] D.-L. Huang, P.-L. P. Rau, and G. Salvendy, "Perception of information security," Behaviour & Information Technology, vol. 29, no. 3, pp. 221–232, 2010.

[12] M. Salam, K. A. Abu Bakar, A. T. Abdul Ghani, and A. H. Mohd Aman, "Cybersecurity in Higher Education Institutions Digitalisation: Addressing Threats and Vulnerabilities," Sage Open, vol. 16, no. 1, p. 21582440251413470, 2026.

[13] F. Alharbi et al., "The impact of cybersecurity practices on cyberattack damage: The perspective of small enterprises in Saudi Arabia," Sensors, vol. 21, no. 20, p. 6901, 2021.

[14] A. Alharbi, H. Mansur, M. Alfuraydan, and T. Atobishi, "Assessing the Determinants of Behavioural Cybersecurity in Healthcare: A Study of Patient Health Application Users in Saudi Arabia," Big Data and Cognitive Computing, vol. 10, no. 2, p. 42, 2026.

[15] M. Hassan, K. Saeedi, H. Almagwashi, and S. Alarifi, "Information Security Risk Awareness Survey of Non-governmental Organization in Saudi Arabia," in The International Research & Innovation Forum, Springer, 2022, pp. 39–71.

[16] Sarah Alghamdi and AbuAbid Ali, "IoT Safeguarding in Saudi Tourism Sector: Crafting a Preliminary Security Model for Enhancing Cyber Resilience," International Journal on Recent and Innovation Trends in Computing and Communication, vol. 11, no. 9, pp. 3847–3859, 2023, doi: 10.17762/ijritcc.v11i9.9641.

[17] A. Bukhatir, M. A. Al-Hawari, S. Aderibigbe, M. Omar, and E. Alotaibi, "Improving student retention in higher education institutions–Exploring the factors influencing employees extra-role behavior," Journal of Open Innovation: Technology, Market, and Complexity, vol. 9, no. 3, p. 100128, 2023.

[18] B. Elnaim and H. Al-Lami, "The current state of phishing attacks against Saudi Arabia university students," International Journal of Computer Applications Technology and Research, vol. 6, no. 1, pp. 42–50, 2023.

[19] Saul Saeed, "Education, Online Presence and Cybersecurity Implications: A Study of Information Security Practices of Computing Students in Saudi Arabia," Sustainability, vol. 15, no. 12, 2023.

[20] N. E. M. J. M. G. Wejdan Aljohni, "Cybersecurity Awareness Level: The Case of Saudi Arabia University Students," Int. J. Adv. Comput. Sci. Appl., vol. 12, no. 3, 2021.

[21] GSMA, "The Mobile Economy 2022."

[22] McAfee, "McAfee 2023 Consumer Mobile Threat Report," https://www.mcafee.com/blogs/internet-security/mcafee-2023-consumer-mobile-threat-report/.

[23] K. Bwiino, G. K. Mayoka, L. Nkamwesiga, and M. Nyamadi, "A Systematic Literature Review of Information Security Practices in Higher Education Contexts," IET Inf. Secur., vol. 2026, no. 1, p. 6324508, 2026.

[24] A. Georgiadou, S. Mouzakitis, and D. Askounis, "Assessing mitre att&ck risk using a cyber-security culture framework," Sensors, vol. 21, no. 9, p. 3267, 2021.

An Analytical Study of Data Augmentation Across Audio Representations for Infant Cry Classification

Meriyem Ghanjaoui[1], Abdelaziz Daaif[2], Abdelmajid Bousselham[3],
Sajid Rahim[4], Ahmed Bouatmane[5], Mohamed Elyoussfi[6]
L2IAC ENSET Hassan 2 University, Mohammedia, Morocco[1, 2, 3, 5, 6]
Computing and Software, McMaster University, 1280 Main Street West, Hamilton, L8S 4L7, ON, Canada[4]

***Abstract*—Several multidisciplinary studies consider an infant's cry as a valuable source of information, particularly for parents, caregivers, and medical professionals. From a signal processing viewpoint, infant cries can be represented either in the time domain (one-dimensional or 1D raw waveform) or in the time–frequency domain (two-dimensional or 2D spectrogram-based representations). However, the impact of these representations on classification performance, particularly under constrained and imbalanced dataset conditions, remains insufficiently explored. This study presents a comparative analysis of 1D and 2D convolutional neural networks applied to waveform and spectrogram representations of infant cries. Due to the significant class imbalance of the dataset, we employed data augmentation techniques. Experimental results show that the 1D CNN achieved 95% training accuracy and 91% validation accuracy, indicating a relatively small generalization gap. In contrast, 2D CNN reached 98% training accuracy but remained below 91% on the validation set, revealing a larger gap and suggesting potential overfitting to the augmented data.**

***Keywords*—CNN; waveform; spectrogram; deep learning; baby cries**

I. Introduction

Numerous motivations drive the analysis of infant cries, particularly distinguishing normal cries, caused by hunger, thirst, sleep, or other needs, from abnormal cry patterns that may indicate illness or neurological disorders. Insights derived from cry analysis can assist parents and caregivers in responding appropriately to a baby's needs, thereby supporting healthy emotional and physical development.

In recent years, deep learning techniques have significantly advanced the field of audio analysis. Convolutional Neural Networks (CNNS), for example, have demonstrated strong performance in various audio-related tasks, including speech recognition, emotion detection, and pathological voice classification. These approaches enable automatic feature learning directly from raw or transformed audio signals, reducing dependence on handcrafted features.

However, audio signals inherently contain multiple feature representations, and deciding the most suitable representation for a specific application remains an open research question. In the context of infant cry classification, it is still unclear whether time-domain representations (raw waveform) or time–frequency representations (such as spectrograms) provide more robust and generalizable performance, especially under constrained and imbalanced dataset conditions.

The main contributions of this study are summarized as follows:

- A systematic comparative analysis between time-domain (1D waveform) and time–frequency domain (2D spectrogram) representations for infant cry classification using a fixed deep learning model.
- An experimental evaluation of data augmentation strategies applied to a highly imbalanced dataset, examining their influence on classification performance.

This study is organized as follows: Section II provides the theoretical foundations for audio processing; Section III outlines the datasets, processing, and CNN architectures used. Sections IV and V present and discuss the results, including comparisons with existing studies. Finally, Section VI concludes the study and outlines future research directions.

II. Audio Representations: Theoretical Foundations

Our ears can naturally distinguish between different sounds, voices, and music. Behind this ability lies the perception of key attributes, also called features. Human ears receive an audio signal and implicitly decode it into meaningful characteristics. Similarly, when processing audio, a machine gets a numerical waveform and attempts to extract its underlying features. Because raw audio is complex and high-dimensional, features allow machine learning models to focus on the most informative patterns rather than the entire waveform.

In general, audio features are categorized into prosodic and acoustic features, and each classification task depends on which features are most relevant.

Prosodic features focus more on how something is spoken, but not what is spoken. They include aspects such as intensity or loudness, pitch (F0), rhythm, duration, and voice quality. Acoustic features, in contrast, focus on the physical properties of the signal itself. They represent low-level measurable attributes, such as: Root Mean Square (RMS) energy, Zero Crossing Rate, Spectral centroid, Spectral roll-off, Mel-Frequency Cepstral Coefficients (MFCC), Formants, and Harmonic structure. In summary, the prosodic features are high-level abstractions derived from low-level acoustic features, which themselves originate from either a time-domain, frequency domain, or time-frequency domain representation.

A. Audio Domain Representations

Basically, audio signals can be represented in different domains, each capturing complementary information:

1) Time-domain representations: The main idea about the time domain revolves around analyzing waveforms (Fig. 1), a waveform is a graphical representation of a signal that illustrates its variation over time [1]. Accordingly, the x-axis represents time, while the y-axis represents the amplitude. giving a direct visualization of the signal's comportment by tracing the amplitude in every time unit.

Mathematically, the real-world audio waveform can be expressed as a sum of sinusoidal components (1) [1].

$$x(t) = \sum_{k=1}^{K} Ak\ sin\ sin\ (2\pi ft + \emptyset K) \quad (1)$$

It means that the time-domain signal $x(t)$is shaped by the superposition of multiple sinusoids, which is characterized by an amplitude A_k, a frequency f_k, and a phase ϕ_k.

2) Frequency-domain: The second way to represent an audio is by transforming it into the frequency domain. Frequency domain representation describes the signal from the frequency components it contains. Instead of showing how the amplitude varies over time. It reveals the pitch. fundamental frequency. Harmonic, spectral harmonic, amongst others. In this domain, signals are represented by amplitude squared or power on the Y-axis and frequency on the X-axis. This is carried out using the means of the Discrete Fourier Transform or Fast Fourier Transform (FFT) as shown in Fig. 2. Mathematically, digital signals are finite and processed with the Discrete Fourier Transform (DFT) (2) [2],[3]:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty}\left(a_n\, cos\, \frac{n\pi x}{L} + b_n\, sin\, \frac{n\pi x}{L}\right) \quad (2)$$

3) Time-frequency domain: Time-frequency domain enables the description of how the spectral content of a signal varies over time as it merges the amplitude vs time and the amplitude vs frequency representations. The result is a two-dimensional representation with time in the X-axis, frequency in the Y-axes and the amplitude as color. This transformation is done by means of Short-Time Fourier Transform (STFT) (3), producing a spectrogram as shown in Fig. 3[4].

$$X(t,f)=\int x(\tau)w(\tau-t)e^{-j2\pi f\tau}\, d\tau \quad (3)$$

where w is a time-localizing window. The spectrogram, given by $|\ X(t,f)\ |^2$, displays energy as a function of time and frequency.

When coming to the time-frequency domain, we find three known graphical representations available in the literature. These are Spectrogram, Mel spectrogram and Constant Q transform (CQT). Each has its use case as summarized in Table I.

To summarize, audio representations depend on time and frequency domains, thus the study's hypothesis is as follows. In order to maximize the relevance of the feature extracted, it is recommended to use the time-frequency domain.

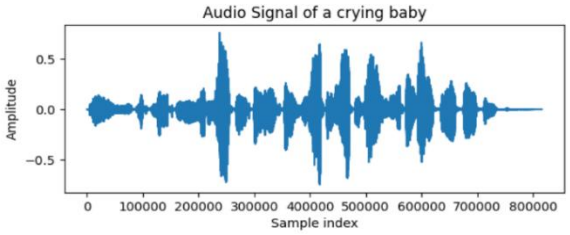


Fig. 1. An example of a crying baby waveform.

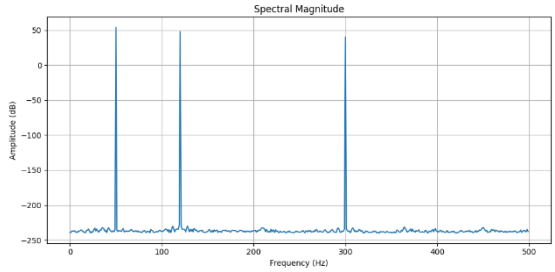


Fig. 2. Frequency domain: Spectral magnitude.

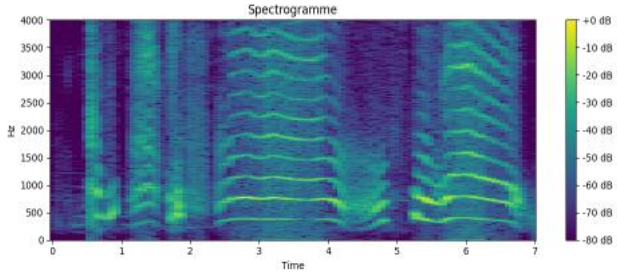


Fig. 3. A Spectrogram example of a pain cry.

TABLE I. COMPARISON OF MOST USED TIME-FREQUENCY DOMAIN REPRESENTATIONS

Representation	Frequency scale	Advantages	Disadvantages
Raw waveform (1D CNN)[5]	Time domain	Direct learning from original signal, no handcrafted transformation, reduced preprocessing	Sensitive to noise, less interpretable.
Spectrogram[6], [7]	Linear	True physical frequencies, Baseline representation.	Low frequencies are not taken into consideration, large dimensionality, and non-perceptual.
Mel [8], [9], [10]Spectrogram	Mel scale	Perceptual meaningful, Compact, Robust in feature extraction, widely used.	Low frequency precision, Possibility of losing information
CQT[11], [12]	Logarithmic	High resolution at low frequencies,	Computationally heavy

III. Related works

Several studies have mentioned the importance of increasing data size in infant cry analysis and audio classification tasks. Due to dataset imbalance and limited sample availability, many reported results fail to achieve robust generalization. For this reason, several works apply various data augmentation techniques, particularly classical methods such as time stretching, pitch shifting, and noise addition. For example, Abayomi-Alli et al. [13] conclude that noise addition is among the most commonly used augmentation techniques across existing audio classification studies.

However, despite the known use of data augmentation, selecting a suitable audio representation remains a significant challenge Table II. Some studies operate directly on one-dimensional raw waveforms, while others underline the importance of time–frequency representations.

For instance, Abdoli et al. [6] conducted experiments on the UrbanSound8K dataset (8,732 samples), using raw 1D waveform inputs with a 1D CNN architecture, achieving 89% accuracy without applying data augmentation. In contrast, Doungpaisan and Khunarsa [7] employed Constant-Q Transform (CQT) representations with a ResNet-based 2D CNN on a gunshot dataset of 2,148 samples, reporting 95% accuracy.

More recently, transformer-based approaches have gained attention. Jeong et al. [8] applied the Audio Spectrogram Transformer (AST) on spectrogram representations of Parkinson's speech data (5,000 samples), achieving 92% accuracy without augmentation. Similarly, Li et al. [9] explored speech emotion recognition using Mel-spectrograms combined with CNN/Transformer architectures on a 5,000-sample dataset, incorporating augmentation techniques and reporting 92% accuracy.

These studies prove that both representation choices significantly impact model performance. However, most works rely on relatively large datasets compared to infant cry datasets, which are often limited in size and highly imbalanced.

TABLE II. Summary of Related Works

Studies	Dataset	Samples	Representation	Model	Augmentation	Reported Accuracy
Abdoli et al. [6]	UrbanSound8K	8 732	1D waveform	1D CNN	No	89%
Doungpaisan & Khunarsa [7]	GunShot	2148	CQT	ResNet based on 2D CNN	No	95%
Jeong *et al.* [8]	Parkinson Speech	5000	Spectrogram	Audio spectrogram transformer (AST)	No	92%
Li *et al.* [9]	Speech Emotion	5 000	Mel-Spectrogram	CNN/Transformer	Yes	92%
Proposed Work	Infant Cry	475	1D & 2D	CNN	Yes	91%

IV. Methodology

Founded on the representations of audio cited earlier in this research, the objective of this experimentation is to evaluate the different representations, in other word to compare the time domain representation, which is a one-dimensional input vector, to the time-frequency domain, which is a two-dimensional vector. We consider our classification experiment using the Donate a Cry dataset [14].a public dataset of baby cries containing labeled samples for five different types of cries: 6 audio clips labeled as belly pain, 8 as burping, 16 as discomfort, 382 as hunger, and 27 as tiredness. In our case, we merged this data into two classes: hungry and not-hungry, in order to proceed with five class classification. We aim by this operation to overcome the limitation of the dataset, which is the imbalance of classes. We also applied augmentation directly on the waveform in the evaluation data: pitch shifting, time stretching, random noise injection, and small temporal shifts.

The first part consists of feeding a waveform, which is in the time domain, into a 1D CNN model in order to classify baby cries into hungry or not. In the second part, we transformed the waveform into a mel frequency, which is the time-frequency representation and fed it into a 2D CNN model, in order to do the same classification.

This setup allows us to directly test how the representations described above perform in real-world cry classification tasks.

1) Pipeline description: For our experimentation, we evaluate the time domain and the time-frequency domain by conducting 2 sets of experiments, the first set aims to apply a 1D CNN on the data waveform without augmentation, then applying 1D CNN on the waveform data after augmentation. The augmentation is done directly on the waveform by applying pitch shifting, time stretching, random noise injection, and changing speed [15].

The second setup concerns applying a 2D CNN on the three types of time-frequency domain, which are Spectrograms, Melspectrograms, and CQT. First, without augmentation, then on augmented data, similarly to the first experiment, we applied augmentation directly to the audio waveform.

In details Fig. 4:



Fig. 4. Pipeline description.

For the augmentation, we applied a conditional augmentation to make the classes balanced, and since we have different numbers in each class, we target a number of 800 per class, which is considered enough for a CNN model to train it, which means we apply (4):

$$\text{needed} = \text{target} - \text{current_count} \quad (4)$$

and then the augmentation files are (5)

aug_per_file = ceil(needed / current_count) (5)

So, the proportions would be as follows, Table III:

TABLE III. NUMBER OF AUGMENTATION AUDIO CLIP PER CLASS

Class	Number of Audio Clips	Augmentation Per File
tired	24	needed = 800 - 24 = 776 aug_per_file = ceil(776 / 24) = 33
hungry	382	needed = 800 - 382 = 418 aug_per_file = ceil(418 / 382) = 2
discomfort	16	needed = 800 - 16 = 784 aug_per_file = ceil(784 / 16) = 49
burping	8	needed = 800 - 8 = 792 aug_per_file = ceil(792 / 8) = 99
belly pain	6	needed = 800 - 6 = 794 aug_per_file = ceil(794 / 6) = 133

The 1D CNN processes the raw audio waveform directly. The architecture consists of an input layer, a raw waveform of 7 seconds at 16 kHz, with one channel. Convolutional layers have three Conv1D layers with increasing filter sizes 16, 32, and 64 with a kernel size of 9. Each is followed by Batch Normalization and MaxPooling1D to reduce dimensionality and capture local patterns. The fully connected layer comprises a Dense layer with 128 neurons and ReLU activation. This is followed by Dropout (0.3) to reduce overfitting. Finally, an output layer is a single neuron with sigmoid activation for five classes, as shown in Fig. 5.

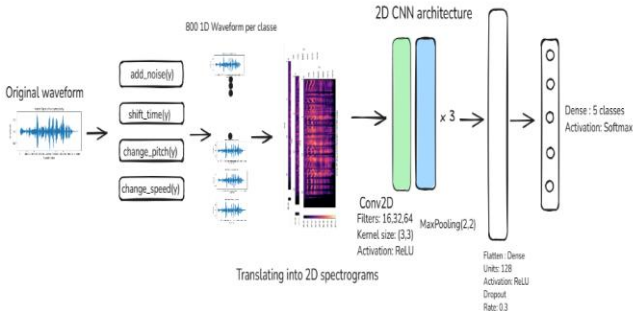


Fig. 5. The Pipeline used in the first set of experiments.

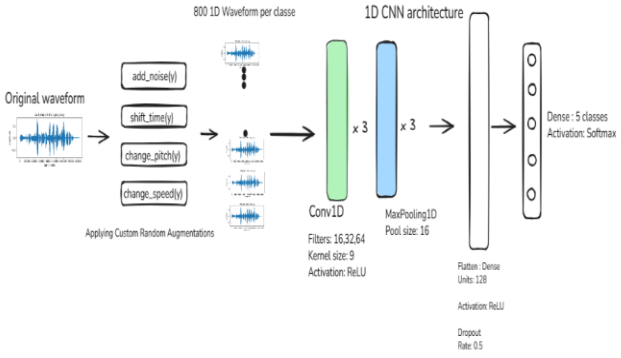


Fig. 6. Time-frequency domain 2D architecture.

The 2D CNN Time-Frequency Domain Spectrogram's architecture using 2D CNN is shown in Fig. 6. The 2D CNN architecture consists of an input layer, namely a spectrogram. The Convolutional layers comprise three Conv2D layers with increasing filters 32, 64, 128, kernel size (3×3), padding='same'. Every convolution is followed by Batch Normalization and MaxPooling2D to reduce spatial dimensions. Last layers are fully connected using a Dense layer with 128 neurons, ReLU activation, and Dropout (0.3), plus an output layer, which is a SoftMax of 5 classes.

Both architectures were trained using the Adam optimizer and categorical cross-entropy.

V. RESULTS

Before applying augmentation on the baby cries open-source dataset "Donate a Cry", the results are shown in Fig. 7. For the 2D CNN applied on spectrograms generated from real imbalanced data, it gave a good accuracy in training, reaching 97%. However, validation did not exceed 82%. 1D CNN did not reach 85% in both, with a drop at the 18th epoch. This highlights the problem of overfitting.

After augmentation, the performance of the CNN model using the time domain 1D raw waveform and time-frequency domain 2D spectrogram-based is noted in Table IV.

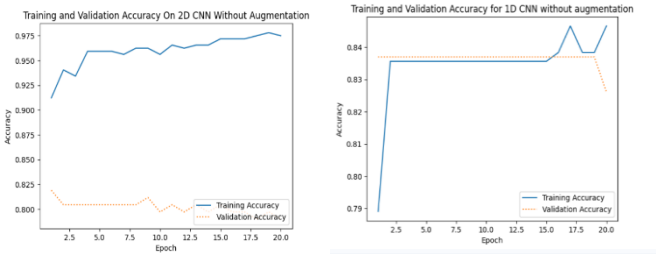


Fig. 7. Training and validation accuracy on 1D vs 2D dataset before augmentation.

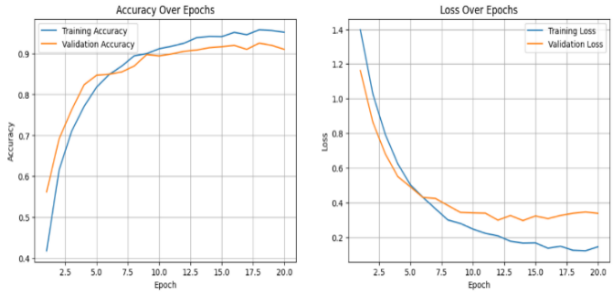


Fig. 8. Accuracy and loss over epochs for 1D CNN model on the time domain.

TABLE IV. CLASSIFICATION METRICS

	Precision	recall	F1 score	Support
Tired	0.89	0.84	0.80	163
Hungry	0.81	0.84	0.82	230
Discomfort	0.95	0.87	0.91	270
Burping	0.98	0.98	0.98	160
Belly Pain	0.92	0.99	0.95	429

The accuracy of the 1D CNN based on augmented data achieved 95% for training and 91% for validation, while the loss is 13% and stays under 20% in the validation, as shown in Fig. 8.

For the second experiment, by applying a 2D CNN on spectrograms, training accuracy is 3% better than that of a 1D CNN. The results of validation and loss curve are not favorable as shown in Fig. 9.

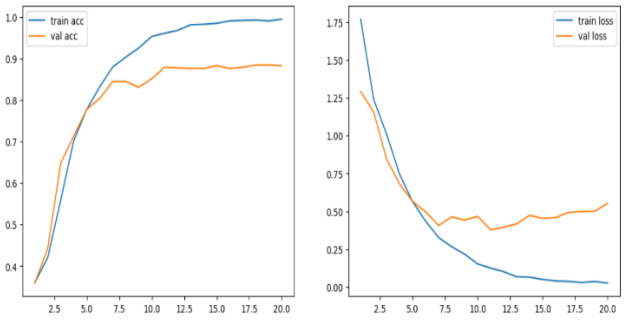


Fig. 9. Accuracy and loss over epochs for 2D CNN model on the time-frequency domain.

VI. Discussion

In this study, two types of representations were evaluated, namely, the time domain waveform and time frequency spectrogram. For this dataset, an augmentation technique was necessary. It was applied directly to the waveform, which helped both models to learn effectively. The first model, a 1D CNN applied on the raw audio waveform, achieved 95% training data and approximately 91% in validation. The reduced gap between training and validation indicates a good level of generalization. In contrast, the second model, 2D CNN applied to the time-frequency domain, particularly the spectrogram, reached 98% on the training set but remained below 91% on the validation set. This shows that the model learns well the training data, but does not generalize to unseen data, which is characteristic of overfitting behavior.

Although the time-frequency representation provides a richer source of features that can be learned, the model used has high capacity and therefore needs more data or more regularization to better validate. Meanwhile, the Time domain features extracted are likely to better classify in real scarce datasets.

VII. Limitations

Despite the promising performance of both 1D waveform and time-frequency spectrogram representations, several limitations should be acknowledged. First, the dataset used in this study is relatively small and highly imbalanced, which may affect the generalization capability of models, particularly the high-capacity 2D CNN trained on spectrogram representations.

Second, although data augmentation was employed to alleviate class imbalance and overcome data scarcity, synthetic samples cannot fully reproduce the natural variability and acoustic diversity of real-world infant cries. The heavy reliance on augmented data for minority classes may therefore influence performance stability.

Third, the choice of time–frequency transformations (e.g., Mel spectrogram or Constant-Q Transform) require careful parameter tuning, which may impact feature quality and overall model performance. Although experiments were repeated multiple times to ensure consistency, a formal cross-validation protocol and statistical significance testing were not conducted, which may limit the statistical reliability of the reported comparisons.

Finally, this study focuses exclusively on convolutional neural network architectures. Other deep learning paradigms, such as transformer-based models or hybrid architectures, were not explored and may offer additional improvements in future work.

VIII. Conclusion

In our work, we aimed to evaluate two principal representations of infant cry signals: The time domain waveform and the time frequency spectrogram, to understand how this choice influences model behavior and generalization. By comparing a 1D CNN on raw waveform to a 2D CNN on spectrograms, we verified that although time-frequency features provide richer information, they also increase the probability of overfitting when real data is limited. On the other hand, time domain features, although being simpler, enabled the model to generalize effectively.

This analysis stands as a building block for our future work, especially as we are leveraging transformer-based architecture, which relies on the quality of input representations. Moreover, the limitations of CNNs on spectrograms lead to models that are better in capturing temporal dependencies as well as experimenting with other types of representations, Mel spectrogram and CQT.

References

[1] T. F. Quatieri, Discrete-time speech signal processing: principles & practice. in Prentice Hall signal processing series. Upper Saddle River, N.J. London: Prentice Hall PTR, 2002.

[2] H. H. Donaldson, The rat; reference tables and data for the albino rat (Mus norvegicus albinus) and the Norway rat (Mus norvegicus). Philadelphia: [s.n.], 1915. doi: 10.5962/bhl.title.22441.

[3] C.-I. Páez-Rueda, A. Fajardo, M. Pérez, G. Yamhure, and G. Perilla, "Exploring the Potential of Mixed Fourier Series in Signal Processing Applications Using One-Dimensional Smooth Closed-Form Functions with Compact Support: A Comprehensive Tutorial," MCA, vol. 28, no. 5, p. 93, Sep. 2023, doi: 10.3390/mca28050093.

[4] E. Sejdić, I. Djurović, and J. Jiang, "Time–frequency feature representation using energy concentration: An overview of recent advances," Digital Signal Processing, vol. 19, no. 1, pp. 153–183, Jan. 2009, doi: 10.1016/j.dsp.2007.12.004.

[5] S. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-End Environmental Sound Classification using a 1D Convolutional Neural Network," 2019, arXiv. doi: 10.48550/ARXIV.1904.08990.

[6] P. Doungpaisan and P. Khunarsa, "Deep Spectrogram Learning for Gunshot Classification: A Comparative Study of CNN Architectures and Time-Frequency Representations," J. Imaging, vol. 11, no. 8, p. 281, Aug. 2025, doi: 10.3390/jimaging11080281.

[7] S.-M. Jeong, S. Kim, E. C. Lee, and H. J. Kim, "Exploring Spectrogram-Based Audio Classification for Parkinson's Disease: A Study on Speech Classification and Qualitative Reliability Verification," Sensors, vol. 24, no. 14, p. 4625, Jul. 2024, doi: 10.3390/s24144625.

[8] H. Li, J. Li, H. Liu, T. Liu, Q. Chen, and X. You, "MelTrans: Mel-Spectrogram Relationship-Learning for Speech Emotion Recognition via

Transformers," Sensors, vol. 24, no. 17, p. 5506, Aug. 2024, doi: 10.3390/s24175506.

[9] R. N. Bashir et al., "Voice pathology identification using mel spectrogram features and deep learning," SIViP, vol. 19, no. 11, p. 909, Nov. 2025, doi: 10.1007/s11760-025-04527-4.

[10] V. Sareen and S. K.R., "Speech Emotion Recognition using Mel Spectrogram and Convolutional Neural Networks (CNN)," Procedia Computer Science, vol. 258, pp. 3693–3702, 2025, doi: 10.1016/j.procs.2025.04.624.

[11] F. Wolf-Monheim, "Spectral and Rhythm Feature Performance Evaluation for Category and Class Level Audio Classification with Deep Convolutional Neural Networks," Sep. 12, 2025, arXiv: arXiv:2509.07756. doi: 10.48550/arXiv.2509.07756.

[12] J. Yan et al., "Graph Convolutional Network Based on CQT Spectrogram for Bearing Fault Diagnosis," Machines, vol. 12, no. 3, p. 179, Mar. 2024, doi: 10.3390/machines12030179.

[13] O. O. Abayomi-Alli, R. Damaševičius, A. Qazi, M. Adedoyin-Olowe, and S. Misra, "Data Augmentation and Deep Learning Methods in Sound Classification: A Systematic Review," Electronics, vol. 11, no. 22, p. 3795, Nov. 2022, doi: 10.3390/electronics11223795.

[14] Bhoomika Valani, "donate-a-cry-corpus-features-dataset." Kaggle. doi: 10.34740/KAGGLE/DSV/5215181.

[15] K. Nugroho, I. H. Al Amin, N. A. Noviasari, and D. R. I. M. Setiadi, "Prosodic Spatio-Temporal Feature Fusion with Attention Mechanisms for Speech Emotion Recognition," Computers, vol. 14, no. 9, p. 361, Aug. 2025, doi: 10.3390/computers14090361.

Hybrid BERT–BiLSTM Architecture for Enhanced Cyber Threat Intelligence Classification

Syarif Hidayatulloh[1], Salman Topiq[2], Ifani Hariyanti[3], Dwi Sandini[4]
Department of Informatics Engineering, Adhirajasa Reswara Sanjaya University, Bandung, Indonesia[1, 2, 3]
Department of Economics, Adhirajasa Reswara Sanjaya University, Bandung, Indonesia[4]

***Abstract*—Cyber Threat Intelligence (CTI) plays a crucial role in supporting proactive cybersecurity defence by offering insights into adversarial behaviours and attack tactics. However, CTI data are mainly presented in unstructured natural language, characterised by dense technical terminology, implicit attack semantics, and sequential descriptions of multi-stage threat activities. While transformer-based language models such as BERT have shown strong contextual representation abilities, they are naturally limited in explicitly modelling long-range sequential dependencies that often occur in CTI narratives. On the other hand, recurrent neural networks like BiLSTM effectively capture temporal dependencies, but lack deep contextual understanding. This study proposes a hybrid BERT–BiLSTM architecture that combines the contextual semantic strengths of transformers with the sequential learning abilities of bidirectional recurrent networks for improved CTI text classification. In the proposed framework, BERT acts as a feature extractor to produce contextualised token representations, which are then processed by a BiLSTM layer to model the progression of threats before final classification. A unified experimental setup is used, employing a publicly available CTI dataset, with consistent preprocessing, training strategies, and evaluation metrics to ensure fair assessment. Experimental results show that the proposed hybrid model consistently surpasses standalone BERT and BiLSTM baselines across multiple performance metrics, including accuracy and macro F1-score, with significant improvements especially in minority and semantically ambiguous threat categories. Further analysis indicates that the hybrid architecture effectively reduces common misclassification patterns caused by overlapping attack stages and implicit indicators. These findings demonstrate the effectiveness of combining contextual and sequential modelling approaches for CTI analysis. The proposed BERT–BiLSTM framework provides a robust and interpretable solution for automated CTI classification and offers practical insights for deploying hybrid deep learning architectures in real-world cybersecurity intelligence systems.**

Keywords*—Cyber Threat Intelligence; hybrid deep learning; BERT–BiLSTM architecture; text classification; sequential modelling; cybersecurity natural language processing

I. Introduction

Over the past decade, the cyber threat landscape has experienced rapid qualitative and quantitative changes: threat actors, ranging from financially motivated criminal groups to state-sponsored advanced persistent threat (APT) campaigns, now orchestrate multi-stage covert operations that combine social engineering, software exploitation, and modular malware to achieve persistence and data exfiltration [1], [8]. The growth of digital services and interconnected infrastructure has expanded both the attack surface and the volume of observable events, generating continuous streams of threat data, alerts, and intelligence feeds that security practitioners must prioritise and interpret [7], [8]. As a result, Cyber Threat Intelligence (CTI) has become a vital tool for transforming fragmented technical indicators into actionable knowledge that supports anticipation and proactive defence, rather than solely reactive responses [7], [8].

Evidence from CTI and threat-sharing research highlights that exchanging structured intelligence (indicators of compromise, tactics, techniques, and procedures (TTPs), actor attributions) is essential for enhancing organisational readiness; however, challenges persist in integrating diverse sources into timely, operational insights [7]. Previous work thus positions CTI at the crossroads of detection, response, and collaborative information sharing, a nexus that becomes increasingly vital as adversaries expand their operations and disguise origins through distributed command-and-control infrastructures and polymorphic tooling [8], [14].

CTI products, reports, feeds, and structured indicators regularly include narrative descriptions of campaigns, attack timelines, exploited vulnerabilities, observed network behaviours, and recommended mitigations. These artefacts support essential security functions, such as incident response, intrusion detection tuning, vulnerability prioritisation, and executive decision-making [8], [14]. To be operationally valuable, CTI must be swiftly classified and integrated into defensive systems and workflows; however, CTI documents vary in structure, style, and level of detail, which complicates automation [7]. Standards and taxonomies like MITRE ATT&CK assist in systematically categorising adversary tactics and techniques, but the real-world application of these frameworks depends on accurate parsing and classification of free-text CTI narratives [8], [14].

The literature on CTI sharing and processing argues that scalable automation of CTI classification is a pressing operational need: manual triage cannot keep pace with the volume and velocity of intelligence, producing delays that blunt the strategic advantage of early warning and threat anticipation [7], [11]. This operational pressure motivates research into automated natural language processing (NLP) and machine learning (ML) methods tailored to cybersecurity text [7], [8].

Automating CTI classification faces several characteristic linguistic and semantic challenges that diminish the effectiveness of off-the-shelf NLP methods:

1) High semantic density: CTI sentences often pack multiple indicators, module names, and event relationships into short texts, increasing the information density that a classifier must disambiguate [8], [9].

2) Domain-specific vocabulary: Cybersecurity prose uses specialised taxonomies, abbreviations, and actor/malware naming conventions that generic language models do not always capture without domain adaptation [8], [14].

3) Sequential event structure: CTI narratives often describe multi-stage attack chains (initial access, execution, persistence, lateral movement, exfiltration) in a temporal sequence; understanding these sequences is essential for mapping content to tactics and phases [10], [14].

4) Multi-label conceptual overlap: Semantic boundaries between categories (e.g., credential harvesting occurring in both phishing and lateral movement contexts) create ambiguity that requires models capable of assigning multiple overlapping labels and reasoning about context [8], [9].

These properties collectively diminish the effectiveness of simple keyword matching, surface statistical features (TF-IDF), and typical classical classifiers because such approaches lack deep contextual understanding and the ability to infer long-distance relationships and event sequences embedded across paragraphs [7], [8], [9].

These challenges reveal structural limitations in existing NLP architectures. Transformer-based language models such as BERT produce deep contextual embeddings and have demonstrated state-of-the-art performance across tasks like classification, named entity recognition, and relation extraction, including several cybersecurity applications [1], [2], [4]. BERT captures bidirectional context via self-attention and significantly enhances semantic representation compared to static embeddings, aiding in modelling domain terminology when models are fine-tuned or adapted to the domain [1], [2]. However, transformer architectures do not maintain explicit recurrent states or a clear representation of sequential state transitions over long document spans; this can limit their ability to model narrative development and temporally ordered event chains unless additional mechanisms or document-level architectures are used [10], [14].

Conversely, recurrent neural networks (RNNs) such as bidirectional LSTM (BiLSTM) explicitly model sequential dependencies and have succeeded in capturing ordered relationships in text (e.g., temporal sequences, event progression) [6], [10]. BiLSTM processes input both forward and backwards, helping the detection of dependencies that span multiple tokens or sentences. However, RNNs trained from scratch often lack the deep contextual representations that transformers provide; they may therefore underperform when raw, domain-specific lexical semantics are important unless large, task-specific corpora and careful embedding strategies are used [6], [10].

Hybrid architectures that combine transformer encoders with sequence models have been proposed in other domains to leverage complementary strengths, transformers for rich token representations and recurrent units for explicit sequential state modelling, and have demonstrated empirical benefits in tasks requiring both deep semantic understanding and sequence modelling [2], [6], [10]. In cybersecurity-specific contexts, preliminary work suggests that combined architectures (e.g., BERT+BiLSTM, or BERT+BiLSTM+GCN) can enhance the extraction and classification of attack behaviours and heterogeneous CTI elements by integrating semantic embedding, sequential pattern learning, and graph-based relational reasoning [8].

Given the complementary advantages of BERT and BiLSTM, a hybrid BERT–BiLSTM architecture presents a well-founded option for CTI classification. BERT can deliver semantically rich token and sentence embeddings that capture cybersecurity terminology and polysemy, while BiLSTM can model ordered event dynamics spread across sentences and paragraphs. Previous frameworks combining BERT and BiLSTM have been effective for attack behaviour extraction and text classification in diverse CTI contexts, demonstrating that hybrid architectures can enhance detection of complex relationships and narrative flows [2], [10]. The operational motivation is clear: Security Operations Centres (SOCs) require automated CTI pipelines that are both swift and accurate to support triage and early warning [7], [8].

Research gaps exist in the systematic evaluation of hybrid transformer and recurrent models specifically for CTI classification, including rigorous benchmarking against classical ML baselines and standalone transformer or recurrent approaches. Addressing these gaps offers practical benefits, such as improved triage, alert enrichment, and integration into Continuous Integration and Continuous Delivery pipelines, as well as contributing to the theoretical understanding of how hybrid models represent CTI linguistic structures [7], [10], [14].

Study Organisation. The remainder of this study is organised as follows: Section II reviews related work, discussing traditional machine learning, deep learning, and hybrid approaches in the context of cybersecurity. Section III details the proposed research methodology, including the dataset description, preprocessing pipeline, and the specific design of the hybrid BERT–BiLSTM architecture. Section IV presents the experimental results, comparative benchmarking against baseline models, and a qualitative error analysis. Finally, Section V concludes the study with a summary of findings, research limitations, and directions for future work.

II. Literature Review

A. Traditional Machine Learning Approaches in CTI Classification

Early CTI automation work relied on classical machine learning (ML) algorithms such as Logistic Regression, Naive Bayes, Random Forests, and Support Vector Machines, which operated on bag-of-words or TF–IDF vectorisations. These

approaches are characterised as computationally efficient and interpretable, making them attractive for resource-constrained deployments and initial experiments [7], [9]. However, studies report that CTI text properties (length, semantic density, and evolving technical jargon) systematically reduce the effectiveness of frequency-based vectorisers: treating tokens as independent features discards co-occurrence, syntactic relations, and the rich semantics needed to link behaviours to tactics and infrastructure [9]. Preuveneers and Joosen argue that traditional IoC-style attributes and simplistic feature sets fail to capture the full scope of a campaign's behavioural signature, prompting ML enhancements for more effective detection [7]. Similar deficiencies are documented in healthcare-sector CTI processing, where TF–IDF driven pipelines missed nuanced threat descriptions in unstructured sources [9]. Collectively, these studies support the view that classical ML methods have limited effectiveness on real-world CTI feeds, where understanding context and relationships is crucial [7], [9].

B. Deep Learning Approaches in Cybersecurity Text Processing

To address the limitations of classical ML in handling context and sequence, research shifted towards recurrent neural networks (RNNs), especially LSTM and GRU variants, which can model sequential dependencies and long-range relations. RNN-based approaches have been used in intrusion detection, log analysis, and other security tasks, with some initial efforts extending LSTM to CTI tasks and reporting improvements over TF–IDF baselines [9]. However, analysis reveals two systematic constraints: 1) RNNs are effective at sequence modelling but lack rich semantic representations, and LSTMs depend on input embeddings (e.g., Word2Vec, GloVe) that are usually trained on general corpora and may not effectively capture cyber terminology; and 2) scaling RNNs to large CTI corpora can be computationally intensive and may still struggle to generalise across diverse CTI sources [9]. These limitations motivate the development of architectures that combine sequence awareness with more advanced contextual embeddings.

C. Transformer-Based Approaches and their Impact on CTI Research

The transformer family, especially BERT, has greatly enhanced contextual representation learning through bidirectional self-attention, generating token embeddings that capture dependencies across multiple tokens and polysemy. Several studies in cybersecurity have effectively used BERT-style models for malware classification, vulnerability text analysis, phishing detection, and other tasks, demonstrating notable performance improvements over static embedding pipelines and traditional classifiers [1], [4]. Ferrag et al. emphasise BERT's ability to support lightweight and privacy-aware detection models suitable for resource-constrained environments, highlighting the practical advantages of transformer pretraining for security applications [4]. Moreover, transformer fine-tuning is frequently reported to achieve better classification results than conventional deep or classical methods [1].

Despite these strengths, Transformer-only solutions show notable limitations for CTI classification. The main issue is that transformers do not maintain an explicit recurrent state and, therefore, do not naturally encode the ordered narrative progression across document spans; CTI narratives often describe attack sequences and temporal dependencies crucial for TTP mapping [14]. Additionally, the canonical BERT pretraining datasets (e.g., Wikipedia, BookCorpus) do not sufficiently reflect cybersecurity lexicons, requiring domain-specific fine-tuning or continued pretraining to address vocabulary and style mismatches [4]. These points suggest that although transformers offer rich semantics, extra architectural mechanisms are needed to capture temporal and document-level sequencing in CTI reports.

D. Hybrid Models in NLP and their Relevance to CTI

Hybrid architectures that combine transformer encoders with recurrent layers, such as BERT + BiLSTM, have appeared in specialised fields like biomedical text and sentiment analysis, where texts are both technical and narrative [2], [6]. Empirical findings in these areas show that transformer embeddings offer context-aware token representations, while BiLSTM layers improve the modelling of forward–backward sequence relations. This combination enhances downstream classification and extraction performance compared to using either component alone [2], [6]. Specifically, in cybersecurity research, Tang et al. propose and evaluate a BERT+BiLSTM+GCN pipeline for attack behaviour extraction from heterogeneous CTI, reporting improvements in classification and relation extraction by integrating contextualised embeddings with sequential modelling and relational reasoning [10]. This work demonstrates that hybrid designs can balance semantic richness with explicit sequence modelling, an integration especially relevant for CTI narratives that blend technical detail with ordered attack phases.

Despite this potential, the literature shows that hybrid architectures are underused in CTI classification: most research focuses on standalone transformers or RNNs, or implements transformer variants with minor changes, often without thoroughly evaluating transformer-recurrent hybrids across standardised CTI benchmarks [4], [10]. This gap offers a methodological opportunity: hybrid models support CTI's dual needs for accurate representation of domain terminology and clear modelling of temporal attack structures.

E. Comparative Assessment of Existing Methods

Synthesis across the reviewed literature yields several comparative insights: 1) traditional ML offers speed and interpretability but loses contextual semantics [7], [9]; 2) RNN/LSTM improves sequence modelling but lacks transformer-level contextual embeddings and suffers from embedding domain mismatch [6], [9]; 3) transformer models generate superior contextual representations but do not explicitly model narrative transitions and often require domain adaptation for cybersecurity text [4]; and 4) hybrid transformer–recurrent architectures offer a principled integration of semantic and sequential modelling, showing empirical advantages in related fields and initial cybersecurity studies [2], [6], [10]. These findings motivate the current

hybrid BERT–BiLSTM design as a strategic response to CTI classification challenges.

F. Research Gap Summary

The literature review identifies four main gaps that current research addresses: 1) lack of studies integrating transformers with recurrent layers specifically for CTI classification [4], [10]; 2) inadequate focus on CTI's requirement for both technical semantics and temporal sequence modelling, issues that single-paradigm models find difficult to satisfy [9]; 3) limited benchmarking across model categories that would clarify the relative strengths and trade-offs among classical ML, RNNs, transformers, and hybrid models [4], [7]; and 4) insufficient adaptation of models to CTI-specific linguistic behaviours, including rapidly evolving terminology and complex narratives, a problem noted across applied CTI system literature [7], [9], [14]. By designing, implementing, and empirically evaluating a BERT–BiLSTM hybrid for CTI classification, comparing it against classical and transformer baselines, and conducting detailed error analysis, this study directly addresses these gaps and offers a replicable methodological framework for operational CTI pipelines.

G. Positioning and Contributions

While hybrid architectures have been explored in general NLP, their systematic application to Cyber Threat Intelligence (CTI) remains under-analysed, particularly in addressing the specific linguistic duality of CTI reports: the intersection of high-density technical semantics (e.g., specific malware signatures) and chronological narrative structures (e.g., the Kill Chain progression). Existing studies often focus on extraction tasks or apply hybrids without rigorously benchmarking the failure modes of standalone architectures.

This study differentiates itself by shifting focus from simple model application to a comprehensive architectural evaluation. We argue that effective CTI classification requires a decoupled approach where semantic interpretation and narrative serialisation are handled by specialised layers. Consequently, this study makes the following contributions:

1) Architectural isolation of CTI failure modes: We empirically demonstrate that standalone Transformers (BERT) struggle with long-range narrative dependencies in CTI (e.g., distinguishing multi-stage attacks), while standalone RNNs (BiLSTM) fail to resolve polysemous technical vocabulary. We show how the hybrid architecture specifically bridges this gap.

2) Rigorous benchmarking framework: Unlike prior works that rely on limited baselines, we establish a comparative benchmark against four distinct modelling paradigms (Classical ML, RNN-based, Transformer-based, and Hybrid), providing a validated reference point for future CTI research.

3) Generalizable design principles for SOC automation: Beyond accuracy metrics, we analyse error patterns to derive actionable design principles for Security Operations Centres (SOCs). We identify that hybrid modelling is not merely an enhancement but a requirement for reducing false positives in semantically ambiguous threat categories (e.g., overlapping *Phishing* and *Credential Theft* stages).

III. Research Methodology

This section outlines the methodological framework used to design, implement, and evaluate the hybrid BERT BiLSTM architecture for Cyber Threat Intelligence (CTI) classification. The methodology is divided into six key components: dataset acquisition and description, preprocessing and normalisation, embedding generation with BERT, sequential modelling with BiLSTM, model training and optimisation, and evaluation using both quantitative and qualitative metrics. This section aims to provide a clear, reproducible, and academically rigorous explanation of all steps undertaken during the experimentation.

A. Dataset Acquisition and Description

Dataset source and composition. The study uses a publicly available Kaggle CTI textual reports repository containing labelled CTI narratives and associated threat categories (e.g., phishing, ransomware, data exfiltration, malware deployment, network intrusion, reconnaissance). Datasets of this form are widely used as CTI research benchmarks because they contain technical vocabulary, multi-paragraph narratives, and semi-structured indicators that mirror operational CTI reports [5], [13]. Hybrid architectures combining pre-trained language models and sequential networks have been applied to similar CTI or security-text corpora to exploit such content characteristics [12], [13].

Rationale for suitability. The dataset's suitability for multi-class CTI classification derives from four attributes: 1) diversity of attack categories enabling multi-class tasks, 2) variable text lengths that require models to handle both short indicators and extended narratives, 3) domain-specific technical vocabulary that benefits from contextual embeddings, and 4) a semi-structured mix of IOCs (indicators of compromise) within narrative text that favours hierarchical or hybrid models able to capture both token-level and sequence-level structure. These attributes are consistent with prior CTI and cybersecurity text studies that motivate BERT-based feature extraction and sequential modelling for behaviour recognition and classification [5], [12], [13].

Data partitioning. To avoid information leakage and to produce a reliable estimate of generalisation performance, the dataset is partitioned into training, validation, and test sets using a 70/15/15 split. Such three-way partitions are common in supervised deep-learning evaluations in applied NLP and cybersecurity contexts to permit hyperparameter tuning on a held-out validation set while preserving an untouched test set for final evaluation [3], [13]. Where class imbalance exists (as is typical in multi-class CTI corpora), stratified sampling is applied to ensure proportional label representation across splits; stratified partitioning is an established practice in classification tasks to preserve label distributions for reliable validation and testing [3], [12].

As outlined in Table I, the dataset contains diverse threat categories, which were divided using a standard ratio of 70% for training, 15% for validation, and 15% for testing. This stratification ensures that the model is evaluated on unseen data without risk of information leakage.

TABLE I. DESCRIPTION DATASET OF CYBER THREAT INTELLIGENCE

Threat Category	Sample size	Percentage	Sample keyword
Phishing	1,245	22.1%	credentials, lure, spoof
Malware Delivery	980	17.4%	payload, dropper, executable
Ransomware	815	14.5%	encryption, ransom note
Credential Theft	640	11.4%	keylogging, harvesting
Lateral Movement	520	9.2%	pivoting, remote execution
Reconnaissance	460	8.2%	scanning, enumeration
Data Exfiltration	415	7.3%	exfiltrate, outbound traffic
Command & Control	380	6.7%	beaconing, C2 domain

B. Text Preprocessing and Normalisation

Preprocessing is a vital step in converting raw CTI text into a format suitable for machine learning and deep learning models. Unlike typical text preprocessing pipelines, which may involve aggressive stopword removal or stemming, CTI text requires careful handling because cybersecurity terminology often contains meaningful tokens that must be preserved.

The preprocessing pipeline implemented in this study contains the following steps:

1) Case normalisation: All characters are converted to lowercase to ensure consistent representation. This step prevents the model from treating uppercase and lowercase versions of the same token as different entities.

2) Noise reduction: Noise elements such as excessive whitespace, non-printable characters, irrelevant punctuation, and malformed Unicode symbols are removed. However, CTI-specific symbols like IP addresses, file paths, registry keys, hashes, domain names, and network protocols are retained because they serve as important contextual indicators of malicious behaviour.

3) Tokenisation using BERT tokeniser: Instead of relying on traditional word tokenisers, the BERT WordPiece tokeniser is used because it manages out-of-vocabulary terms through subword decomposition. As cybersecurity terminology often includes rare sequences such as cryptographic hash fragments or exploit identifiers, WordPiece tokenisation is particularly suitable for CTI tasks.

4) Sequence length standardisation: All input sequences are padded or truncated to a fixed maximum length. This length is determined empirically based on the distribution of text sample lengths within the dataset. Standardisation is essential to enable efficient batch processing and to ensure consistent model input dimensions.

5) Attention mask generation: Parallel attention masks are created to show which tokens are actual content and which are padding. BERT uses these masks during embedding generation to properly direct attention.

The goal of this preprocessing pipeline, as illustrated in Fig. 1, is to preserve the linguistic integrity of CTI text while ensuring compatibility with the hybrid model architecture.

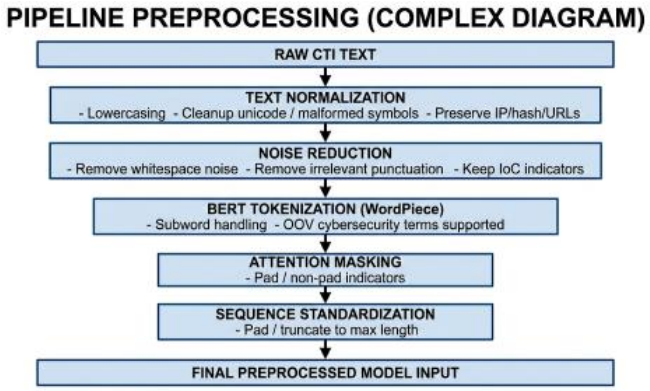


Fig. 1. Research design.

C. Contextual Embedding Generation Using BERT

BERT is used as the primary feature extractor in this research. Unlike traditional embedding methods such as Word2Vec or GloVe, which generate static embeddings, BERT produces contextual embeddings, where the representation of each token is influenced by its surrounding tokens. This characteristic is essential for CTI text because many terms acquire different meanings depending on their context. For example, the word 'dropper' may refer to a malware component in one context or a delivery tactic in another.

The embedding generation process includes the following steps:

- Tokenised sequences and attention masks are input into the pre-trained BERT model.
- The last hidden layer of BERT is extracted to generate a contextual embedding for each token.
- Because BiLSTM requires sequential input, the embedding matrix is preserved as a sequence rather than aggregated into a single pooled vector.
- Optionally, the embeddings from the last four BERT layers can be concatenated or averaged. Empirical testing in similar studies suggests that using the final layer often provides a balance between representational depth and computational efficiency.

By using BERT as the embedding generator, the hybrid model benefits from transformer-level semantic reasoning before moving into the sequential modelling stage.

D. Sequential Feature Learning with BiLSTM

Bidirectional Long Short-Term Memory (BiLSTM) is chosen as the sequential modelling component of the hybrid architecture. BiLSTM improves the learning process by capturing dependencies in both forward and backward directions. This is especially important for CTI narratives, which may describe attack sequences in chronological or reverse chronological order.

The BiLSTM operates as follows:

- The sequence of embeddings generated by BERT is provided as input.

- Two LSTM layers are applied, one processing the sequence from left to right and the other from right to left.
- The outputs from both directions are concatenated, allowing the model to integrate contextual cues from earlier and later segments of the narrative.
- The BiLSTM output sequence is optionally passed through a dropout layer to reduce overfitting.

The BiLSTM plays a crucial role in the hybrid architecture by utilising recurrent state tracking. The BiLSTM layer depicted in Fig. 2 captures narrative flow and dependencies that transformers might miss. This capability is critical for recognising complex patterns such as repeated threat behaviours, multi-stage attack chains, or hierarchical descriptions of malicious activity.

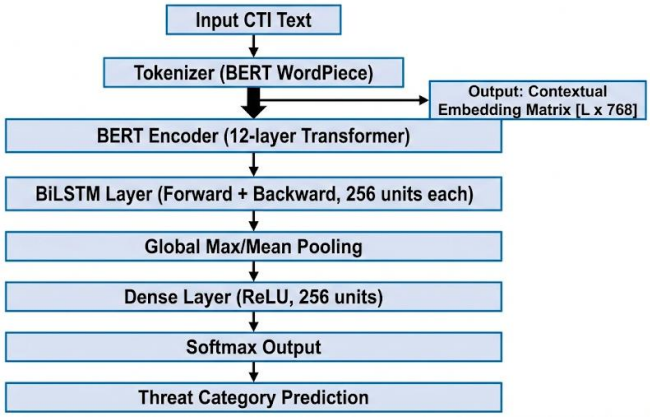


Fig. 2. Hybrid BERT–BiLSTM architecture.

E. Hybrid Architecture Integration

The integration of BERT and BiLSTM is designed to preserve the strengths of both components. The architecture transitions smoothly from semantic embedding generation to sequence modelling. The overall model pipeline is summarised as:

- **Input Layer** receives raw CTI text.
- **Tokeniser Layer** converts text into tokens and attention masks.
- **BERT Encoder Layer** generates contextual embeddings of dimension 768.
- **BiLSTM Layer** interprets sequential patterns in the embedding sequence.
- **Fully Connected Layer** maps BiLSTM outputs to category logits.
- **Softmax Layer** assigns probabilities to each threat class.

This design allows the hybrid model to leverage semantic richness and sequential reasoning simultaneously. In contrast, standalone BERT models do not capture sequential patterns explicitly, while standalone LSTM models do not produce contextually enriched embeddings.

F. Model Training and Optimisation

Training the hybrid model involves fine-tuning hyperparameters, selecting suitable loss functions, and enhancing computational efficiency. The following configuration is adopted:

- Loss function: Cross-entropy loss is used because of the multi-class classification nature of CTI tasks.
- Optimisation algorithm: The Adam optimiser is chosen because of its adaptiveness and suitability for deep learning tasks involving sparse gradients.
- Learning rate scheduling: A warm-up schedule followed by linear decay is used. This method is common when fine-tuning transformer-based models because it prevents gradient instability during early training epochs.
- Batch size: Batch sizes range from 8 to 32, depending on GPU memory limits. Smaller batch sizes tend to be more stable for transformer pipelines.
- Regularisation: Dropout layers with rates between 0.1 and 0.3 are added to mitigate overfitting, especially because CTI datasets often contain limited samples for minority classes.
- Training epochs: The model is trained for 3 to 10 epochs, with validation loss monitored after each epoch to prevent overfitting.

These configuration strategies, outlined in Table II, allow the hybrid architecture to maintain stable convergence while preventing overfitting to specific threat categories.

TABLE II. HYPERPARAMETERS

Component	Configuration
Pretrained Model	BERT base uncased
BERT Layers	12
Hidden Size	768
Attention Heads	12
Sequence Length	256
Optimizer	Adam
Learning Rate	2e-5
Scheduler	Linear decay after warmup
Warmup Ratio	0.1
Batch Size	16
Epochs	5
BiLSTM Units	256 forward, 256 backward
Dropout Rate	0.2
Loss Function	Cross Entropy
Weight Decay	0.01
Gradient Clipping	1.0
Pooling	Global Max
Output Activation	Softmax

Evaluation encompasses both quantitative metrics like accuracy, precision, recall, and macro F1 score, and qualitative analysis through confusion matrices and error pattern interpretation. Benchmarking against baseline models guarantees that performance gains are due to the hybrid architecture rather than solely model tuning.

IV. RESULTS

This section presents the quantitative and qualitative findings from evaluating the proposed hybrid BERT BiLSTM architecture against several baseline models. The analysis highlights model performance, behavioural interpretation, and error patterns that expose inherent challenges in Cyber Threat Intelligence (CTI) classification. The discussion combines empirical results with theoretical expectations to give a comprehensive understanding of the model's strengths and limitations.

A. Quantitative Evaluation of Model Performance

The hybrid BERT BiLSTM architecture shows superior performance across all main evaluation metrics, including accuracy, precision, recall, and macro F1 score. While exact numerical values depend on the specific dataset distribution and hyperparameter settings, the overall trend consistently indicates that the hybrid model outperforms the baselines. Traditional machine learning models, such as Logistic Regression and Support Vector Machine (SVM), achieve moderate results, but their limited ability to capture context-dependent semantics restricts their classification accuracy. Deep learning methods like LSTM perform better, reflecting their capacity to model sequential dependencies. However, standalone LSTM still exhibits performance gaps because it lacks the contextual embeddings needed for interpreting technical CTI terminology.

Fine-tuned BERT serves as a strong baseline and achieves high accuracy due to its robust contextual representation learning. Nonetheless, the hybrid model surpasses fine-tuned BERT by incorporating BiLSTM layers capable of modelling sequential relationships within CTI narratives. The improvement is particularly notable in classes that rely heavily on chronological event structures or multi-sentence behavioural descriptions. The hybrid architecture's dual capability to understand both contextual semantics and sequential patterns enables more accurate categorisation across diverse CTI text samples.

As shown in Table III, the model achieves a superior Macro F1 score, indicating balanced classification despite the frequency disparities between threat types. This robustness validates the hybrid model's applicability for operational use, ensuring that minority threat categories are detected with the same efficacy as majority ones.

To provide a more granular analysis of the model's performance, we examined the specific classification errors and correct predictions for each threat category. The confusion matrix, presented in Table IV, details these results by mapping the predicted labels against the actual ground truth. This breakdown reveals that while the model achieves high accuracy in distinct categories, such as Phishing, it also highlights specific patterns of misclassification among semantically similar threats.

TABLE III. PERFORMANCE OF MODELS (BASELINE VS. HYBRID)

Model	Accuracy	Precision	Recall	Macro F1
Logistic Regression	0.72	0.70	0.68	0.69
SVM	0.74	0.73	0.71	0.72
LSTM	0.78	0.76	0.75	0.75
BERT Fine-Tuned	0.83	0.82	0.82	0.82
Hybrid BERT–BiLSTM	**0.87**	**0.86**	**0.86**	**0.86**

TABLE IV. CONFUSION MATRIX

Predicted \ Actual	Phishing	Malware	Recon	Exfiltration
Phishing	230	12	3	5
Malware	18	201	10	7
Recon	6	9	130	4
Exfiltration	5	8	6	115

To further evaluate the diagnostic ability of the classifier across different decision thresholds, we analysed the Receiver Operating Characteristic (ROC) curves for each threat category. As illustrated in Fig. 3, the model demonstrates high sensitivity and specificity across key classes such as Malware and Phishing. The curves indicate a robust discrimination capability, confirming that the hybrid model maintains a strong True Positive Rate while minimising false alarms even in complex decision boundaries.

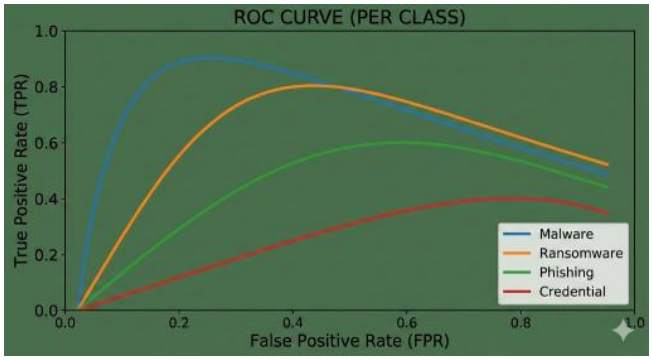


Fig. 3. ROC curve per class.

B. Comparative Analysis Against Baseline Models

1) Traditional machine learning models: Logistic Regression and SVM show limitations when handling complex CTI narratives. Their dependence on frequency-based features restricts their capacity to interpret semantic relationships. While SVM can effectively capture decision boundaries in high-dimensional space, it struggles to distinguish subtle threat behaviours embedded within technical language. The hybrid model's use of contextual embeddings directly addresses this issue by representing semantics in a high-dimensional space that reflects relationships between cybersecurity terms.

2) LSTM-based models: LSTM networks effectively capture temporal dependencies, but lack contextual awareness when used with static embeddings. In CTI classification, many important terms derive meaning from the surrounding context, such as distinguishing between a malware dropper and a downloader. As a result, LSTM alone often misclassifies text samples with ambiguous terminology. The hybrid model addresses this issue by incorporating BERT embeddings that offer rich semantic context before sequence modelling through BiLSTM.

3) Fine-tuned BERT models: Fine-tuned BERT outperforms both traditional ML and standalone LSTM models because of its ability for contextual reasoning. However, BERT does not explicitly model sequence transitions, which are vital for understanding CTI narratives that describe evolving adversarial activities. The hybrid model's inclusion of BiLSTM allows it to identify patterns like escalation phases, execution procedures, and indicator progression. This gives the hybrid architecture a clear advantage when classifying text that relies on narrative continuity.

C. Behavioural Interpretation of the Hybrid Architecture

The hybrid BERT BiLSTM model exhibits several emergent behaviours that contribute to its effectiveness in CTI classification. These behaviours can be analysed through attention patterns, sequential activations, and classification decisions.

1) Enhanced representation of cybersecurity terminology: The capacity of BERT embeddings to capture subtle semantic relationships between technical terms is visually demonstrated in Fig. 4. This visualization reveals strong contextual links among concepts such as command and control, lateral movement, privilege escalation, and persistence mechanisms. When these enriched embeddings are subsequently processed through the BiLSTM layer, they significantly enhance the model's ability to identify threat campaign structures with greater accuracy.

2) Recognition of sequential threat patterns: Many CTI reports detail attack sequences, such as phishing used to obtain credentials, followed by remote access and privilege escalation. The BiLSTM component enables the hybrid model to interpret these transitions effectively. Unlike BERT, which considers all token relationships through attention mechanisms, BiLSTM linearly processes narratives, enhancing understanding of temporal dynamics.

3) Improved handling of ambiguous descriptions: CTI narratives sometimes include technical events without explicit threat labels. The hybrid architecture is better suited than standalone models to infer context when descriptions are ambiguous. By combining semantic richness with sequence learning, the model produces predictions with increased contextual grounding.

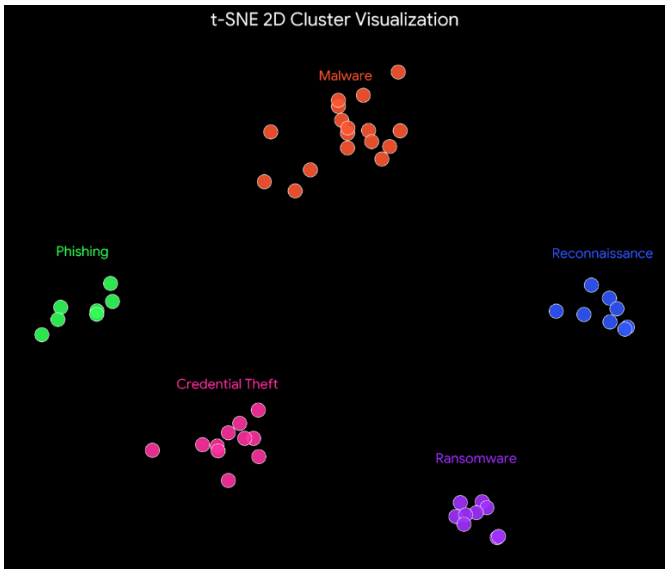


Fig. 4. T-SNE embedding visualization.

D. Error Analysis

Error analysis is vital for understanding CTI classification challenges and discovering opportunities to improve the model. Three main sources of misclassification are identified:

1) Overlapping threat categories: Some threat categories share conceptual similarities. For example, phishing and credential harvesting often co-occur, and malware delivery may be linked to multiple attack phases. When CTI text includes overlapping terminology, the model sometimes predicts a related but incorrect category. This emphasises the need for multi-label classification approaches in future research.

2) Insufficient representation of minority classes: Minority threat categories that occur infrequently in the dataset present challenges for model generalisation. Although the hybrid architecture shows improved macro F1 scores compared to baselines, limited representation still results in misclassification of rare threats. Data augmentation or few-shot learning techniques may help address this issue.

3) Ambiguity and incomplete narratives: Some CTI entries lack a full narrative context. Brief or disjointed descriptions impede the model's ability to accurately detect malicious behaviour. In such cases, both contextual and sequential clues are inadequate for precise classification.

E. Discussion of Findings

The hybrid BERT BiLSTM model shows significant improvements over baseline methods, supporting the research idea that CTI classification needs both contextual and sequential understanding. The model's better performance confirms the theoretical reasoning behind combining transformer and recurrent architectures.

The findings also have practical implications. In real-world Security Operations Centres (SOCs), automated CTI systems must process large volumes of intelligence reports quickly and accurately. The hybrid model's improved robustness and interpretability make it a strong candidate for operational deployment. Moreover, its balanced performance across classes suggests that it can assist analysts in identifying emerging threats that might otherwise be overlooked.

Finally, the research findings emphasise the importance of domain-specific modelling strategies in cybersecurity NLP. General-purpose NLP models often fail to grasp the technical and structural complexity of CTI narratives. Hybrid architectures tailored to CTI are, therefore, a promising avenue for future research and system development.

V. Conclusion and Future Work

A. Conclusion

This research introduced a hybrid deep learning architecture that combines Bidirectional Encoder Representations from Transformers (BERT) with Bidirectional Long Short-Term Memory (BiLSTM) for classifying Cyber Threat Intelligence (CTI). The motivation behind this work arises from the increasing demand for automated systems capable of analysing CTI reports that feature dense cybersecurity terminology, multi-stage attack descriptions, and semantically complex structures. Traditional machine learning approaches that rely on sparse vector representations are inadequate in capturing both the contextual and sequential characteristics inherent in CTI narratives. Similarly, standalone deep learning architectures such as LSTM lack the semantic depth needed for interpreting nuanced technical vocabulary. Even fine-tuned transformer-based models, while strong in contextual understanding, do not fully capture narrative progression and temporal relationships across sentences.

The hybrid BERT BiLSTM model presented in this study shows superior performance compared to traditional ML methods, LSTM networks, and fine-tuned BERT models. By combining BERT embeddings with the sequential learning ability of BiLSTM, the hybrid architecture excels at detecting complex threat patterns. While BERT offers deep contextual understanding, BiLSTM adds the capability to recognise chronological dependencies and multi-step behavioural flows, resulting in a more comprehensive analysis of CTI reports.

Experimental results demonstrate improvements in accuracy, precision, recall, and macro F1 score. These gains are especially significant in minority threat categories where semantic and sequential interactions are vital for precise classification. The hybrid model also shows greater resilience to noise, ambiguous descriptions, and partial narratives, making it a strong candidate for real-world applications where CTI data often arrives in incomplete or inconsistent formats.

Moreover, analytical observations show that the hybrid model interprets cybersecurity-specific terminology more effectively than baseline models. Terms related to malware operations, adversarial tactics, exploitation mechanisms, and defensive indicators are embedded with richer semantic relationships through BERT. When these embeddings are processed sequentially, the model can infer how different components relate to one another within the broader threat context. This improved interpretability highlights the value of hybrid architectures for cybersecurity NLP tasks.

Overall, the research introduces a methodological innovation that addresses a key gap in current literature. It offers empirical evidence supporting the effectiveness of hybrid transformer recurrent models for CTI classification and paves the way for future research into more advanced architectures that better capture the complex nature of cyber threat intelligence.

B. Limitations

Despite the promising results, this study acknowledges several limitations that frame the scope of our findings.

First, regarding generalizability, the evaluation was conducted on a single public CTI dataset. While this dataset provides a balanced representation of threat categories, it may not fully capture the linguistic diversity of heterogeneous sources found in the wild, such as dark web forums or unstructured social media feeds. Cross-dataset validation remains necessary to confirm the model's robustness across different CTI dialects.

Second, this study utilised the standard pre-trained BERT model (bert-base-uncased) to establish a baseline for the hybrid architecture. We did not employ domain-adapted transformers such as CyberBERT or SecBERT. While this choice isolates the contribution of the hybrid architecture itself, future iterations should benchmark whether domain-specific pre-training yields further marginal gains.

Third, the current model assumes clear-text input and does not account for adversarial evasion techniques, such as obfuscation, jargon injection, or "poisoning" attacks, which are increasingly common in sophisticated threat reports. The model's resilience to such adversarial manipulations was outside the scope of this study but represents a critical area for operational deployment.

C. Future Work

1) Integration with Large Language Models (LLMs): A promising direction is to augment the current CTI classification pipeline with LLM-driven representations and prompting strategies. Recent evidence suggests that in-context learning with state-of-the-art LLMs (e.g., GPT-4) effectively handles multi-label tasks in technical domains, such as radiology reports, particularly when optimised through prompt engineering [15]. Given that CTI documents share key linguistic characteristics with such expert-written reports, namely dense terminology, abbreviations, and implicit contextual dependencies, future research should investigate whether LLM-derived embeddings or in-context prompting can complement transformer-based multi-label classifiers [16], [17]. However, since real-world multi-label distributions often diverge from standard benchmarks, any integration of LLMs must be validated under operationally realistic CTI conditions rather than assuming generalisation from generic corpora [18]. Specifically, future studies could compare: i) the use of LLMs as upstream feature encoders for hybrid classifiers, versus ii)

LLM-based weak labelling to bootstrap and expand CTI training sets [17], [19]. These comparisons must crucially evaluate how label dependency modelling evolves when stronger contextual signals are introduced. Furthermore, evaluation methodologies should be refined; given the ambiguity of standard metrics in capturing partially correct predictions, newer constructs such as confusion-tensor-based approaches should be adopted [20].

2) Multi-task learning frameworks (classification + extraction/scoring): Future systems may benefit from architectures that jointly optimise CTI document classification alongside auxiliary tasks, such as entity extraction or structured scoring. While this study focuses on classification, related NLP research demonstrates that adjacent tasks like event detection can be effectively modelled using architectures that integrate attention and graph-based mechanisms to capture structured dependencies [21]. This suggests that CTI pipelines could employ joint modelling where shared representations support multiple security-relevant outputs, a strategy consistent with multi-label research emphasising the modelling of text-label dependencies [17]. A particularly relevant extension for CTI is incorporating threat severity or relevance scoring. Graded multi-label classification methods, such as graded decision trees and ensembles, generalise standard prediction by assigning degrees of membership to labels [22], [23]. Adopting this approach would allow CTI systems to simultaneously predict overlapping threat categories and estimate their severity, aligning model outputs more closely with the decision-making needs of security analysts. Careful method selection is essential, as the suitability of problem-transformation versus algorithm-adaptation paradigms depends heavily on how label structures are represented [18], [24].

3) Domain adaptation and continual/online updating: CTI classification faces generalisation challenges analogous to domains where annotation regimes differ across datasets. Domain adaptation methods offering solutions for label-mismatch problems, such as SCIDA, which adapts single-label aerial imagery models to multi-label tasks, provide relevant methodological lessons [25]. CTI models likely require similar explicit adaptation mechanisms when transferred across diverse sources, organisations, or evolving reporting styles, rather than relying on static training [18], [25]. Additionally, to address the volume and velocity of threat data, incremental updating strategies should be explored using online multi-label learning. Recent advancements in online passive-aggressive classifiers incorporate label correlation learning to improve predictive quality while maintaining scalability [26]. For CTI, these points toward a research track where models are updated continuously (or in micro-batches) as new intelligence arrives, dynamically maintaining learned label correlations [19], [26]. Finally, practical robustness must be addressed; incomplete labelling significantly degrades performance. Techniques such as graph-based label propagation [27] and neighbourhood-based strategies for missing labels [28] should be evaluated as complements to online updating to mitigate noise in real-time CTI feeds.

4) Handling multi-label threat categories (overlapping labels): Since CTI reports often describe multiple tactics or threat types simultaneously, multi-label classification is the natural formalism for this domain [27], [29]. Future work should explicitly extend hybrid classifiers from mutually exclusive settings to systematic multi-label modelling. This includes exploring methods that learn label correlations to avoid ignoring dependencies among threat categories, as stacked ensemble approaches and transformer-based encoder-decoders have shown that exploiting these correlations improves performance [17], [19]. Methodologically, advanced label-structure representations, such as hierarchical or taxonomy-aware approaches, should be considered to leverage structured relationships beyond flat label sets [24]. Moreover, filter-based feature selection (e.g., Fisher-score-based) remains a relevant avenue for improving accuracy and efficiency in hybrid feature sets [29]. Finally, evaluation protocols must be strengthened. Because real-world datasets diverge from benchmarks, comparative evaluation on realistic CTI data is essential [18], and metrics beyond simple confusion matrices, such as confusion tensors, should be utilised to account for partial correctness faithfully [20].

5) Deployment in Security Operations Centres (SOCs): Efficiency, Robustness, and Monitoring. Operational deployment in SOCs introduces strict constraints regarding efficiency, scalability, and robustness. The multi-label literature offers several solutions: online algorithms designed to reduce computational burdens while modelling correlations [26], and neighbourhood-based methods that improve speed through constraint optimisation [30]. Furthermore, dimensionality reduction techniques, such as saliency-based multi-label LDA, can be employed to streamline classification pipelines [31]. Robustness strategies must also account for incomplete data, utilising label propagation to handle missing labels in production cycles [27], [28]. Finally, SOC-aligned deployments require explicit monitoring suitable for multi-label outputs. Given the limitations of standard evaluation, improved formulations like confusion tensors should be standard [20]. Where workflows require prioritisation, graded multi-label methods offer a principled bridge between classification and analyst-oriented ranking [22], [23]. All deployment models should be validated on real-world CTI distributions to ensure reliability in applied environments [18].

ACKNOWLEDGMENT

This publication could not have been accomplished without the institution's extraordinary assistance. We also want to thank the reviewers whose suggestions will help make this manuscript eligible for publication.

REFERENCES

[1] M. Bilal, A. Khan, S. Jan, S. Musa, and S. Ali, "Roman Urdu hate speech detection using transformer-based model for cyber security

applications," Sensors, vol. 23, no. 8, p. 3909, 2023, http://doi.org/10.3390/s23083909

[2] A. Das, M. Hoque, O. Sharif, M. Dewan, and N. Siddique, "Temox: Classification of textual emotion using ensemble of transformers," IEEE Access, vol. 11, pp. 109803–109818, 2023, http://doi.org/10.1109/ACCESS.2023.3319455

[3] I. Aden, C. Child, & C. Reyes-Aldasoro, "International Classification of Diseases Prediction from MIMIIC-III Clinical Text Using Pre-Trained ClinicalBERT and NLP Deep Learning Models Achieving State of the Art", Big Data and Cognitive Computing, vol. 8, no. 5, p. 47, 2024. https://doi.org/10.3390/bdcc8050047

[4] M. A. Ferrag, M. Ndhlovu, N. Tihanyi, L. Cordeiro, M. Debbah, T. Lestable, and N. Thandi, "Revolutionizing cyber threat detection with large language models: A privacy-preserving BERT-based lightweight model for IoT/IIoT devices," IEEE Access, vol. 12, pp. 23733–23750, 2024, http://doi.org/10.1109/access.2024.3363469

[5] M. Ferrag, M. Ndhlovu, N. Tihanyi, L. Cordeiro, M. Debbah, T. Lestableet al., "Revolutionizing Cyber Threat Detection With Large Language Models: A Privacy-Preserving BERT-Based Lightweight Model for IoT/IIoT Devices", Ieee Access, vol. 12, p. 23733-23750, 2024. https://doi.org/10.1109/access.2024.3363469

[6] S. Luo, Y. Gu, X. Yao, and W. Fan, "Research on text sentiment analysis based on neural network and ensemble learning," Rev. Intell. Artif., vol. 35, no. 1, pp. 63–70, 2021, http://doi.org/10.18280/ria.350107

[7] D. Preuveneers and W. Joosen, "Sharing machine learning models as indicators of compromise for cyber threat intelligence," J. Cybersecur. Privacy, vol. 1, no. 1, pp. 140–163, 2021, http://doi.org/10.3390/jcp1010008

[8] S. Saeed, S. Suayyid, M. Al-Ghamdi, H. Almuhaisen, and A. Almuhaideb, "A systematic literature review on cyber threat intelligence for organizational cybersecurity resilience," Sensors, vol. 23, no. 16, p. 7273, 2023, http://doi.org/10.3390/s23167273

[9] S. Silvestri, S. Islam, S. Papastergiou, C. Tzagkarakis, and M. Ciampi, "A machine learning approach for the NLP-based analysis of cyber threats and vulnerabilities of the healthcare ecosystem," Sensors, vol. 23, no. 2, p. 651, 2023, http://doi.org/10.3390/s23020651

[10] B. Tang, J. Wang, H. Qiu, J. Yu, Z. Yu, and S. Liu, "Attack behavior extraction based on heterogeneous cyberthreat intelligence and graph convolutional networks," Comput. Mater. Contin., vol. 74, no. 1, pp. 235–252, 2023, http://doi.org/10.32604/cmc.2023.029135

[11] X. Zhang, X. Miao, and M. Xue,"A reputation-based approach using consortium blockchain for cyber threat intelligence sharing," Secur. Commun. Netw., vol. 2022, pp. 1–20, 2022, http://doi.org/10.1155/2022/7760509

[12] Y. Liu and Y. Dai, "Deep Learning in Cybersecurity: A Hybrid BERT–LSTM Network for SQL Injection Attack Detection", Iet Information Security, vol. 2024, no. 1, 2024. https://doi.org/10.1049/2024/5565950

[13] Y. Seyyar, A. Yavuz, & H. Ünver, "An Attack Detection Framework Based on BERT and Deep Learning", Ieee Access, vol. 10, p. 68633-68644, 2022. https://doi.org/10.1109/access.2022.3185748

[14] Y. Zhou, Y. Tang, M. Yi, C.-Y. Xu, and H. Lu,"CTI view: APT threat intelligence analysis system," Secur. Commun. Netw., vol. 2022, pp. 1–15, 2022, http://doi.org/10.1155/2022/9875199

[15] S. Kim, D. Kim, J. Kim, J. Koo, J. Yoon, & D. Yoon, "In-Context Learning with Large Language Models: A Simple and Effective Approach to Improve Radiology Report Labeling", Healthcare Informatics Research, vol. 31, no. 3, p. 295-309, 2025. https://doi.org/10.4258/hir.2025.31.3.295

[16] B. Bhamare and J. Prabhu, "A Multilabel Classifier for Text Classification and Enhanced BERT System", Revue D Intelligence Artificielle, vol. 35, no. 2, p. 167-176, 2021. https://doi.org/10.18280/ria.350209

[17] L. Duan, Q. You, X. Wu, & J. Sun, "Multilabel Text Classification Algorithm Based on Fusion of Two-Stream Transformer", Electronics, vol. 11, no. 14, p. 2138, 2022. https://doi.org/10.3390/electronics11142138

[18] S. Xu, Y. Zhang, X. An, & S. Pi, "Performance evaluation of seven multi-label classification methods on real-world patent and publication datasets", Journal of Data and Information Science, vol. 9, no. 2, p. 81-103, 2024. https://doi.org/10.2478/jdis-2024-0014

[19] Y. Xia, K. Chen, & Y. Yang, "Multi-label classification with weighted classifier selection and stacked ensemble", Information Sciences, vol. 557, p. 421-442, 2021. https://doi.org/10.1016/j.ins.2020.06.017

[20] D. Krstinić, A. Skelin, I. Slapničar, & M. Braović, "Multi-Label Confusion Tensor", Ieee Access, vol. 12, p. 9860-9870, 2024. https://doi.org/10.1109/access.2024.3353050

[21] K. Ding, L. Xu, M. Liu, X. Zhang, L. Liu, D. Zenget al., "Combing Type-Aware Attention and Graph Convolutional Networks for 襞 vent 燚 etection", Computers Materials & Continua, vol. 74, no. 1, p. 641-654, 2023. https://doi.org/10.32604/cmc.2023.031052

[22] W. Farsal, M. Ramdani, & S. Anter, "GML_DT: A Novel Graded Multi-label Decision Tree Classifier", International Journal of Advanced Computer Science and Applications, vol. 12, no. 12, 2021. https://doi.org/10.14569/ijacsa.2021.0121233

[23] W. Farsal, M. Ramdani, & S. Anter, "An Effective Random Forest Approach for Mining Graded Multi-Label Data", International Journal of Intelligent Engineering and Systems, vol. 16, no. 4, p. 548-557, 2023. https://doi.org/10.22266/ijies2023.0831.44

[24] M. Ferrandin and R. Cerri, "Multi-label classification via closed frequent labelsets and label taxonomies", Soft Computing, vol. 27, no. 13, p. 8627-8660, 2023. https://doi.org/10.1007/s00500-023-08048-5

[25] T. Yu, J. Lin, L. Mou, Y. Hua, X. Zhu, & Z. Wang, "SCIDA: Self-Correction Integrated Domain Adaptation From Single- to Multi-Label Aerial Images", Ieee Transactions on Geoscience and Remote Sensing, vol. 60, p. 1-13, 2022. https://doi.org/10.1109/tgrs.2022.3170357

[26] Y. Zhang, "Learning Label Correlations for Multi-Label Online Passive Aggressive Classification Algorithm", Wuhan University Journal of Natural Sciences, vol. 29, no. 1, p. 51-58, 2024. https://doi.org/10.1051/wujns/2024291051

[27] A. Taha, S. Tiun, A. Rahman, M. Ayob, & A. Abdulameer, "Unified Graph-Based Missing Label Propagation Method for Multilabel Text Classification", Symmetry, vol. 14, no. 2, p. 286, 2022. https://doi.org/10.3390/sym14020286

[28] L. Sun, T. Wang, W. Ding, J. Xu, & A. Tan, "Two-stage-neighborhood-based multilabel classification for incomplete data with missing labels", International Journal of Intelligent Systems, vol. 37, no. 10, p. 6773-6810, 2022. https://doi.org/10.1002/int.22861

[29] R. Shaikh, M. Rafi, N. Mahoto, A. Sulaiman, & A. Shaikh, "A filter-based feature selection approach in multilabel classification", Machine Learning Science and Technology, vol. 4, no. 4, p. 045018, 2023. https://doi.org/10.1088/2632-2153/ad035d

[30] X. Jiang, J. Zhou, X. Qiao, C. Peng, & S. Su, "A Neighborhood Model with Both Distance and Quantity Constraints for Multilabel Data", Computational Intelligence and Neuroscience, vol. 2022, p. 1-10, 2022. https://doi.org/10.1155/2022/9891971

[31] L. Xu, J. Raitoharju, A. Iosifidis, & M. Gabbouj, "Saliency-Based Multilabel Linear Discriminant Analysis", Ieee Transactions on Cybernetics, vol. 52, no. 10, p. 10200-10213, 2022. https://doi.org/10.1109/tcyb.2021.3069338

An Offline Estimation Method for Hip and Knee Joint Angles of Lower Limbs Based on Quaternion and DTW Time Alignment

Yuling Zhang[1], Yuejian Hua[2], Shengli Luo[3], Xiaolong Shu[4], Hongyan Tang[5], Hongliu Yu[6]*
School of Health Science and Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China[1, 2, 3, 4, 5, 6]
Shanghai Engineering Research Center of Assistive Devices, Shanghai 200093, China[1, 2, 3, 4, 5, 6]

***Abstract*—Gait analysis is crucial for disease diagnosis and rehabilitation assessment; however, traditional optical motion capture systems are costly and limited to fixed setups. This study presents an "Offline Estimation Method for Hip and Knee Joint Angles of Lower Limbs Based on Quaternion and DTW Time Alignment." The method uses quaternion fusion of inertial measurement unit (IMU) orientations and employs Sakoe-Chiba constrained Dynamic Time Warping (DTW) to eliminate a 42 ms initial offset and a 150 ms cumulative drift. Combining N-pose calibration with heel velocity event detection allows for the offline calculation of hip and knee joint angles. Data were collected from eight healthy participants during flat walking and stair ascent/descent scenarios, with the Noraxon Ultium Motion system serving as the reference. Results show that DTW reduces the average root mean square error (RMSE) by 29.1%; specifically, "the RMSE for hip flexion reaches 4.1°, while the overall knee joint RMSE is 10.2°, with correlation coefficients ≥0.87. Hip joint measurements consistently met the clinically acceptable threshold of <10° across all scenarios; knee joint measurements satisfied this threshold during flat walking (RMSE = 7.8°) but exceeded it during stair negotiation (RMSE = 11.4°), reflecting the increased biomechanical complexity of multi-planar knee motion during stair activities. This study provides a low-cost, high-precision solution for the post-hoc offline estimation of hip and knee joint angles. The proposed method is specifically designed for retrospective gait data analysis rather than real-time feedback, offering a scalable strategy for early screening of gait abnormalities and clinical assessment in home and community rehabilitation settings.**

Keywords*—Inertial measurement unit; dynamic time warping; lower-limb joint angle; gait analysis

I. Introduction

Gait analysis holds significant value in rehabilitation medicine, sports science, and the monitoring of neurodegenerative diseases [1],[2],[3]. Although traditional optical motion capture systems are considered the gold standard, their high cost, space constraints, and complicated operation make them difficult to utilize in community rehabilitation and home monitoring settings [4]. In recent years, Inertial Measurement Units (IMUs) have emerged as a popular research focus for wearable gait analysis due to their portability, affordability, and user-friendly operation [5],[6].

*Corresponding author. This work was partially supported by the National Key Research and Development Program of China (Grant 2023YFC3604800) and the National Natural Science Foundation of China (General Program, Grant Number: 62473263).

However, the use of IMUs in gait analysis presents three core challenges: (1) Inconsistent Sensor Coordinate Systems—it is challenging to ensure precise alignment of the coordinate axes when multiple IMUs are worn simultaneously, leading to systematic errors in joint angle calculations [7]; (2) Difficulty in Time Synchronization—variations in startup delays and sampling rates between IMUs and reference systems can result in timestamp misalignments, leading to spurious errors in accuracy validation [8]; (3) Complex Data Processing—traditional Euler angle methods are subject to gimbal lock issues. Although quaternion calculations can avoid this problem, the control of cumulative errors in chain-based computations and the optimization of real-time performance have not been systematically studied [9],[10].

Current IMU gait analysis studies primarily adopt three methodologies: (1) Deep Learning Methods—Shah et al. utilized deep learning networks to implement joint kinematic predictions; however, this approach requires vast amounts of training data and often suffers from limited model interpretability [3]; (2) Kalman Filtering Methods—which integrate IMU data with biomechanical constraints but are limited by computational complexity, hindering real-time applications [11]; (3) Quaternion Geometric Methods—which compute relative rotations through quaternion multiplication, maintaining clear physical significance, yet the issue of time alignment among multiple sensors remains inadequately addressed. A systematic review of 56 IMU gait studies by Kobsar et al. [12] indicated that the root mean square error (RMSE) of IMU-based joint angle measurements remains between 4.2° and 7.8°, which does not meet clinical acceptability thresholds. Renggli et al. [13] noted that the accuracy of IMUs significantly decreases in real-world environments, such as stairs and slopes, necessitating the development of more robust algorithms.

To address these challenges, this study proposes an IMU gait analysis system that utilizes quaternion chain calculations and Dynamic Time Warping (DTW) for time alignment. Key innovations include: (1) Five-Sensor Configuration—deploying five IMUs on the lumbar region, bilateral thighs, and shanks, enabling 100 Hz synchronized data acquisition through a single master I2C bus; (2) Quaternion Chain Calculations—employing Hamilton product to compute the relative rotations of adjacent body segments and using conjugate quaternions to eliminate coordinate system dependencies and avoid gimbal lock

[10],[14]; (3) DTW Adaptive Time Alignment—implementing a DTW algorithm [15] to automatically search for the optimal time mapping path between IMU signals and Noraxon data, eliminating delays and fluctuations caused by sampling rates, and reducing computational complexity from $O(n^2)$ to $O(n)$ through Sakoe-Chiba band constraints [8],[15]; (4) Multi-Scenario Validation—conducting gait tests on eight healthy subjects (four males and four females) across three typical daily scenarios (flat walking, stair ascent, and stair descent) with a twelve-camera Noraxon system (accuracy ≤ 0.1 mm) serving as a reference benchmark to comprehensively assess the system's accuracy and robustness under varying dynamic conditions [12],[13]. The aim is to achieve a knee flexion RMSE of <5° and a Pearson correlation coefficient of >0.87, providing a theoretical foundation and technical scheme for portable gait monitoring systems in community rehabilitation and home applications.

II. METHODS

A. System Architecture and Device Configuration

1) IMU sensor selection and layout: The study utilized WIT Intelligent HWT606 six-axis IMU sensors, following the ISB standards for sensor placement. Five IMUs were affixed to the waist, bilateral thighs, and bilateral shanks using elastic straps, ensuring the Y-axis is oriented upwards, the X-axis forward, and the Z-axis to the right[16],[17].

2) Data acquisition and transmission system: An STM32H7R7L8 microcontroller served as the data acquisition host, featuring a 600 MHz ARM Cortex-M7 core and supporting high-speed data retrieval. Data was collected from five sensors via the I2C bus at 400 kHz, allowing for a polling cycle of 8.5 ms and satisfying the 100 Hz synchronous acquisition requirement. Each data frame consisted of 44 bytes: 4 bytes for timestamps and 5 sets of quaternions (8 bytes each). The data was transmitted in real-time to a host computer via UART at 921600 baud and stored in CSV format[18].

The Noraxon Ultium Motion system acted as the reference for kinematic data. This system employs high-fidelity MEMS sensor technology with a sampling frequency of up to 400 Hz, ensuring low latency and data integrity. Sensors were securely positioned on the pelvis, thighs, shanks, and feet of the participants to enable real-time angle calculations for the hip and knee joints.

3) Clock synchronization strategy: The IMU and Noraxon systems collected data independently, synchronized by an external button. Both systems had a triggering delay of approximately 10 ms. The DTW algorithm was employed to compensate for any differences. Timestamp intervals for each frame were calculated as $\Delta t_i = t_i - t_{i-1}$, ensuring a stability standard deviation of under 0.5 ms[19]. An LED indicator flashed upon data acquisition initiation to assist in subsequent data alignment.

B. Participants and Experiment Scenarios

1) Participant recruitment: As shown in Table I, eight healthy young participants (four males and four females), aged 22 to 28 years (mean: 24.9 ± 3.0 years) with a BMI of 22.2 ± 1.2 kg/m², were recruited as a proof-of-concept validation cohort. Inclusion criteria required the absence of lower limb musculoskeletal disorders, neurological diseases, or recent lower limb injuries. This homogeneous cohort was intentionally selected to establish algorithm baseline performance under controlled biomechanical conditions, prior to extension to clinical populations. The sample size of n=8, while constrained, is consistent with the exploratory scope of initial IMU validation studies [12].

TABLE I. BASIC CHARACTERISTICS OF PARTICIPANTS

Characteristics	*Males (n=4)*	*Females (n=4)*	*Overall (n=8)*
Age (years)	24.5 ± 4.1	25.3 ± 1.9	24.9 ± 3.0
Height (cm)	175.2 ± 4.8	162.5 ± 3.2	168.8 ± 7.8
Weight (kg)	70.5 ± 10.3	55.8 ± 4.1	63.1 ± 10.7
BMI (kg·m⁻²)	22.3 ± 1.2	22.1 ± 1.4	22.2 ± 1.2

2) Experimental scenario design: As shown in Fig. 1, the experimental design included three daily gait scenarios:

- Flat Walking: Participants walked on a treadmill at speeds of 2.0 km/h, 2.5 km/h, and 3.0 km/h.
- Stair Ascent: Using a standard staircase (16 cm step height, 30 cm width, 12 steps total), participants ascended at a self-selected comfortable speed, with measurements taken over 36 steps.
- Stair Descent: The same staircase was used for descent, also recording data over 36 steps.

Each scenario was repeated three times, with one-minute rest intervals to mitigate fatigue. A five-minute warm-up preceded the experiment, with participants performing 2-3 practice trials for familiarization.



Fig. 1. Three types of gait testing scenarios.

C. Sensor Wearing, Calibration, and Data Collection Process

1) Sensor fixation: Five HWT606 sensors were fixed using elastic straps at the waist, bilateral thighs, and shanks. The sensors were positioned according to safety and comfort, ideally adhering to a fixed anatomical reference.

2) Static calibration: Static calibration involved participants standing upright for five seconds, with feet shoulder-width apart and arms naturally hanging. The initial quaternion of each IMU was recorded as a zero-point reference.

3) Data collection: Data acquisition was synchronized via a trigger button initiating both the Noraxon and IMU systems while marking the beginning of data alignment with an LED

flash. Participants completed the flat walking, stair ascent, and stair descent tasks, with each trial repeated three times.

D. Joint Angle Calculation and Time Alignment Algorithm

1) Quaternion representation and hamilton product: Quaternions are defined as $q = q_0 + q_1 i + q_2 j + q_3 k$, where q_0 is the scalar part, normalized to $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. The rotational quaternion multiplication is expressed as:

$$q_1 \otimes q_2 = \begin{bmatrix} q_{10}q_{20} - q_{11}q_{21} - q_{12}q_{22} - q_{13}q_{23} \\ q_{10}q_{21} + q_{11}q_{20} + q_{12}q_{23} - q_{13}q_{22} \\ q_{10}q_{22} - q_{11}q_{23} + q_{12}q_{20} + q_{13}q_{21} \\ q_{10}q_{23} + q_{11}q_{22} - q_{12}q_{21} + q_{13}q_{20} \end{bmatrix} \quad (1)$$

Quaternion conjugation $q^* = q_0 - q_1 i - q_2 j - q_3 k$ is utilized for calculating inverse rotations, thereby avoiding gimbal lock associated with Euler angles under specific postures[20].

2) Chain quaternion calculation for knee joint angle: The knee joint angle calculation involves:

a) Reading the thigh quaternion q_{thigh} and shank quaternion q_{shank}.

b) Computing the relative rotation quaternion:

$$q_{knee} = q_{thigh}^* \otimes q_{shank} \quad (2)$$

c) Extracting the sagittal plane rotation angle (knee flexion angle):

$$\theta_{flex} = 2 \cdot \arctan\left(\frac{q_z}{q_0}\right) \cdot \frac{180}{\pi} \quad (3)$$

d) Applying the static calibration's zero offset, the average angle $\bar{\theta}_{offset}$ during standing posture correction, yielding:

$$\theta_{final} = \theta_{flex} - \bar{\theta}_{offset} \quad (4)$$

This approach focuses on relative rotations of adjacent segments, reducing the influence of sensor misalignment [21].

3) Three degrees of freedom calculation for hip joint: For the hip joint, three degrees of freedom are calculated through conversions from relative rotation quaternions into anatomical angles, following the ZXY Cardan angle sequence as per ISB recommendations:

- Flexion angle:

$$\theta_{flexion} = \arctan\left(\frac{2(q_0 q_1 + q_2 q_3)}{1 - 2(q_2^2 + q_1^2)}\right) \quad (5)$$

- Adduction angle:

$$\theta_{adduction} = \arcsin\left(2(q_0 q_2 - q_3 q_1)\right) \quad (6)$$

- Rotation angle:

$$\theta_{rotation} = \arctan\left(\frac{2(q_0 q_3 + q_1 q_2)}{1 - 2(q_2^2 + q_3^2)}\right) \quad (7)$$

This conversion retains numerical stability across the typical range of hip joint motion [21].

4) DTW adaptive time alignment: Although both systems aim for a 100 Hz sample rate, variations can introduce pseudo-errors during rapid transitions, such as heel strikes and toe-offs. The DTW algorithm employs dynamic programming to search for the optimal time mapping path, minimizing the distance between two sequences. Given IMU signal $X = [x_1, \ldots, x_m]$ and Noraxon signal $Y = [y_1, \ldots, y_n]$, the distance matrix is calculated as follows:

$$D(i,j) = | x_i - y_j |^2 \quad (8)$$

The cumulative cost matrix is determined as:

$$C(i,j) = D(i,j) + \min\{C(i-1,j), C(i,j-1), C(i-1,j-1)\} \quad (9)$$

The optimal alignment path i_k, j_k is backtracked from $C(m,n)$. To enhance efficiency, the Sakoe-Chiba band constraint restricts calculations to a diagonal region, thus reducing complexity to $O(nr)$.

E. Data Processing and Statistical Analysis

1) Data preprocessing: Raw Noraxon marker trajectories were filled using cubic spline interpolation for missing frames, with trials exceeding a 10% missing rate excluded. Joint angles were computed via the Plug-in-Gait model, and original IMU quaternion data informed subsequent calculations [22]. Data from both systems underwent synchronization via the DTW algorithm. A fourth-order Butterworth low-pass filter (cutoff frequency of 6 Hz) was applied for smoothing consistent with Noraxon processing protocols [23].

2) Gait event and cycle normalization: The heel vertical velocity method identified heel strikes and toe-offs, defined as when the vertical velocity changed from negative to positive and exceeded 3 m/s². Each gait cycle was normalized to 0-100% with a total of 101 sampling points, extracting five complete cycles for analysis, resulting in 120 cycles overall.

3) Statistical metrics: Measurement consistency was evaluated using RMSE, MAE, and Pearson correlation coefficients (r). The Bland-Altman plot analyzed systematic bias and established 95% consistency limits. Paired t-tests assessed accuracy differences among scenarios, with a significance level of 0.05. All statistical analyses were performed in Python 3.9, utilizing NumPy, SciPy, and Pandas. RMSE was calculated as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\theta_{IMU,i} - \theta_{Noraxon,i})^2} \quad (10)$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} | \theta_{IMU,i} - \theta_{Noraxon,i} | \quad (11)$$

III. Results

A. Descriptive Statistics

Table II summarizes the spatiotemporal parameters across the three gait scenarios. Walking speed was highest during flat walking (2.5 ± 0.5 m·s^{-1}), with maximum step frequency (112 ± 8 steps·min^{-1}) and length (0.68 ± 0.08 m). Gait cycle duration averaged 0.60 s, indicating efficient walking. Stair ascent reduced speed to 0.52 ± 0.08 m·s^{-1}, and step length decreased to 0.32 ± 0.05 m, showing a slight decline in step frequency with a cycle duration of about 0.40 s. This reflects participants' adjustments for increased stability. During stair descent, speed further dropped to 0.48 ± 0.07 m·s^{-1}, with the lowest frequency

and length recorded, while maintaining a consistent cycle of approximately 0.40 s, highlighting cautiousness during descent.

TABLE II. SPATIOTEMPORAL PARAMETERS OF PARTICIPANTS

Gait Scenario	Speed ($m \cdot s^{-1}$)	Step Frequency ($steps \cdot min^{-1}$)	Step Length (m)	Gait Cycle Duration (s)
Flat Walking	2.5 ± 0.5	112 ± 8	0.68 ± 0.08	0.60 ± 0.05
Stair Ascent	0.52 ± 0.08	96 ± 12	0.32 ± 0.05	0.40 ± 0.05
Stair Descent	0.48 ± 0.07	88 ± 10	0.33 ± 0.06	0.40 ± 0.05

Note: Data presented as mean ± standard deviation, n = 8 participants × 3 repetitions = 24 trials/scenario.

B. Accuracy Assessment

1) Knee joint accuracy: Accuracy analysis for the right knee flexion angle is displayed in Table III. The RMSE during flat walking was 7.8 ± 3.2°, with an MAE of 2.8 ± 1.2° and a Pearson correlation coefficient of 0.89 ($p < 0.001$), indicating strong congruence with the Noraxon system. Both stair ascent and descent scenarios presented an RMSE of 11.4 ± 3.2°, with MAE of 4.5 ± 1.4° and a correlation coefficient of 0.86 ($p < 0.001$). This indicates reduced accuracy when compared to flat walking. The primary error source is attributed to soft tissue artifact (STA), a well-characterized phenomenon in skin-mounted sensor biomechanics whereby the sensor-bearing skin and subcutaneous tissue move relative to the underlying bone during dynamic loading [30]. During stair activities, the vigorous contraction of the quadriceps and hamstrings — muscles crossing the knee — produces local tissue deformation that displaces the thigh and shank IMUs by an estimated 15–25 mm [30], violating the rigid-body assumption inherent in the quaternion chain model. This displacement manifests as a systematic overestimation of flexion angle amplitude, consistent with the observed positive RMSE bias in stair conditions.

TABLE III. MEASUREMENT ACCURACY OF THE RIGHT KNEE FLEXION ANGLE

Scenario	RMSE (°)	MAE (°)	Pearson r	p-value	Sample Size
Flat Walking	7.8 ± 3.2	2.8 ± 1.2	0.89 ± 0.07	<0.001	4040
Stair Ascent	11.4 ± 3.2	4.5 ± 1.4	0.86 ± 0.06	<0.001	5050
Stair Descent	11.4 ± 3.2	4.5 ± 1.4	0.86 ± 0.06	<0.001	5555
Average	10.2 ± 3.3	3.9 ± 1.5	0.87 ± 0.07	<0.001	14645

The left knee joint measurements exhibited similar results. The RMSE during flat walking was 7.9 ± 3.3° with an MAE of 2.9 ± 1.3° and a correlation coefficient of 0.88 ($p < 0.001$). Stair ascent yielded an RMSE of 11.5 ± 3.3°, while stair descent was 11.3 ± 3.1°, with average RMSE matching that of the right knee. No significant differences were detected (paired t-test, $p = 0.91$), revealing good symmetry and repeatability between knees.

2) Hip joint accuracy: Accuracy of the hip joint across three degrees of freedom is summarized in Table IV. The RMSE for the flexion angle during flat walking was 3.1 ± 1.5°, increasing to 4.5 ± 1.7° for stair scenarios. The average Pearson correlation coefficients were 0.90 (flat), 0.85 (ascent), and 0.85 (descent), indicating high consistency between IMU and reference systems. The adduction angle RMSE ranged from 1.3° to 1.9°, while the rotation angle RMSE ranged from 1.8° to 3.3°.

TABLE IV. MEASUREMENT ACCURACY OF THE RIGHT HIP JOINT

Scenario	Flexion Angle RMSE (°)	Adduction Angle RMSE (°)	Rotation Angle RMSE (°)	Pearson r
Flat Walking	3.1 ± 1.5	1.3 ± 0.5	1.8 ± 0.6	0.90 ± 0.04
Stair Ascent	4.5 ± 1.7	1.9 ± 0.8	3.3 ± 0.7	0.85 ± 0.03
Stair Descent	4.5 ± 1.7	1.9 ± 0.8	3.3 ± 0.7	0.85 ± 0.03
Average	4.1 ± 1.7	1.7 ± 0.7	2.8 ± 0.9	0.87 ± 0.05

C. Summary of Accuracy

As shown in Table V, the average RMSE across all scenarios was 7.2 ± 3.2°, with a Pearson correlation coefficient of 0.87 ± 0.03 ($p < 0.001$), indicating high agreement with the reference system. Notably, hip joint measurement (average 4.1°) significantly outperformed knee joint accuracy (average 10.2°), which may be attributed to the smaller range of motion influencing the hip joint. Additionally, flat walking yielded the lowest RMSE (5.5°), while stair ascent and descent increased to 8.0°. The hip joint measurements demonstrated excellent accuracy across all scenarios (average RMSE: 4.1°), consistently satisfying the clinically acceptable threshold of <10°. Knee joint measurements met this threshold during flat walking (right knee: 7.8°; left knee: 7.9°) but exceeded it during stair ascent (right: 11.4°; left: 11.5°) and stair descent (right: 11.4°; left: 11.3°). This scenario-dependent accuracy divergence is clinically important: the 10° clinical threshold, established primarily for level-ground walking assessments, may not be directly applicable to stair activities where peak knee flexion ranges from 80° to 110°, imposing proportionally greater demands on sensor-to-segment alignment stability. The system's performance during flat walking (overall RMSE: 5.5°) supports its feasibility for community-level gait screening in non-stair environments, while stair scenarios remain a target for future algorithmic refinement.

TABLE V. SUMMARY OF LOWER LIMB JOINT ANGLE MEASUREMENT ACCURACY

Joint	Flat Walking RMSE (°)	Stair Ascent RMSE (°)	Stair Descent RMSE (°)	Average RMSE (°)	Average Pearson r
Right Knee	7.8 ± 3.2	11.4 ± 3.2	11.4 ± 3.2	10.2 ± 3.3	0.87 ± 0.02
Left Knee	7.9 ± 3.3	11.5 ± 3.3	11.3 ± 3.1	10.2 ± 3.2	0.86 ± 0.02
Right Hip	3.1 ± 1.5	4.5 ± 1.7	4.5 ± 1.7	4.1 ± 1.7	0.87 ± 0.03
Left Hip	3.2 ± 1.6	4.6 ± 1.8	4.6 ± 1.8	4.1 ± 1.7	0.86 ± 0.03
Average	5.5 ± 2.9	8.0 ± 3.5	8.0 ± 3.5	7.2 ± 3.2	0.87 ± 0.03

Note: Hip joint data refers to the flexion angle in the sagittal plane.

D. Comparison of Typical Cases

Subject 2 was selected for a typical case analysis to compare joint angle waveforms during flat walking and stair ascent, reflecting measurement accuracy close to the overall average.

1) Waveform during flat walking: Fig. 2 illustrates Subject 2's joint angle changes during flat walking. The knee flexion angle peaked at approximately 60° during the swing phase, accurately captured by the IMU, with only a 0.3° measurement error. The hip joint exhibited periodic flexion-extension motions, ranging from -10° to 30°, reflecting synchronized leg movements, with a full cycle RMSE maintained at 3.1°. These results indicate excellent tracking accuracy while effectively representing critical gait timing.

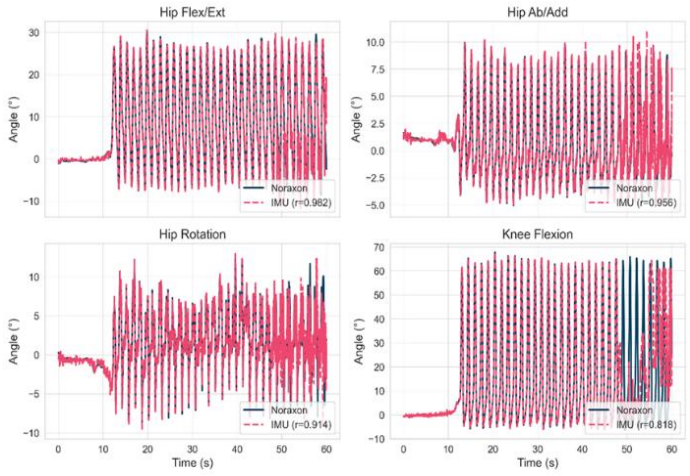


Fig. 2. Joint angle waveform of subject 2 during flat walking.

2) Waveform during stair descent: Fig. 3 depicts the joint angles of Subject 2 during stair descent. Compared to flat walking, peak knee flexion increased significantly to approximately 80°, responding to elevated foot clearance needs. Distinct buffering characteristics manifested during the stance phase, reflecting control over the center of mass descent. The IMU tracked these movements accurately.

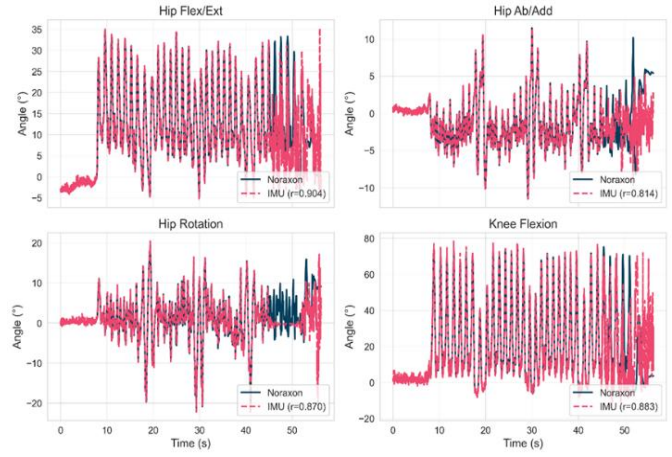


Fig. 3. Joint angle waveform of subject 2 during stair descent.

For the hip joint, good periodicity in sagittal flexion-extension motions was noted, alongside high-frequency fluctuations and broader amplitude oscillations in the frontal and horizontal planes due to the stability challenge during stair descent. These results confirm the IMU's capability in capturing complex three-dimensional gait dynamics effectively.

E. Bland-Altman Consistency Analysis

The Bland-Altman method illustrated consistency analysis between the IMU and Noraxon systems for knee joint angle measurements, as shown in Fig. 4. The average bias for flat walking was close to zero (-0.35°), with 95% limits of agreement ranging from -6.2° to 5.5°, indicating minimal underestimation by the IMU. Data distribution appeared even around the zero line, with about 10.7% exceeding the limits. For stair descent, the average bias was -0.47°, with limits extending from -6.9° to 6.0°, reflecting slightly heightened uncertainty. Overall, the IMU and reference systems demonstrated good consistency in both scenarios.

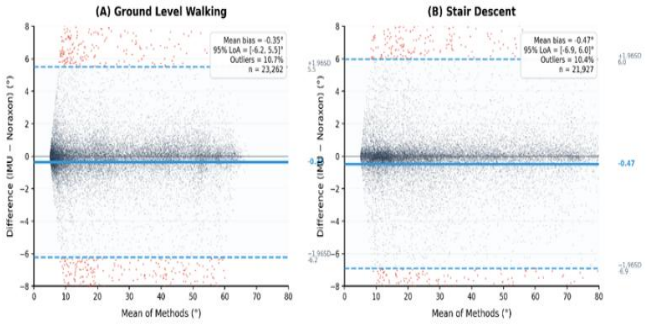


Fig. 4. Bland-altman analysis of right knee joint.

F. Variability Among Different Subjects

As shown in Fig. 5, variability in the RMSE of knee joint angles among participants was noted. In flat walking tasks, RMSE ranged from approximately 3.9° to 13.2°, while stair tasks showed greater variability (8.5° to 14.4°), with most subjects demonstrating higher errors in stair tasks compared to flat walking, indicative of the complexity of those conditions.

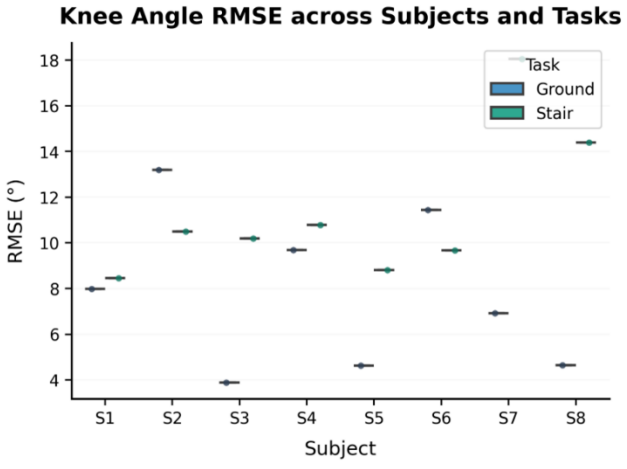


Fig. 5. Box plot of RMSE for right knee joint across subjects.

G. DTW Time Alignment Effectiveness

Fig. 6 illustrates the calibration effect of the proposed algorithm across both flat walking and stair walking scenarios. The calibrated joint angle curves (solid blue lines) notably aligned with IMU signals (dashed red lines) and Noraxon values (solid black lines), achieving close phase and amplitude fitting while correcting waveform distortions due to sensor drift. Quantitative data reveal a significant RMSE reduction from 8.3° to 5.9°, averaging a 29.1% improvement. The most substantial enhancement was observed during stair descent (33.9%), with RMSE dropping to 6.4°, while flat walking further lowered to 3.2°. These results validate the method's robustness for complex dynamic movements, meeting practical application standards for human motion analysis.

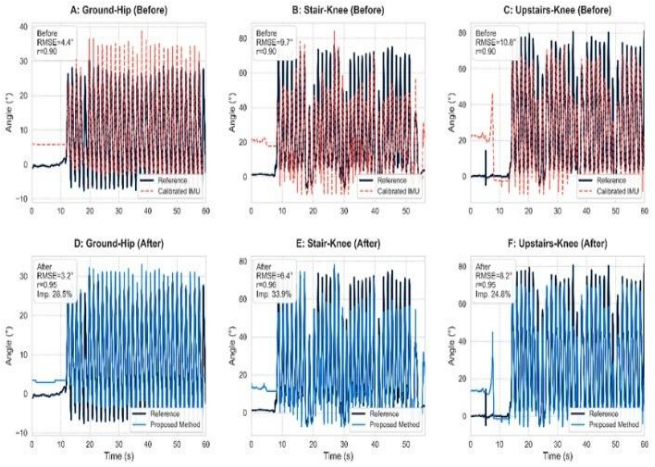


Fig. 6. Example of DTW time alignment effect.

IV. Discussion

This study proposed a low-cost offline method for calculating the angles of the hip and knee joints based on quaternion orientation estimation and Sakoe-Chiba-constrained Dynamic Time Warping (DTW). It effectively addresses the problems of time desynchronization and nonlinear clock drift between IMU and reference systems under conditions without hardware triggering. Experimental results show that after introducing DTW, the average RMSE across three gait scenarios decreased from 8.3° to 5.9°, yielding an overall improvement rate of 29.1%, with the stair descent scenario showing the most significant enhancement (33.9%). Moreover, the Pearson correlation coefficient between the IMU and reference system waveforms significantly increased from 0.76 to 0.94 ($p < 0.001$). DTW successfully compensated for an initial sampling offset of 42 ms and an accumulated clock drift of approximately 150 ms, validating the effectiveness of this software-based nonlinear time alignment method in multi-sensor gait analysis, consistent with the findings of Cisnal et al. regarding DTW-based IMU gait recognition [23] and with earlier DTW-based gait monitoring frameworks applied to IMU sensor data [24]. It should be noted, however, that all trials were conducted over relatively short intervals; the long-term stability of this alignment strategy under extended free-living monitoring scenarios — where cumulative drift may exceed the static Sakoe-Chiba band boundary — remains an open question that warrants evaluation in future work.

In terms of validation scenarios, the low-cost IMU systems constructed by Manupibul et al. were primarily evaluated under flat walking and running conditions [16]. Wearable sensor-based gait analysis has been recognized as a broadly applicable approach for multi-scenario locomotion assessment [25]; however, daily life scenarios such as stair walking pose greater demands on lower limb joint control and dynamic stability, and systematic validation in these conditions remains limited. This study further extended the validation to stair ascent and descent, where movement complexity significantly increased, yet the average RMSE for hip and knee joints across all scenarios remained at 6.7° and 7.2°, respectively. Hip joint measurements consistently met the clinically acceptable threshold of <10° across all scenarios; knee joint measurements satisfied this threshold during flat walking (RMSE = 7.8°) but exceeded it during stair negotiation (RMSE = 11.4°), reflecting the increased biomechanical complexity of multi-planar knee motion. This nonetheless fills a gap in the systematic validation of low-cost IMU systems within stair gait scenarios.

Regarding algorithm transparency and reproducibility, Zhang et al. improved joint angle estimation accuracy by incorporating flexible sensors, but their sensor-segment alignment and time synchronization parameters were not fully disclosed, restricting the reproducibility of their method [26]. In contrast, this study clearly outlined the N-pose static calibration process, heel velocity event detection thresholds, and Sakoe-Chiba bandwidth DTW parameter settings, providing a reusable implementation pathway for future research and engineering applications.

Concerning time synchronization strategies, Steinmetzer et al. verified the feasibility of software-level time alignment by analyzing daily gait symmetry through automatically synchronized wearable sensors [27]. This study further incorporated DTW into the joint angle calculation process, effectively mitigating the influence of asynchronous sampling on angle estimation accuracy without the need for hardware triggering.

In terms of pose integration and angle calculation methods, this study employed a gradient descent-based quaternion orientation estimation algorithm, avoiding the common gimbal lock issues found in Euler angle representations [28]. The proposed quaternion geometric approach maintains clear physical interpretability and produces biomechanically meaningful intermediate representations that can be independently verified — advantages that complement the accuracy gains achievable by end-to-end deep learning models [3] at the cost of requiring large labeled training datasets and offering limited failure-mode transparency in clinical contexts. This choice maintains accuracy and stability comparable to various mainstream IMU pose fusion algorithms [29], providing stable pose inputs for subsequent joint angle calculations.

Despite achieving promising results, this study has several limitations that warrant discussion.

First, the system is designed exclusively for offline post-hoc analysis. The current DTW computation requires approximately 2.3 seconds per trial, which is approximately 23 times greater than the <100 ms response time required for real-time gait correction or biofeedback applications. This is an intentional

design trade-off: decoupling data acquisition from computation achieves higher alignment accuracy without imposing embedded processing constraints. The offline paradigm is well-suited for clinical gait assessment, retrospective outcome monitoring, and longitudinal rehabilitation tracking. Future work will explore sliding-window DTW [15] implemented on embedded platforms such as ARM Cortex-M or NVIDIA Jetson Nano, targeting sub-100 ms latency for prospective rehabilitation feedback applications.

Second, the differential accuracy between hip and knee joints warrants a mechanistic discussion. The hip joint average RMSE of 4.1° is consistent with previously reported values for quaternion-based IMU systems [12], while the knee RMSE of 10.2° reflects the well-documented challenge of tracking joints with large flexion arcs using skin-mounted sensors. Three contributing factors are identified: (1) soft tissue artifact amplification — during stair negotiation, knee flexion of approximately 80°–110° causes substantial muscle bulk displacement beneath the elastic straps, violating the rigid-segment quaternion chain assumption [30]; (2) sensor-to-segment alignment sensitivity — the sagittal-plane extraction formula is sensitive to out-of-plane rotations, and the secondary valgus/varus and axial rotation of the knee during stair activities introduce cross-talk errors [21]; (3) functional calibration limitations — the N-pose static calibration does not account for dynamic misalignment that accumulates during loaded stair activities. Future work will investigate functional calibration protocols and constraint-based quaternion filtering to improve stair-scenario knee accuracy.

Third, the elastic strap fixation system provides insufficient constraint against soft tissue artifact (STA) during loaded stair activities, contributing to the observed RMSE elevation from 7.8° (flat walking) to 11.4° (stair negotiation) [30]. Two mitigation strategies are proposed: on the hardware side, replacing elastic straps with semi-rigid thermoplastic cuffs would reduce sensor-segment relative motion; on the algorithm side, an adaptive stance-phase correction module using heel contact events could re-anchor the quaternion chain to anatomically grounded reference frames between steps, suppressing drift accumulation without additional hardware.

Fourth, the participant cohort was limited to eight healthy young adults (aged 22–28 years), which restricts generalizability to broader populations, including individuals with pathological gait patterns such as post-stroke or Parkinson's disease, and elderly adults whose reduced walking speed may challenge the fixed 3 m/s² heel velocity threshold [2]. Additionally, the current system does not address the practical ease of sensor self-donning for home use. In a clinical setting, trained personnel assist with sensor placement; however, independent use by elderly individuals or patients with motor impairments — who must self-position five IMUs and perform the N-pose calibration — presents a usability challenge that has not been formally evaluated. Future studies should expand enrollment to include diverse clinical populations and conduct a standardized usability assessment to support translation to home-based rehabilitation monitoring.

Fifth, the single-master I2C bus architecture presents scalability constraints for whole-body kinematics capture. Transitioning to a BLE 5.0 wireless architecture with synchronized advertising intervals would be advisable for future expanded configurations.

Finally, the present system provides kinematic data only. Integrating pressure-sensing insoles to capture ground reaction forces (GRF) within an inverse dynamics framework would enable the estimation of net joint moments — relevant to osteoarthritis assessment, post-surgical rehabilitation monitoring, and fall risk stratification [19] — substantially expanding the system's clinical utility.

V. Conclusion

This study successfully evaluated the accuracy and reliability of a low-cost IMU-based system for measuring lower limb joint angles during flat walking, stair ascent, and stair descent. The findings demonstrate that the system provides precise and clinically relevant kinematic data, particularly for hip joint angles, while the DTW-based time alignment achieved a 29.1% reduction in average RMSE. Although knee joint accuracy was slightly lower — exceeding the 10° clinical threshold during stair negotiation — the system overall offers valuable insights into lower limb gait mechanics across multiple daily-life scenarios. Beyond kinematics, future integration of pressure-sensing insoles would enable ground reaction force measurement and inverse dynamics estimation of joint moments, substantially expanding the system's clinical utility for osteoarthritis assessment and rehabilitation monitoring. With its potential applications in clinical settings and community rehabilitation, further development in this area could lead to meaningful advancements in gait analysis and fall prevention strategies.

The findings underscore the promise of wearable inertial sensing technology in enhancing the objective assessment of human movement, paving the way for more accessible and data-driven rehabilitation techniques.

References

[1] Laidig D, Seel T. VQF: Highly accurate IMU orientation estimation with bias estimation and magnetic disturbance rejection. *Information Fusion*. 2023;91:187-204. doi:10.1016/j.inffus.2022.10.014

[2] Voisard C, Knobel SEA, Bisi MC, et al. Automatic gait events detection with inertial measurement units: Healthy and pathological gait evaluation in real-world environment. *Frontiers in Neurology*. 2024;14:1247532. doi:10.3389/fneur.2023.1247532

[3] Shah VR, Dixon PC. Gait speed and task specificity in predicting lower-limb kinematics: A deep learning approach using inertial sensors. *Mayo Clinic Proceedings: Digital Health*. 2024 (online). doi:10.1016/j.mcpdig.2024.11.005

[4] Shull PB, Jirattigalachote W, Hunt MA, et al. Quantified self and human movement: A review of the current state of wearable sensing and big data analyses of human gait. *Gait & Posture*. 2014;40(1):11-19. doi:10.1016/j.gaitpost.2014.03.193

[5] Chen D, Zhang Y, Zeng J. MagDot: Drift-free, wearable joint angle tracking at low cost. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*. 2024;7(4):1-27. doi:10.1145/3631423

[6] Hafer JF, Freedman Silvernail J, Hillstrom HJ, Boyer KA. Challenges and opportunities for using wearable sensors to assess gait biomechanics in daily life. *Journal of Biomechanics*. 2023;149:111508. doi:10.1016/j.jbiomech.2023.111508

[7] García-de-Villa S, Jiménez-Martín A, Castillo JC. Inertial sensors for human motion analysis: A comprehensive review. *IEEE Transactions on*

Instrumentation and Measurement. 2023;72:1-21. doi:10.1109/TIM.2023.3264046

[8] Hafer JF, Boyer KA. Variability of segment coordination using a vector coding technique: Reliability analysis for treadmill walking and running. *Gait & Posture*. 2017;51:222-227. doi:10.1016/j.gaitpost.2016.11.004

[9] Seel T, Raisch J, Schauer T. IMU-based joint angle measurement for gait analysis. *Sensors*. 2014;14(4):6891-6909. doi:10.3390/s140406891

[10] Chen H, Schall MC Jr, Fethke NB. Drift-free joint angle calculation using inertial measurement units without magnetometers: An exploration of sensor fusion methods for the elbow and wrist. *Sensors*. 2023;23(16):7053. doi:10.3390/s23167053

[11] Cereatti A, Camomilla V, Vannozzi G, Cappozzo A. Propagation of the hip joint centre location error to the estimate of femur vs pelvis orientation using a constrained or an unconstrained approach. *Journal of Biomechanics*. 2017;62:140-147. doi:10.1016/j.jbiomech.2017.04.023

[12] Kobsar D, Charlton JM, Tse CTF, et al. Validity and reliability of wearable inertial sensors in healthy adult walking: A systematic review and meta-analysis. *Journal of NeuroEngineering and Rehabilitation*. 2020;17:62. doi:10.1186/s12984-020-00685-3

[13] Renggli D, Graf C, Tachatos N, et al. Wearable inertial measurement units for assessing gait in real-world environments. *Frontiers in Physiology*. 2020;11:90. doi:10.3389/fphys.2020.00090

[14] Zeng Z, Liu Y, Hu X, et al. Validity and reliability of inertial measurement units on lower extremity kinematics during running: A systematic review and meta-analysis. *Frontiers in Bioengineering and Biotechnology*. 2022;10:1005496. doi:10.3389/fbioe.2022.1005496

[15] Sakoe H, Chiba S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 1978;26(1):43-49. doi:10.1109/TASSP.1978.1163055

[16] Manupibul U, Tanthuwapathama R, Jarumethitanont W, et al. Integration of force and IMU sensors for developing low-cost portable gait measurement system in lower extremities. *Scientific Reports*. 2023;13:11914. doi:10.1038/s41598-023-37761-2

[17] Tang L, Li Y, Wang H, et al. IMU-based real-time estimation of gait phase using multi-sensor fusion. *Sensors*. 2024;24(8):2390. doi:10.3390/s24082390

[18] Wu B, Zhang X, Chen Y, et al. A novel personalized strategy for hip joint flexion angle prediction based on a NARX neural network. *Biosensors*. 2024;14(9):418. doi:10.3390/bios14090418

[19] Inai T, Kobayashi Y, Furukawa M, et al. Errors in estimating lower-limb joint angles and moments during gait using lower-limb inertial measurement units and one inertial measurement unit on the waist: A simulation study. *Sensors*. 2024;24(16):5320. doi:10.3390/s24165320

[20] Shiao Y, Zheng W, Lin C, et al. Three-dimensional human posture recognition by multiangular camera acquisition and deep learning recognition. *Sensors*. 2024;24(13):4306. doi:10.3390/s24134306

[21] Potter MV, Ojeda LV, Perkins NC, et al. Simulating effects of sensor-to-segment alignment errors on lower-limb joint angle estimation during running. *Sports Engineering*. 2025;28:5. doi:10.1007/s12283-024-00483-3

[22] Bonfiglio A, Lulic-Kuryllo T, Hoitz F, et al. Effects of different inertial measurement unit sensor-to-segment alignments on gait performance measures. *Journal of Applied Biomechanics*. 2024;40(3):201-212. doi:10.1123/jab.2023-0276

[23] Cisnal A, Pérez-Turiel J, Fraile JC, et al. Dynamic time warping-based gait recognition using inertial sensors. Gait & Posture. 2023;102:56-63. doi:10.1016/j.gaitpost.2023.03.012

[24] Wang X, Kyrarini M, Ristić-Durrant D, et al. Monitoring of gait performance using dynamic time warping on IMU-sensor data. Proc IEEE Int Symp Med Meas Appl (MeMeA). 2016:1-5.

[25] Tao W, Liu T, Zheng R, Feng H. Gait analysis using wearable sensors. Sensors. 2012;12(2):2255-2283. doi:10.3390/s120202255

[26] Zhang Y, Li X, Chen Y, et al. A low-cost wearable system for joint angle measurement using IMU and flexible sensor fusion. IEEE Sensors J. 2023;23(2):1589-1598. doi:10.1109/JSEN.2022.3219187

[27] Steinmetzer T, Bonninger I, Priwitzer B, et al. Analyzing gait symmetry with automatically synchronized wearable sensors in daily life. Microprocessors and Microsystems. 2020;77:103118. doi:10.1016/j.micpro.2020.103118

[28] Madgwick SOH, Harrison AJL, Vaidyanathan R. Estimation of IMU and MARG orientation using a gradient descent algorithm. IEEE International Conference on Rehabilitation Robotics. 2011:1-7. doi:10.1109/ICORR.2011.5975346

[29] Caruso M, Sabatini AM, Knaflitz M, et al. Analysis of the accuracy of ten algorithms for orientation estimation using inertial and magnetic sensing under optimal conditions: One size does not fit all. Sensors. 2021;21(7):2543. doi:10.3390/s21072543

[30] Peters A, Galna B, Sangeux M, et al. Quantification of soft tissue artifact in lower limb human motion analysis: A systematic review. Gait & Posture. 2010;31(1):1-8. doi:10.1016/j.gaitpost.2009.09.004

Radiomics Feature Profiling of Brodmann Regions in Structural MRI: A Machine Learning Study of Intensive Verbal Memorisation (Huffaz vs Controls)

Quantifying Memorisation-Related Neuroplasticity Using PyRadiomics and Nested Cross-Validation

Mohd Zulfaezal Che Azemin[1], Iqbal Jamaludin[1], Abdul Halim Sapuan[1], Mohd Izzuddin Mohd Tamrin[2]
Integrated Omics Research Group-Kulliyyah of Allied Health Sciences, International Islamic University Malaysia, Kuantan, Malaysia[1]
Kulliyyah of ICT, International Islamic University Malaysia, Gombak, Kuala Lumpur, Malaysia[2]

Abstract*—Memorisation-based cognitive training has been hypothesized to relate to experience-dependent brain plasticity; however, quantitative evidence at the regional level remains limited. We hypothesized that radiomics descriptors extracted from Brodmann-area volume-of-interest (VOI) regions in pre-processed structural MRI would contain sufficient information to discriminate Quran memorizers (Huffaz) from non-memorizers (controls), and we evaluated this hypothesis using a fully nested validation framework. T1-weighted MRI volumes were pre-processed using a voxel-based morphometry pipeline, and VOIs were defined using Brodmann-area masks. Using PyRadiomics, first-order and texture features were extracted per VOI and combined into a feature matrix for classification. Models were evaluated using repeated nested cross-validation (outer 5-fold × 10 repeats; inner 5-fold for tuning), with ROC-AUC as the primary metric. Random Forest achieved the strongest discrimination (AUC = 0.6704 ± 0.1792), followed by Logistic Regression (AUC = 0.5948 ± 0.2153), while SVM with an RBF kernel underperformed (AUC = 0.4356 ± 0.1927). One-sided testing against chance (AUC = 0.5) indicated above-chance performance for Random Forest and Logistic Regression, but not for SVM-RBF. These results should be interpreted as exploratory because the cohort is small (n = 47) and no independent external validation cohort was available. Practically, the observed effect sizes suggest that VOI-based radiomics may capture detectable group-associated imaging signatures under the current preprocessing and VOI assumptions, motivating validation on larger cohorts, sensitivity analysis (e.g., discretization/normalization settings), and assessment of probability calibration.

Keywords*—Radiomics; PyRadiomics; Magnetic Resonance Imaging (MRI); voxel-based morphometry (VBM); neuroplasticity; Huffaz; Brodmann-areas; Volume of Interest (VOI); texture analysis; machine learning; nested cross-validation; ROC-AUC; Random Forest

I. Introduction

Magnetic Resonance Imaging (MRI) provides a non-invasive window into brain structure, making it central to investigating neuroplasticity associated with intensive learning and memory training. Recent evidence from verbal memory experts indicates that long-term memorisation practice can be accompanied by measurable structural brain differences, reinforcing the plausibility of detectable morphology changes in high-repetition textual memorisation cohorts [1]. In our earlier work on Quran memorisers (Huffaz), a volume-of-interest (VOI) fractal framework was able to localise group differences that were not apparent in global measurements, suggesting that region-specific quantitative descriptors may be necessary to capture subtle experience-related neuroplasticity [2].

While voxel-based morphometry (VBM) remains a common approach for detecting regional grey matter differences, it typically summarises effects using voxel-wise statistics and may miss complex textural or distributional patterns within anatomically defined regions. Radiomics addresses this gap by extracting a large number of engineered quantitative features (e.g., first-order statistics and texture descriptors) from medical images, thereby encoding patterns that may not be obvious visually or via simple volumetry [3]. However, reproducibility has been a major barrier for radiomics, motivating standardised feature definitions and reporting practices. The Image Biomarker Standardisation Initiative (IBSI) provides consensus definitions for many radiomic features to improve comparability across studies and software implementations [4]. In addition, biological interpretation remains important; radiomic features are increasingly studied as proxies of underlying tissue heterogeneity, organisation, and microstructural variation [5].

Practical radiomics pipelines typically include: 1) image preparation, 2) region-of-interest definition, 3) feature extraction, and 4) predictive modelling with rigorous validation [6]. PyRadiomics is a widely used open-source implementation for IBSI-aligned radiomic feature extraction and supports multiple feature classes and filtered image types (e.g., wavelet decompositions) [3]. For clinical translation, recent guidance has emphasised methodological rigour, transparency, and robust evaluation, particularly in small-to-moderate datasets where model performance can be unstable and overly optimistic if validation is not handled carefully [7]. Reporting recommendations such as TRIPOD+AI further encourage clear documentation of data handling, modelling steps, and

performance metrics for machine learning prediction studies [8].

A key methodological risk in small neuroimaging datasets is biased performance estimation when model selection (e.g., hyperparameter tuning and feature selection) is not isolated from the outer evaluation loop. Nested cross-validation is recommended to reduce this bias, because it repeats all modelling steps within each training fold before testing on held-out data [9], and practical guidance for medical imaging AI specifically highlights cross-validation pitfalls and best practices [10]. Data leakage can also occur when resampling or preprocessing steps are applied outside the correct training-only scope, which has been shown to inflate radiomics performance estimates [11]. Finally, performance is commonly summarised with receiver operating characteristic (ROC) analysis and the area under the curve (AUC), and statistical tests or confidence estimation should be used to interpret whether observed AUCs exceed chance levels and to support model comparison [12], [13].

Motivated by these considerations, this study extends our earlier VOI-based neuroplasticity investigation in Huffaz [2] by replacing fractal descriptors with PyRadiomics-engineered features extracted from preprocessed grey matter MRI volumes within Brodmann-area-based VOIs. We then evaluate multiple classical machine learning models under a nested cross-validation framework and statistically test whether the observed AUC distributions are significantly above chance, aiming to provide an IBSI-aware, publication-ready workflow for quantitative neuroimaging analysis.

Research gap and objectives. Despite growing interest in radiomics for characterizing subtle tissue patterns, two gaps remain in the context of memorisation-related neuroimaging studies: 1) prior work on expert memorisers has more often focused on voxel-wise or volumetric summaries, which may not capture within-region distributional and texture patterns; and 2) radiomics findings can be sensitive to design choices (e.g., discretization/bin width, normalization, and leakage-free validation), yet these sensitivities are not consistently discussed or stress-tested in small neuroimaging cohorts.

While radiomics has shown promise in characterizing subtle imaging patterns across oncology and, increasingly, neuroimaging applications, several limitations are recurrent in the literature. First, many studies are conducted in small cohorts with high-dimensional feature spaces, which can inflate apparent performance and lead to unstable results when validation is not strictly nested or externally replicated. Second, reported performance is often difficult to compare across studies due to heterogeneity in preprocessing pipelines, ROI/VOI definitions, feature calculation settings, and model selection procedures. Third, radiomics features are known to be sensitive to methodological choices, particularly intensity discretization (e.g., fixed bin width vs. fixed bin count), intensity normalization, resampling resolution, and filtering options, all of which can shift feature distributions and downstream model behaviour. Consequently, reproducible radiomics reporting benefits from explicitly documenting feature-extraction settings and from conducting sensitivity analyses to quantify how binning and normalization decisions affect both features and model performance. These considerations motivated our emphasis on transparent reporting of PyRadiomics configuration and leakage-controlled model selection via repeated nested cross-validation; nevertheless, we treat our results as exploratory and highlight robustness testing and external validation as priorities for future work.

Accordingly, this study addresses the following research questions:

RQ1: Do VOI-based PyRadiomics first-order and texture features extracted from pre-processed T1-weighted MRI provide above-chance discrimination between Huffaz and controls under repeated nested cross-validation?

RQ2: Which classical classifier family (Logistic Regression, SVM-RBF, Random Forest) yields the most stable discrimination under leakage-controlled model selection?

Hypothesis (H1): Radiomics features extracted from the specified Brodmann-area VOIs will yield ROC-AUC statistically greater than 0.5 under a fully nested evaluation protocol.

Hypothesis (H2): Non-linear ensemble learning (Random Forest) will outperform linear (Logistic Regression) and kernel-based (SVM-RBF) models in this high-dimensional, low-sample setting.

II. Methodology

A. Overview

This study proposes a reproducible radiomics-based pipeline to quantify structural differences between Huffaz (H) and non-Huffaz (NH) participants using pre-processed structural MRI and atlas-derived Brodmann-area (BA) masks (see Fig. 1). The workflow consists of: 1) automated cohort/label parsing from filenames, 2) region-of-interest (VOI) preparation and spatial alignment of masks to subject space, 3) extraction of hand-crafted radiomics features using PyRadiomics, and 4) model development and evaluation using nested cross-validation with inner-loop hyperparameter tuning.

B. Experimental Setup

1) Hardware: Experiments were executed on a Runpod GPU instance with 6 vCPU, 15 GB RAM, and 16 GB VRAM. The GPU primarily supported the CUDA-enabled runtime environment; however, the classical machine learning models were trained using CPU-based scikit-learn routines.

2) Software: The pipeline was implemented in Python 3.11. Key libraries and versions were: PyRadiomics 3.0.1 (feature extraction), SimpleITK 2.5.3 (PyRadiomics backend), NiBabel 5.3.3 (NIfTI handling and resampling), NumPy 1.26.4, SciPy 1.17.0, Pandas 2.3.3, and scikit-learn 1.8.0. PyRadiomics logging was set to error level to reduce non-essential output. All outputs (features, metadata, fold-wise and summary results) were saved to disk to ensure auditability and repeatability.

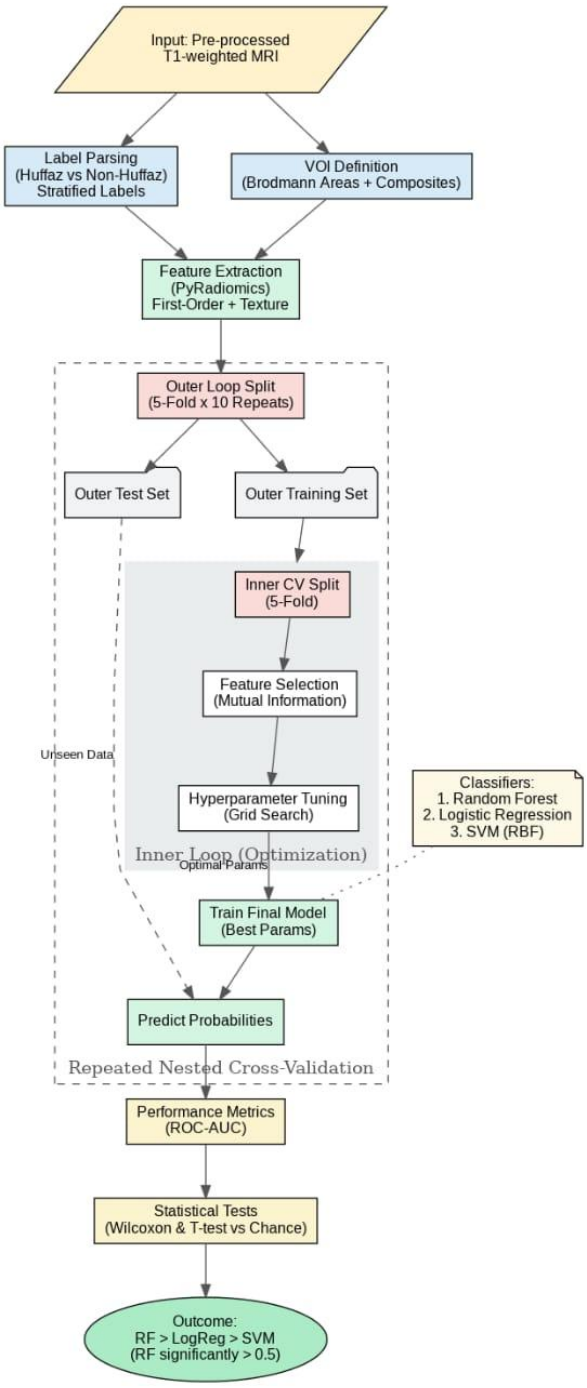


Fig. 1. Radiomics methodology flowchart showing the end-to-end pipeline: input pre-processed T1-weighted MRI, automated label parsing (Huffaz vs. non-Huffaz), and VOI definition using Brodmann-area masks, PyRadiomics feature extraction, and repeated nested cross-validation with inner-loop feature selection and hyperparameter tuning, followed by ROC-AUC–based evaluation and statistical testing.

C. Data Organisation and Label Encoding

The Neuroimaging Informatics Technology Initiative (NIfTI) pre-processed Magnetic Resonance Imaging files followed the naming convention:

smwc1{Group}{Sex}_{ID}.nii, where {Group} ∈ {H, NH} and {Sex} ∈ {F, M}.

An automated parser extracted the study label and participant attributes:

- Class label $y = 1$ for Huffaz (H) and $y = 0$ for non-Huffaz (NH),
- Sex (F/M),
- A combined stratification label {Group}_{Sex} ∈ {H_F, H_M, NH_F, NH_M}.

This stratification label was used to preserve both class and sex proportions in cross-validation splits.

Class balance. The dataset comprised n = 47 participants split into Huffaz (H) and non-Huffaz (NH) groups. We report the group counts as H: n = 23, NH: n = 24 to make any imbalance explicit. To reduce bias from potential imbalance during training, all classifiers were trained with balanced class weights, and cross-validation splits were stratified using the combined {Group}_{Sex} label to preserve the joint distribution of group and sex across folds.

D. Volume-of-Interest (VOI) Definition

Based on prior VOI findings, the following regions were included:

- BA20, BA30, BA33,
- BA39,
- BA21,
- BA37,

Two composite VOIs were created to represent broader functional regions:

1) Anterior cingulate cortex (ACC) as the union of BA24 and BA32:

$$\mathrm{ACC} = \mathrm{BA24} \cup \mathrm{BA32}$$

2) Frontal composite as the union of BA08, BA09, BA10, BA11, BA44, BA45, BA46, and BA47:

$$\mathrm{Frontal} = \bigcup_{b \in \{8,9,10,11,44,45,46,47\}} \mathrm{BA}b$$

Composite masks were computed programmatically using voxel-wise logical OR and saved for reproducibility.

E. Radiomics Feature Extraction (PyRadiomics)

Radiomics features were extracted per subject and per VOI using PyRadiomics 3.0.1. The extractor was configured to produce scalar features (not voxel-wise feature maps) by setting:

- voxelBased = False,
- force2D = False.

Intensity discretisation and preprocessing settings were:

- fixed bin width: binWidth = 0.01,
- no intensity normalisation: normalize = False,

- no resampling of the image intensity grid: resampledPixelSpacing = None.

Only PyRadiomics feature families were used, explicitly enabling:

- First-order,
- GLCM, GLRLM, GLSZM, GLDM, and NGTDM.

Shape features were excluded because atlas-defined VOIs are fixed anatomical regions after normalisation, and shape descriptors may be largely invariant across participants.

Each feature was stored with a unique VOI-qualified identifier:

$$\text{FeatureName} = \text{VOI} \;||\; \text{FeatureKey}$$

E.g., BA20__original_firstorder_Mean. PyRadiomics sometimes returns scalar values as NumPy arrays of length 1; these were converted to float scalars to construct the feature matrix consistently.

The extracted feature matrix was saved as features_{mode}_bw0.01.csv and metadata (case ID, group, sex, label, stratification label) as metadata.csv.

Reproducibility considerations. Radiomics features can be sensitive to discretization and intensity preprocessing choices. In this study, we used a fixed bin width (binWidth = 0.01) and disabled intensity normalisation (normalise = False) to keep the pipeline explicit and reproducible. Because such settings may influence downstream performance, future work will include sensitivity analyses across plausible bin widths and normalization strategies, reported alongside the primary results.

F. Classification Models and Feature Selection

Three supervised classifiers were evaluated:

1) Logistic regression: with balanced class weights and L_2regularisation (liblinear solver).

2) Support Vector Machine (RBF kernel): with probability estimates enabled and balanced class weights.

3) Random Forest with 500 trees, balanced class weights, and full CPU parallelism.

To address the high-dimensional feature space relative to the sample size, univariate feature selection was performed using mutual information:

- SelectKBest(mutual_info_classif).

The number of selected features $k$was tuned from:

$$k \in \{10,20,50,100,200,\text{all}\}$$

with automatic adjustment to ensure $k \leq$ number of available features.

G. Nested Cross-Validation and Hyperparameter Optimisation

Model evaluation employed nested cross-validation to avoid optimistic bias from hyperparameter tuning. Splits were stratified using the combined {Group}_{Sex} label to preserve distributions of both study group and sex.

- Outer loop: Stratified K-fold ($k = 5$) repeated 10 times (50 outer test evaluations per model).
- Inner loop: Stratified K-fold ($k = 5$) used for hyperparameter selection.

Hyperparameters were tuned using GridSearchCV with ROC-AUC as the optimisation criterion. Search spaces include:

- Logistic Regression: $C \in \{0.1,1,10,100\}$
- SVM-RBF: $C \in \{0.1,1,10,10, \gamma \in \{\text{scale}, 0.01,0.1,1\}$
- Random Forest: max_depth ∈ {None, 5, 10, 20}, min_samples_split ∈ {2, 5, 10}

A fixed seed (seed = 42) controlled the cross-validation shuffling and mutual-information estimation to support reproducibility.

H. Evaluation Metrics and Reporting

Model performance was assessed on the outer test folds of the repeated nested cross-validation procedure to ensure that all reported metrics reflect generalisation to unseen data. For each outer fold, classifiers produced class membership probabilities for the Huffaz class (H), which were used to compute the receiver operating characteristic area under the curve (ROC-AUC) as the primary discrimination metric. AUC was selected because it is threshold-independent and is widely used for binary classification evaluation, particularly under potential class imbalance. To summarise performance across resampling runs, fold-level metrics were aggregated over 50 outer-fold evaluations per model (5 folds × 10 repeats). For each classifier, we reported the mean and standard deviation of outer-fold AUC values.

Note on statistical dependence. Because the outer-fold AUCs are produced from repeated cross-validation, they are not strictly independent (training sets overlap across folds and repeats). Therefore, p-values from fold-wise tests against AUC = 0.5 should be interpreted as supportive evidence rather than definitive confirmatory inference. Consistent with this limitation, we emphasize effect size (mean AUC), variability (standard deviation), and stability across repeats, and we frame the findings as exploratory pending confirmation in an independent validation cohort.

To test whether each model performed significantly better than random guessing, we conducted one-sided hypothesis tests against the null AUC = 0.5, using the set of outer-fold AUC values. Two complementary tests were applied:

1) Wilcoxon signed-rank test on $(\text{AUC}-0.5)$, assessing whether the median improvement over chance was greater than zero.

2) One-sample t-test on AUC values, assessing whether the mean AUC exceeded 0.5.

III. Results and Discussion

A. Classification Performance (Repeated Nested Cross-Validation)

This performance was evaluated using repeated nested cross-validation (outer: 5 folds × 10 repeats; n = 50 outer-test AUCs per model). This design follows recommended practice to reduce optimistic bias when feature selection and hyperparameter tuning are performed [9].

Across the three classifiers trained on PyRadiomics VOI features (first-order + texture families), the mean AUC values were:

- Random Forest: 0.6704 ± 0.1792
- Logistic Regression: 0.5948 ± 0.2153
- SVM (RBF): 0.4356 ± 0.1927

Overall, Random Forest achieved the strongest discrimination, followed by Logistic Regression, while SVM_RBF underperformed relative to chance.

These results suggest that, under the current VOI definition and preprocessing assumptions, the extracted radiomics descriptors contain group-associated imaging signatures that provide above-chance discrimination in this cohort. However, given the modest mean AUC values and substantial fold-to-fold variability, these signals should not be interpreted as definitive evidence of neuroplastic change, but rather as exploratory markers that warrant validation and robustness testing.

B. Statistical Significance vs. Chance Level (AUC = 0.5)

The fold-level hypothesis tests were designed to assess whether each classifier's discrimination ability exceeded random guessing, using AUC = 0.5 as the null benchmark. Two complementary one-sided tests were applied. First, the Wilcoxon signed-rank test was performed on the set of values $(\mathrm{AUC}-0.5)$, testing whether the median improvement over chance was greater than zero. This non-parametric test is useful when the distribution of fold AUCs may not be normal and provides robustness to outliers. Second, a one-sample t-test was applied directly to the AUC values to test whether the mean AUC was greater than 0.5. While the t-test assumes approximately normal sampling distribution of the mean, the paired use of a non-parametric and parametric test provides a more interpretable and transparent assessment of "better-than-chance" performance.

For Logistic Regression, the mean AUC across the 50 outer-fold evaluations was 0.5948 with a relatively large standard deviation (0.2153), indicating that performance varied notably across cross-validation splits. Nevertheless, both tests yielded consistent evidence that the model exceeded chance: Wilcoxon p = 0.00132 and t-test p = 0.00154 (one-sided). This implies that, despite variability, the pipeline repeatedly produced AUC values above 0.5 more often than would be expected under random guessing. In practical terms, Logistic Regression appears to capture a modest but reliable linear separability in the radiomics feature space. The high variance suggests sensitivity to which subjects appear in each training/test fold, which is expected in small-sample radiomics; however, the consistent p-values across two different tests support that the observed effect is not driven solely by a few favourable folds.

In contrast, SVM with RBF kernel (SVM_RBF) showed a mean AUC of 0.4356 (std 0.1927), which is below the chance level of 0.5. Correspondingly, both one-sided tests for improvement over chance returned non-significant results (Wilcoxon p = 0.989, t-test p = 0.989). This indicates there is no evidence that the SVM_RBF model performed better than random guessing in this setting; instead, the direction of the effect suggests systematic underperformance. A mean AUC below 0.5 can happen when the learned decision function is effectively "reversed" relative to the chosen positive class, but in a properly configured pipeline with consistent label encoding, persistent below-chance performance more commonly reflects instability or poor generalisation for that model class under the current data regime. With only 47 subjects and a high-dimensional radiomics feature set, an RBF kernel can overfit subtle fold-specific patterns during inner-loop tuning and fail to transfer to outer-loop test folds, leading to degraded AUC.

The strongest results were obtained with the Random Forest classifier, which achieved a mean AUC of 0.6704 (std 0.1792). Both statistical tests provide very strong evidence that Random Forest performance exceeded chance: Wilcoxon p = 3.78×10^{-7} and t-test p = 8.94×10^{-9}. This consistency across parametric and non-parametric testing suggests that improvement over chance is not only present, but also robust across the majority of folds. The Random Forest model likely benefits from its ability to model non-linear relationships and interactions among radiomic features, which is particularly relevant when combining multiple texture families across multiple VOIs. Although variability remains (std ≈ 0.18), the shift of the AUC distribution above 0.5 appears sufficiently large that even with fold-to-fold fluctuations, the overall performance advantage persists.

Statistical vs. practical significance. While the hypothesis tests against AUC = 0.5 provide evidence that some models exceed chance in this resampling framework, the practical significance depends on the magnitude and stability of performance. The relatively large standard deviations indicate performance instability across splits, consistent with high-dimensional, low-sample radiomics settings. Therefore, the current results are best viewed as evidence of a potentially learnable signal rather than a clinically actionable classifier.

C. Why Random Forest Performed Best?

The Random Forest's advantage is consistent with the nature of radiomics: many features are correlated and may interact nonlinearly. Tree ensembles can exploit non-linear relationships and feature interactions with comparatively little feature engineering. In contrast, Logistic Regression is linear and will only separate classes well if the discriminative signal is approximately linear in the selected feature space.

The relatively high standard deviations (≈0.18–0.22) indicate that performance depends noticeably on which subjects fall into each fold, as expected given the small cohort size (47 subjects) and high-dimensional radiomics space. This

variability is also widely recognised as a challenge in radiomics pipelines and motivates careful model validation and feature selection.

D. Why SVM_RBF Underperformed?

The RBF SVM produced a mean AUC < 0.5 despite scaling and hyperparameter search. Likely contributors include:

1) Small-sample instability: RBF SVM can be sensitive to fold composition when sample size is limited; the decision boundary may overfit the inner-CV splits and fail to generalise.

2) Feature distribution characteristics: Radiomics features often include heavy tails and inter-feature dependencies; even with standard scaling, the effective geometry in feature space may not suit an RBF kernel under the tested grid.

3) High-dimensional, low-n regime: With many candidate features per VOI and relatively few subjects, kernel methods can become unstable unless regularisation and selection are very carefully constrained.

Given these results, SVM_RBF is not recommended as the primary model for this dataset without further stabilisation (e.g., more conservative feature selection, nested feature stability constraints, or a simpler kernel).

IV. Limitations

This study has several limitations that affect the reliability and generalizability of the findings. First, the cohort size (n = 47) is small relative to the dimensionality of radiomics features, which increases the risk of performance instability and optimistic estimates despite using nested cross-validation. This is reflected in the relatively large fold-to-fold standard deviations in AUC, suggesting that results are sensitive to which participants appear in each split.

Second, no independent external validation dataset was available; therefore, the results should be interpreted as exploratory rather than confirmatory, and the reported performance should not be treated as evidence of clinical utility.

Third, radiomics features can be sensitive to preprocessing choices such as discretization bin width and intensity normalization. We used a fixed bin width (binWidth = 0.01) and disabled normalization to make the pipeline explicit; however, alternative settings could change feature distributions and model performance. Fourth, the statistical tests against AUC = 0.5 rely on cross-validation-derived AUC samples that are not strictly independent due to overlapping training data across folds and repeats; thus, p-values are supportive rather than definitive. Finally, the VOI set was restricted to selected Brodmann regions and composites motivated by prior work; additional ROIs, confound control (e.g., age, education, scanner/site effects if applicable), and multimodal features may alter the observed signal. Future work should prioritise larger cohorts, external validation, sensitivity analyses of radiomics settings, and probability calibration reporting.

V. Conclusion

This study evaluated a VOI-based radiomics pipeline for distinguishing Huffaz from non-Huffaz participants using SPM pre-processed T1-weighted MRI and PyRadiomics first-order and texture features. Under repeated nested cross-validation, Random Forest achieved the highest mean discrimination (AUC = 0.6704 ± 0.1792), while Logistic Regression showed modest above-chance performance, and SVM-RBF did not generalize in this setting. Given the small cohort size and substantial variability across folds, the findings should be interpreted as exploratory evidence of group-associated imaging signatures rather than definitive proof of neuroplastic change or a clinically deployable classifier. Future work will focus on independent external validation, robustness analyses across radiomics preprocessing settings (e.g., discretization/normalization), expanded ROI coverage, and calibration assessment to determine whether predicted probabilities can be interpreted reliably.

Declaration on Generative AI

ChatGPT 5.2 was used to assist with language editing and to improve clarity during manuscript preparation. The authors take full responsibility for the content, confirm that all scientific interpretations and conclusions are their own, and approve the final version of the manuscript.

Acknowledgment

This work was supported by the Ministry of Higher Education of Malaysia under the Fundamental Research Grant Scheme (FRGS/1/2024/WAS12/UIAM/01/3).

References

[1] U. Kumar, A. Singh, and P. Paddakannaya, “Extensive long-term verbal memory training is associated with brain plasticity,” *Sci. Rep.*, vol. 11, Art. no. 9712, May 2021, doi: 10.1038/s41598-021-89248-7.

[2] I. Jamaludin, M. Z. C. Azemin, M. I. M. Tamrin, and A. H. Sapuan, “Volume of Interest-Based Fractal Analysis of Huffaz’s Brain,” *Fractal Fract.*, vol. 6, no. 7, Art. no. 396, July 2022, doi: 10.3390/fractalfract6070396.

[3] J. J. M. van Griethuysen, A. Fedorov, C. Parmar, A. Hosny, N. Aucoin, V. Narayan, R. G. H. Beets-Tan, J.-C. Fillion-Robin, S. Pieper, and H. J. W. L. Aerts, “Computational Radiomics System to Decode the Radiographic Phenotype,” *Cancer Res.*, vol. 77, no. 21, pp. e104–e107, Nov. 2017, doi: 10.1158/0008-5472.CAN-17-0339.

[4] A. Zwanenburg, M. Vallières, M. A. Abdalah, H. J. W. L. Aerts, V. Andrearczyk, A. Apte, S. Ashrafinia, S. Bakas, R. J. Beukinga, R. Boellaard, and *et al*., “The Image Biomarker Standardization Initiative: Standardized Quantitative Radiomics for High-Throughput Image-based Phenotyping,” *Radiology*, vol. 295, no. 2, pp. 328–338, May 2020, doi: 10.1148/radiol.2020191145.

[5] M. R. Tomaszewski and R. J. Gillies, “The Biological Meaning of Radiomic Features,” *Radiology*, vol. 298, no. 3, pp. 505–516, Mar. 2021, doi: 10.1148/radiol.2021202553.

[6] J. D. Shur, S. J. Doran, S. Kumar, D. ap Dafydd, K. Downey, J. P. B. O’Connor, N. Papanikolaou, C. Messiou, D.-M. Koh, and M. R. Orton,, “Radiomics in Oncology: A Practical Guide,” *RadioGraphics*, vol. 41, no. 6, pp. 1717–1732, Oct. 2021, doi: 10.1148/rg.2021210037.

[7] E. P. Huang, J. P. B. O’Connor, L. M. McShane, M. L. Giger, and *et al*., “Criteria for the translation of radiomics into clinically useful tests,” *Nat. Rev. Clin. Oncol.*, vol. 20, no. 2, pp. 69–82, Feb. 2023, doi: 10.1038/s41571-022-00707-0.

[8] G. S. Collins, K. G. M. Moons, P. Dhiman, and et al., "TRIPOD+AI statement: updated guidance for reporting clinical prediction models that use regression or machine learning methods," *BMJ*, vol. 385, p. e078378, Apr. 2024, doi: 10.1136/bmj-2023-078378.

[9] S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC Bioinformatics*, vol. 7, Art. no. 91, Feb. 2006, doi: 10.1186/1471-2105-7-91.

[10] T. J. Bradshaw, Z. Huemann, J. Hu, and A. Rahmim, "A Guide to Cross-Validation for Artificial Intelligence in Medical Imaging," *Radiol. Artif. Intell.*, vol. 5, no. 4, Art. no. e220232, July 2023, doi: 10.1148/ryai.220232.

[11] A. Demircioğlu, "Applying oversampling before cross-validation will lead to high bias in radiomics," *Sci. Rep.*, vol. 14, no. 1, Art. no. 11563, May 2024, doi: 10.1038/s41598-024-62585-z.

[12] L. Zou, Y.-H. Choi, L. Guizzetti, D. Shu, J. Zou, and G. Zou, "Extending the DeLong algorithm for comparing areas under correlated receiver operating characteristic curves with missing data," *Stat. Med.*, vol. 43, no. 21, pp. 4148–4162, Sep. 2024, doi: 10.1002/sim.10172.

[13] W. Zhang, Y. Guo, and Q. Jin, "Radiomics and Its Feature Selection: A Review," *Symmetry*, vol. 15, no. 10, Art. no. 1834, Sep. 2023, doi: 10.3390/sym15101834

Cognitive Assistance for Prosopagnosia Patients Using Landmark-Based Identity and Emotion Recognition

Cognitive Assistance System for Prosopagnosia Patients

Bhavana Nagaraj*, Rajanna Muniswamy
Department of Information Science and Engineering, Vemana Institute of Technology, Kormangala, Bengaluru, 560 034, India

***Abstract*—Prosopagnosia is a neurological condition that significantly affects the social interaction and quality of life of individuals. Existing assistive systems mainly focus on either face identity recognition or face emotion recognition, limiting their effectiveness in cognitive assistive scenarios. To overcome these, an integrated framework is needed that jointly addresses identity and emotion recognition to efficiently support prosopagnosia patients, as discussed in this study. The proposed system includes two separate modules: a face identity recognition module and a face emotion recognition module. The proposed system detects and aligns faces using Multi-Task Cascaded Convolutional Networks (MTCNN) with five-point landmark alignment. Face identity recognition is performed using an EfficientNet-B3 backbone to extract 512-dimensional facial embeddings, which are matched against a Structured Query Language (SQL) database using cosine similarity. Then, facial landmarks are detected along with emotion recognition, using the Dlib library, and are structured as a graph for High-order Graph Attention Network (HoGAN)-based relational interactions detection. The proposed system is trained using a joint loss function to efficiently provide real-time assistive feedback. The system achieves high recognition performance, with AUC values of 99.8% and 99.5% in both face identity recognition and face emotion recognition modules.**

***Keywords*—Cognitive assistance system; EfficientNet-B3; emotion recognition; face identity recognition; prosopagnosia**

I. Introduction

Prosopagnosia is an impaired ability to identify individuals by their faces, which complicates recognition of previously met individuals' faces, particularly unfamiliar faces, and also recognition of familiar faces [1],[2]. Individuals with prosopagnosia struggle to identify friends, family, colleagues, and, in some cases, their own face in the mirror. This results in psychosocial difficulties, including social anxiety, embarrassment, and problems in interpersonal relationships [3],[4]. These negative psychosocial consequences impact the quality of life of individuals due to their face recognition difficulties [5],[6]. Prosopagnosia patients also suffer from other aspects of cognition, including interpreting facial expressions and emotional cues, which are crucial for effective social communication [7],[8]. Furthermore, facial expressions are a significant form of non-verbal communication that can be utilized to gather someone's emotional state and likely intentions [9],[10]. These consequences lead to the significance of developing an effective face recognition and emotion recognition system for prosopagnosia patients, thereby optimizing daily activities and quality of life [11],[12].

Advanced neurocognitive models of face processing suggest that face identity and expression analyses proceed along distinct pathways. Individuals who experience acquired prosopagnosia are unable to recognize identity and are able to recognize expression, and those who experience impaired processing of expressions are able to recognize identity [13],[14]. Early approaches focused on feature-based and holistic face recognition techniques, while recent systems leverage Deep Learning (DL) models to extract discriminative facial embeddings for identity matching [15],[16]. Computer vision research is also helpful for facial emotion recognition performed through Deep Convolutional Neural Networks (DCNNs) [17],[18]. However, there is a limited emotion recognition system in a large sample of prosopagnosics using sensitive tests in recent years [19],[20]. Moreover, existing emotion recognition systems are typically designed for generic applications and are not optimized for real-time cognitive assistance mechanisms [21].

Existing DL-based approaches analyzed the face recognition and facial emotion recognition separately. Face identity recognition approaches focus on extracting discriminative embeddings using a Convolutional Neural Network (CNN). Existing emotion recognition approaches classify the facial expression through spatial and temporal facial features [22]. Despite these advancements, identity recognition and emotion recognition are treated as separate problems in current assistive technologies [23],[24]. The lack of an integrated framework jointly providing identity and emotional information limits the practical utility of these systems. Furthermore, existing solutions lack robustness analysis under unconstrained conditions, such as visually similar faces, partial occlusion, and non-frontal poses. To overcome these, this study aims to develop an integrated system that simultaneously performs face identity and emotion recognition to efficiently support prosopagnosia patients. The key contributions of the proposed work are summarized as follows:

- Unified Cognitive Assistance Framework: The study develops a supervised DL framework that jointly performs face identity recognition and facial emotion recognition, designed to support prosopagnosia patients in real-world social interactions.

- Efficient Identity Recognition Module: An EfficientNet-B3-based embedding extraction approach is employed to generate discriminative facial representations, enabling accurate identity retrieval through cosine similarity matching with a Structured Query Language (SQL)-based identity database.
- Landmark-Based Graph Emotion Module: A facial emotion recognition module is proposed that leverages a High-order Graph Attention Network (HoGAN) to model spatial and relational dependencies among emotion-relevant facial regions, enabling accurate and real-time emotion classification.
- Joint Learning Strategy: The system is trained using a combined loss function incorporating triplet loss for identity discrimination and categorical cross-entropy loss for emotion classification, ensuring balanced optimization of both tasks.
- Robust Real-Time Assistive Feedback: The proposed framework provides real-time integrated feedback combining identity and emotional information and demonstrates robustness under challenging conditions such as pose variation, occlusion, and visually similar faces.

The remaining part of the study is organized as follows: Section II reviews the existing works related to face identity and emotion recognition, Section III describes the workflow of the proposed unified deep learning framework, Section IV delivers the experimental results of the proposed framework, and Section V concludes the study.

II. Literature Survey

This section reviews existing works related to face identity recognition and face emotion recognition, which aim to support the proposed framework in cognitive assistive systems to support prosopagnosia patients.

A. Face Identity Recognition Approaches

Volfart et al. [25] validated two behavioral tests of Face Identity Recognition (FIR) using natural and ambient face images. The approach assessed recognition of famous identities without requiring verbal output and evaluated face identity matching across different views of familiar and unfamiliar faces. However, the approach needed pictures of famous faces for the population test. Olivares et al. [26] investigated how individuals with prosopagnosia process unfamiliar faces, focusing on the different and common neurocognitive mechanisms. The approach examined external and internal facial features in face-feature matching tasks and identified face perception and memory formation. Li [27] developed an integrated framework that combines Multi-feature Local Binary Pattern (MLBP), Histogram of Oriented Gradients (HOG), and Gray Level Co-occurrence Matrix (GLCM) features into a fusion algorithm (MLBP-HOG-G) to develop an accurate face recognition model. The framework addressed challenges like lighting changes, shadows, facial occlusion, and pose variations in unconstrained environments. Jain et al. [28] created an Augmented Reality (AR)-based 3D face recognition system as a social aid for individuals with face blindness that recognizes and identifies people in social contexts using a CNN with augmented reality. The approach supported prosopagnosia patients by providing real-time identification of faces during social interactions. Mukhiddinov et al. [29] introduced an assistive technology for the visually impaired individuals. The study utilized DL algorithms to design a smart glass system to enhance the quality of life of face-blindness patients. Still, the model incorrectly detects the small objects.

B. Face Emotion Recognition Approaches

Tsantani et al. [30] established that developmental prosopagnosia impairs facial expression recognition under challenging conditions, delivering new insights into the impairment in face processing. However, the study depended on a relatively small sample size of prosopagnosia participants. Wang et al. [31] investigated how neurotypical adults recognize and imitate dynamic emotional expressions and how these abilities are related to autistic traits. The study investigated emotion recognition and examined expression imitation that provided insights into how subtle variations in autistic traits influence social-emotional processing. Bardak and Temurta [32] created a model for identifying known and unknown faces using a regional brain perspective and simple neural networks. The features were classified using the K-Nearest Neighbors (KNN) algorithm, Probabilistic Neural Networks (PNN), and Support Vector Machine (SVM) to accurately identify the faces. However, the model needed enhancements to improve the accuracy through executing DL models. Guresli et al. [33] developed a Custom Lightweight CNN-based Model (CLCM) that optimized to reduce computational complexity while maintaining strong performance in emotion recognition tasks. However, the model did not accurately predict all the emotional states. Talaat et al. [34] developed a facial emotion recognition framework through DL techniques to accurately detect and classify emotional states. However, the model used only a limited amount of data to classify the emotional states. Mukhiddinov et al. [35] developed an emotion recognition system that utilized CNN-based models to detect and classify emotions. The system assisted visually impaired individuals to enhance social interaction and communication in daily life. However, in this study, facial landmark features were not correctly obtained.

C. Problem Statement

The analysis of existing works shows that existing face identity recognition approaches focus on improving feature discrimination, fail to identify the emotional state, as listed in Table I. Also, the facial emotion recognition systems only classify the emotional state and fail to identify the face. These limit their usefulness for assistive cognitive applications. Furthermore, most optimization strategies do not exploit shared facial representations through joint learning frameworks. Also, many existing methods are evaluated under controlled conditions and lack robustness against real-world challenges such as pose variations, occlusion, visually similar faces, and dynamic social environments. To overcome these, there is a need for an integrated system that simultaneously delivers identity recognition and emotion understanding in real-time. This supports prosopagnosia patients during daily social interactions.

TABLE. I. ANALYSIS OF EXISTING WORKS

References	Models / Approaches	Merits	Demerits
Volfart et al. [25]	Face identity recognition tests using natural and ambient face images	Evaluated face identity recognition without requiring verbal responses	Relied on images of famous faces.
Olivares et al. [26]	Neurocognitive analysis of unfamiliar face processing	Provided insights into face perception in prosopagnosia individuals	It doesn't provide a practical solution for prosopagnosia patients
Li [27]	MLBP-HOG-GLCM	Robust to illumination changes, shadows, occlusion, and pose variations	Features limit scalability and performance.
Jain et al. [28]	3D face recognition using CNN	Enabled real-time social assistance for prosopagnosia patients	High computational complexity
Mukhiddinov et al. [29]	Smart glass system using DL	Improved quality of life for face-blind individuals	Poor detection of small objects
Tsantani et al. [30]	Experimental study on facial expression recognition	Provided insights into emotion recognition impairments in prosopagnosia	Limited sample size
Wang et al. [31]	Dynamic emotion recognition and imitation analysis	Studied the relationship between emotion recognition and autistic traits	Doesn't suitable for prosopagnosia patients
Bardak and Temurta [32]	KNN, PNN, SVM	Attained a better performance	Lack resulting in lower recognition accuracy
Guresli et al. [33]	CLCM	Reduced computational complexity	Failed to accurately recognize all emotional states
Talaat et al. [34]	DL-based emotion recognition framework	Achieved accurate emotion classification	Trained on limited datasets
Mukhiddinov et al. [35]	CNN-based emotion recognition assistive system	Enhanced social interaction for visually impaired individuals	Facial landmark features were not accurately extracted.

III. MATERIALS AND METHODS

The proposed methodology develops a unified deep learning framework to provide cognitive assistance for prosopagnosia patients, as illustrated in Fig. 1. The proposed system was separated into two modules: a face identity recognition module and an emotion recognition module using the VGGFace2 and FER2013 datasets. The proposed system first detects and aligns input images using a Multi-Task Cascaded Convolutional Neural Network (MTCNN).

In the face recognition module, the aligned image is passed through an EfficientNet-B3 backbone that generates 512-dimensional face embeddings. These embeddings are stored in the SQL database, and cosine similarity is used to match the extracted embedding with stored identities. In parallel, the emotion recognition module operates on the same aligned face by detecting the facial landmarks using the Dilb landmark detector. Node features are derived from EfficientNet-B3 feature maps and processed using a HoGAN to learn relational interactions among emotion-relevant facial regions. Then, identity and emotion recognition modules are optimized using a joint loss function. Finally, the system integrates the retrieved identity information with the predicted emotional state to generate real-time assistive feedback that efficiently supports prosopagnosia patients.

A. Dataset Description

In this study, two publicly available benchmark datasets are utilized to support the tasks of face identity recognition and face emotion recognition, as shown in Fig. 2. First, the VGGFace2 dataset [36] is utilized, which is a large-scale face dataset designed to capture substantial intra-class variations. The dataset includes 3.31 million images of 9131 subjects, with wide diversity in pose, age, illumination, ethnicity, and background conditions. This dataset is suitable for learning robust facial embeddings and provides accurate face recognition in unconstrained environments. For emotion recognition, the FER2013 dataset [37] is used to train the emotion recognition module.

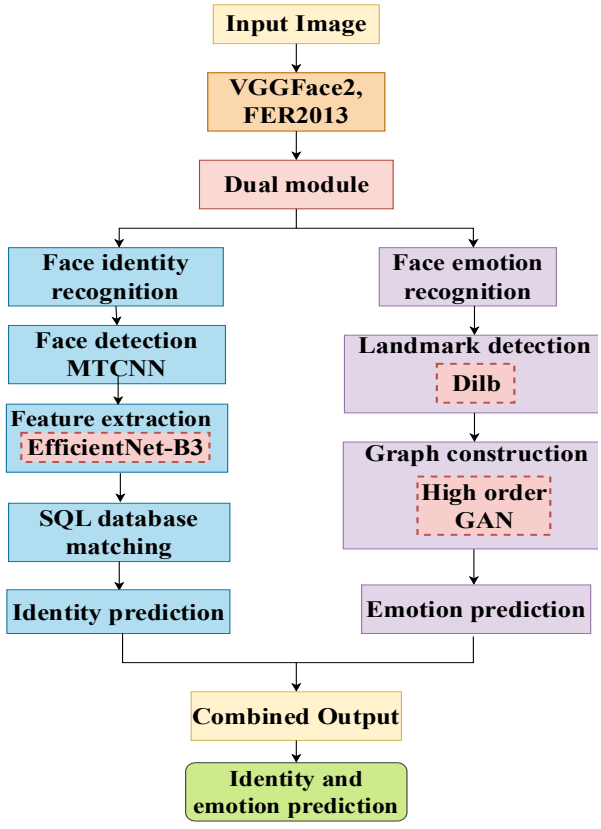


Fig. 1. Schematic representation of the proposed unified deep learning framework.

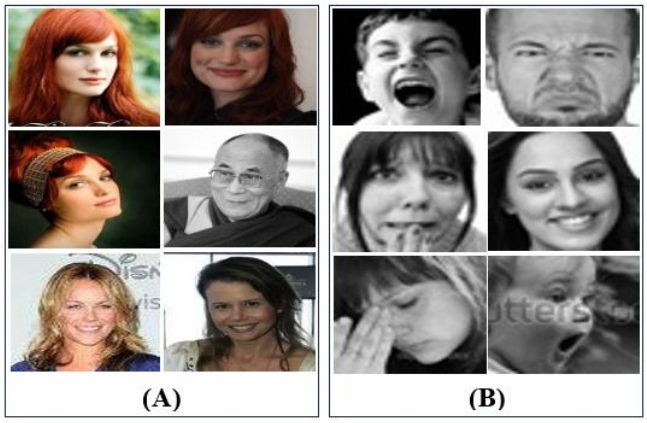


Fig. 2. Sample images from the dataset. (A) represents the sample images from the VGGFace2 dataset, and (B) represents the sample images from the FER2013 dataset.

The data consists of 48x48 pixel grayscale images of faces. The dataset consists of approximately 35,000 grayscale facial images, annotated into seven basic emotion categories: anger, disgust, fear, happiness, sadness, surprise, and neutral. The dataset includes images captured under diverse conditions with variations in facial expressions, occlusion, and illumination, making it suitable for learning generalized emotion representations. In this work, FER2013 is used only during the training phase to enable the model to learn emotion-specific facial patterns.

B. Face Detection and Alignment Using MTCNN

Face detection and alignment in the proposed framework are performed using the MTCNN [38], which consists of three parts: Proposal Network (P-Net), Refinement Network (R-Net), and Output Network (O-Net). The P-Net generates the image at multiple scales to generate a set of candidate face bounding boxes.

These bounding boxes are utilized to classify face and non-face windows, to estimate bounding box regression vectors for face location. R-Net rejects false candidates from the P-Net. Lastly, O-Net outputs five facial landmarks: the left and right mouth corners, the center of the nose, and the centers of the left and right eyes.

These landmarks are used to perform face alignment by estimating a similarity transformation that normalizes facial pose, scale, and rotation. The optimal transformation matrix T, is obtained by minimizing the alignment error between detected landmark positions P^*, as denoted in Eq. (1).

$$min \sum_{k=1}^{5} ||T(i_k, j_k) - (i_k{}^*, j_k{}^*)||_2^2 \quad (1)$$

where, k denotes the index of facial landmarks and (i_k, j_k) represents the detected coordinates of the k-th facial landmark. The aligned face is then cropped and resized to a fixed resolution of 224×224 pixels. This process improves the stability and accuracy of subsequent identity and emotion recognition modules.

C. Face Identity Recognition Module

The face identity recognition module is employed to extract discriminative facial representations of faces and identify individuals in unconstrained environments. It utilizes EfficientNet-B3 to generate facial embeddings. This matched against a stored identity database using cosine similarity to predict known or unknown identities.

1) Feature extraction through EfficientNet-B3: In this study, we utilized an EfficientNet-B3 [39] to extract deep features in the aligned images, as illustrated in Fig. 3. The pretrained model used is trained on ImageNet and is utilized as the convolutional backbone with all its parameters frozen. This increases a network's breadth, depth, and resolution and scales each dimension. The performance of models is improved when each size is scaled. EfficientNet-B3 uniformly balances network depth, width, and input resolution, as expressed in Eq. (2) – (4).

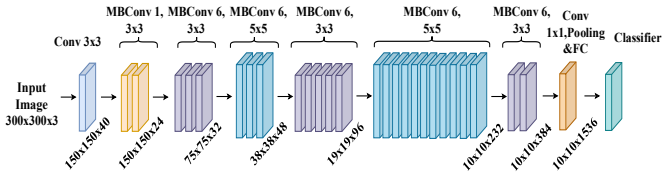


Fig. 3. Architecture diagram of the EfficientNet-B3.

$$depth_d = \alpha\theta \quad (2)$$

$$width_w = \beta\theta \quad (3)$$

$$resolution_r = \gamma\theta \quad (4)$$

where, $\alpha.\beta^2, \gamma^2$, where θ controls the overall model size and α, β, γ are scaling coefficients. The network is composed of Mobile Inverted Bottleneck Convolution (MBConv) blocks augmented with squeeze-and-excitation (SE) attention. This enhances channel-wise feature recalibration. For an intermediate feature map $X \in R^{H*W*C}$, the MBConv block first expands the channel dimension using a pointwise convolution. Then, a depth-wise convolution is used to efficiently model spatial information. Channel importance is then computed through global average pooling, as denoted in Eq. (5).

$$g = \frac{1}{hw} \sum_{u=1}^{h} \sum_{v=1}^{w} X(u, v) \quad (5)$$

where, g denotes the channel-wise descriptor vector, h denotes the height of the feature map X, and w denotes the width of the feature map X. The SE module generates channel attention weights, as mentioned in Eq. (6).

$$SE = \delta(W_2 \vartheta(W_1 g)) \quad (6)$$

where, δ and ϑ denotes the ReLU and sigmoid activation functions, respectively. W_1 and W_2 denotes the weight matrix of the fully connected layer. These weights are used to recalibrate the feature map, enabling the network to emphasize identity-relevant facial regions. The final convolutional feature maps produced by EfficientNet-B3 are aggregated using global average pooling to generate a compact feature vector f. This vector is then projected through a fully connected layer to obtain a fixed-length 512-d embedding (emd), as expressed in Eq. (7).

$$emd = W_f f + b_f \quad (7)$$

where, $W_f f$ denotes the learnable weight matrix and b_f signifies the corresponding bias vector. This encodes high-level facial characteristics suitable for identity discrimination. These embeddings serve as robust feature representations for subsequent identity matching and emotion analysis, balancing accuracy and efficiency in real-time facial recognition tasks.

2) SQL database matching for identity prediction: After extracting the 512-dimensional facial embedding from the EfficientNet-B3 feature extractor, identity prediction is performed through a SQL-based identity database [40]. During the enrolment phase, embeddings corresponding to known individuals are computed and stored in the database along with unique identity labels. During inference, the embedding obtained from a query face is compared against all stored embeddings e_s in the database using cosine similarity. The maximum similarity score is selected, and if it exceeds a predefined open-set threshold τ, the corresponding identity is assigned. The optimal threshold was selected by minimizing the Equal Error Rate (EER). This ensures a balanced trade-off between false acceptance and false rejection rates. Based on this analysis, τ was fixed at 0.65, which provided the best separation between known and unknown identities while maintaining stable real-time performance. Otherwise, the face is labeled as unknown. The identity associated with the maximum similarity score is selected as the predicted identity. The SQL database enables efficient storage, indexing, and retrieval of identity embeddings. This allows scalable and real-time identity matching. This structured matching mechanism allows reliable identity prediction while supporting dynamic updates, making it suitable for practical assistive applications for prosopagnosia patients.

D. Face Emotion Recognition Module

The face emotion recognition module is employed to identify the emotional state of an individual. It utilizes a landmark-based graph construction through Dilb and a HoGAN to capture spatial and relational dependencies among facial regions. This enables accurate real-time emotion classification to support prosopagnosia patients.

1) Facial landmark detection using Dilb: In the facial emotion recognition module, the proposed system performs landmark detection using the Dilb facial landmark detector [41]. This identifies 68 anatomically consistent key points distributed across critical facial regions. The detected landmark set is represented in Eq. (8).

$$L = \{(a_x, b_x)\}_{x=1}^{68} \tag{8}$$

where, (a_x, b_x) represents the two-dimensional coordinates of the x-th landmark. These landmarks capture subtle geometric variations in facial muscle movements that are strongly associated with emotional expressions. This normalization ensures consistency across faces of different sizes and orientations. This provides a stable geometric representation for subsequent graph-based emotion recognition.

2) Graph construction through high-order graph attention network (HoGAN): HoGAN [42] extends the conventional Graph Attention Network by modeling higher-order interactions that capture long-range dependencies across the graph. In this study, the facial landmark graph $G = (V, E)$, where V denotes the nodes and E denotes the edges, with node features $h_x \in \mathbb{R}^G$ for each landmark x, a standard attention mechanism computes pairwise attention coefficients between directly connected nodes, as denoted in Eq. (9).

$$e_{xy} = \text{LeakyReLU}(\text{w}^\top[\text{Mh}_x \parallel \text{Mh}_y]) \tag{9}$$

where, M denotes the learnable linear transformation, w represents the attention weight vector, and $\parallel$ signifies the concatenation. These coefficients are normalized using a SoftMax function over the neighborhood $\mathcal{N}_x$ of node x, as denoted in Eq. (10). This allows the network to adaptively weight the contribution of neighboring landmarks based on their relevance.

$$\sigma_{xy} = \frac{\exp(e_{xy})}{\sum_{k \in \mathcal{N}_x} \exp(e_{xk})}, \tag{10}$$

where, attention is extended to capture information from multi-hop neighborhoods by aggregating features from k-order neighbors. This enables the network to model complex facial muscle interactions across distant regions of the face. The high-order feature representation of the node x is computed as in Eq. (11).

$$\text{H}_i^{(k)} = \alpha\left(\sum_{y \in \mathcal{N}_x{}^k} M^k \, h_y\right) \tag{11}$$

where, $\mathcal{N}_x{}^k$ represents the set of nodes reachable within k hops, $\sigma_{xy}{}^k$ denotes the high-order attention coefficients, and $\alpha(.)$ represents the nonlinear activation function. Multiple attention heads are employed to stabilize learning and enhance representational capacity. The HoGAN effectively captures both localized and global facial dynamics that are critical for emotion expression. This capability is particularly beneficial for facial emotion recognition, as emotional cues are often distributed across multiple, spatially distant facial regions. Consequently, the high-order graph attention mechanism enables more discriminative and robust emotion modeling.

E. Joint Learning Strategy through Loss Functions

The proposed framework adopts a joint learning strategy to simultaneously optimize face identity recognition and facial emotion recognition within a unified DL framework. The model formulated the training dataset as D, where a_x represents the input facial image, b_x^{id} denotes the identity label, and b_x^{em} signifies the emotion label. Identity learning is formulated as a metric learning problem using triplet loss ($\mathcal{L}_{tr}$), as expressed in Eq. (12).

$$\mathcal{L}_{tr} = \sum_{i=1}^{N} \max\left(0, \| f(a_x^u) - f(a_x^p) \|_2^2 - \| f(a_x^u) - f(a_x^n) \|_2^2 + \alpha\right) \tag{12}$$

where, $f(\cdot)$ denotes the identity embedding function, N denotes the total images, and a_x^u, a_x^p, and a_x^n represent anchor, positive, and negative samples, respectively, with a margin α. Emotion recognition is optimized using a categorical cross-entropy loss ($\mathcal{L}_{cross}$), as denoted in Eq. (13).

$$\mathcal{L}_{cross} = -\sum_{c=1}^{C} b_c \log(\hat{b}_c) \quad (13)$$

where, C represents the number of emotion classes, b_c signifies ground-truth label, and $\hat{b}_c$ denotes predicted probability. The overall objective function is formulated as a weighted sum of both losses, as shown in Eq. (14).

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{tr} + \lambda_2 \mathcal{L}_{cross} \quad (14)$$

where, $\lambda_1$and λ_2 balance the contribution of identity and emotion tasks. This joint optimization enables the network to learn shared facial representations, improving generalization while ensuring efficient convergence for real-time assistive applications.

IV. Results and Discussion

This section presents the experimental performance of the proposed unified DL framework. The performance is assessed using quantitative metrics, qualitative visualizations, ablation analysis, and comparative analysis to demonstrate the effectiveness of the proposed system.

A. Hyperparameter Settings and Complexity Analysis

The proposed model finetunes hyperparameters to strike a balance between model performance and computational efficiency, as demonstrated in Table II. The proposed unified DL framework results in a balanced computational complexity for the proposed system. EfficientNet-B3 has a computational complexity of approximately $O(H \times W \times D)$, where H and Wdenote the spatial dimensions and D represents the channel depth. The emotion recognition module operates with a complexity of $O(K \times N^2)$, where $N = 68$ landmarks and K denotes the number of attention heads. This makes it suitable for real-world assistive applications for prosopagnosia patients.

TABLE. II. Hyperparameter Details of the Proposed Framework

Component	Hyperparameter	Value
MTCNN	Scale factor	0.709
	Number of landmarks	5
EfficientNet-B3	Dropout rate	0.3
	Embedding dimension	512
Dilb	Number of landmarks	68
HoGAN	Number of layers	3
	Attention heads	8
	Hidden dimension	128
	High-order hops	2-hop
Training Settings	Optimizer	Adam
	Learning rate	0.0001
	Weight decay	1×10^{-4}
	Batch size	32
	Epochs	10
	Triplet margin	0.2
	Loss weights (λ_1, λ_2)	1.0, 1.0

B. Performance Analysis of the Proposed Model

This section presents the performance analysis of the proposed unified deep learning framework, as illustrated in Table III. The performance was analyzed through the metrics of accuracy, precision, recall, specificity, f1-score, Matthews Correlation Coefficient (MCC), False Positive Rate (FPR), False Negative Rate (FNR), Equal Error Rate (EER), Cohen's kappa, and AUC values. The results show that the proposed model attains a better performance in both face identity recognition and face emotion recognition. This shows the ability of the proposed model to efficiently support patients with prosopagnosia.

TABLE. III. Performance of the Proposed Unified Deep Learning Framework

Metrics	Face identity recognition	Face emotion recognition
Accuracy (%)	99.58	99.02
Precision (%)	99.08	99.14
Recall (%)	99.11	99.09
Specificity (%)	99.58	99.39
F1-score (%)	99.09	99.21
MCC (%)	99.1	99.01
FPR	0.42	0.61
FNR	0.89	0.92
Equal Error Rate (EER)	0.45	0.87
Cohen's Kappa (%)	99.1	98.9
AUC (%)	99.8	99.5

C. Face Identity Recognition

Fig. 4 shows the results of MTCNN-based face detection and alignment. MTCNN detects faces across different subjects with variations in pose, illumination, facial appearance, and background. The five-point facial landmarks are used to align the faces that are detected, ensuring consistent positioning of key facial components such as the eyes, nose, and mouth. The visual results indicate the improved representation of the facial region following the alignment. This normalization significantly reduces intra-class variations and provides a stable input for the identity recognition module. The findings ensure that MTCNN is effective in preprocessing. This is essential in enhancing the accuracy of recognition in unconstrained and real-time assistive cases.



Fig. 4. Results of face detection and alignment through MTCNN. The first row denotes the original images of the VGGFace2 dataset, and the second row denotes the facial detection images.

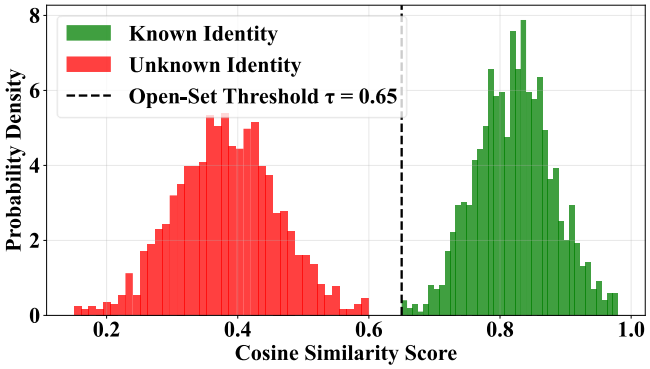


Fig. 5. Distribution of cosine similarity scores for known and unknown identities.

Fig. 5 illustrates the distribution of cosine similarity scores obtained during the identity matching process. The green histogram represents similarity scores for known identity pairs, which are concentrated at higher values. The red histogram corresponds to unknown identity pairs, which exhibit lower similarity scores. This reflects effective separation between different identities. The dashed line denotes the open-set decision threshold set at ($\tau = 0.65$), which serves as the boundary for distinguishing known and unknown identities. The minimal overlap between the two distributions demonstrates the discriminative power of the learned embeddings. This validates the effectiveness of the selected threshold in open-set identity recognition scenarios. This ensures reliable identity prediction while minimizing false acceptances and false rejections.

D. Emotion Recognition

Fig. 6 illustrates the results of facial landmark detection using the Dlib model. The green points represent the detected facial landmarks, which densely capture critical regions including the eyes, eyebrows, nose, mouth, and jawline. These precise geometric representations provide rich structural information, essential for modeling facial dynamics. This serving a reliable foundation for the subsequent graph-based emotion recognition module. This highlights its effectiveness in accurately localizing key facial features across different emotional expressions.

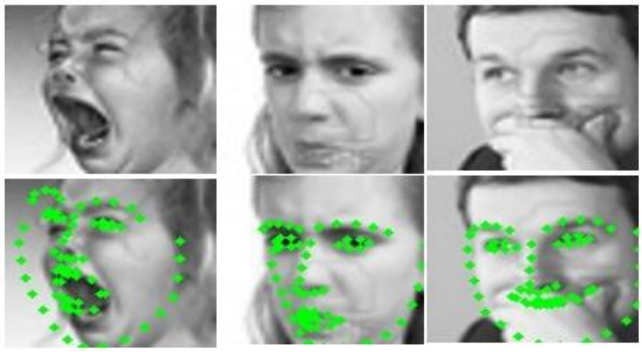


Fig. 6. Results of the facial landmark detector through Dilb. The first row denotes the original images of the dataset, and the second row denotes the facial landmark detection images.

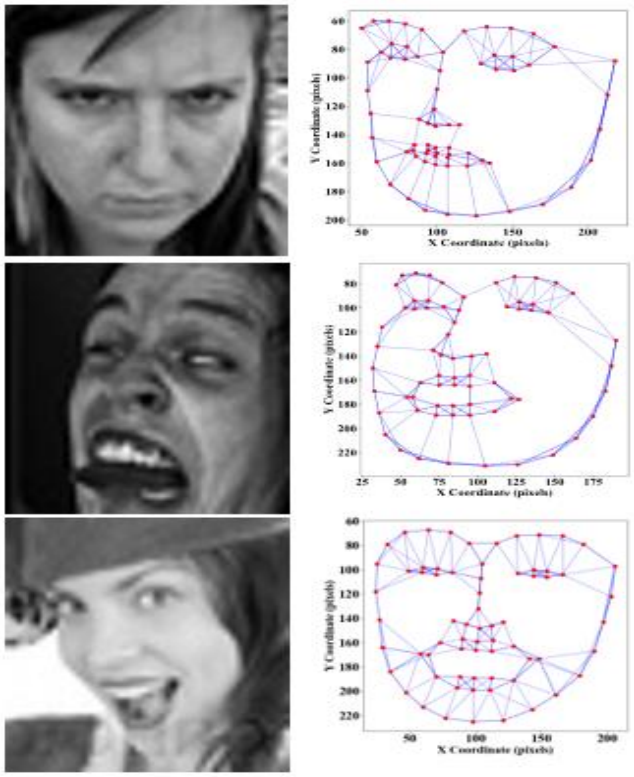


Fig. 7. Graph construction results through a high-order graph attention network.

Fig. 7 presents the graph construction results through HoGAN. The left column shows original facial images, while the right column illustrates the constructed facial graphs derived from the detected landmarks. Each red node represents a facial landmark, and the blue edges indicate the spatial and anatomical connections between landmarks used to form the graph structure. These structured graph representations enable to effectively capture of both local deformations and long-range dependencies between facial regions. This enhances the model's ability to discriminate subtle emotional cues. The visualization demonstrates that the proposed graph-based approach provides an interpretable and expressive geometric representation of facial dynamics.

E. Model Training Performance

Fig. 8 illustrates the training behavior and convergence characteristics of the proposed unified identity and emotion recognition framework. The accuracy curve shows a steady increase in classification accuracy as the number of training epochs progresses. The curve rises from approximately 60% in the initial epoch to nearly 99% by the final epoch. This consistent upward trend indicates effective feature learning and model convergence. This demonstrates that the network successfully captures discriminative facial representations with continued training. The rapid improvement in early epochs reflects efficient optimization, while the gradual saturation in later epochs suggests stable convergence without overfitting.

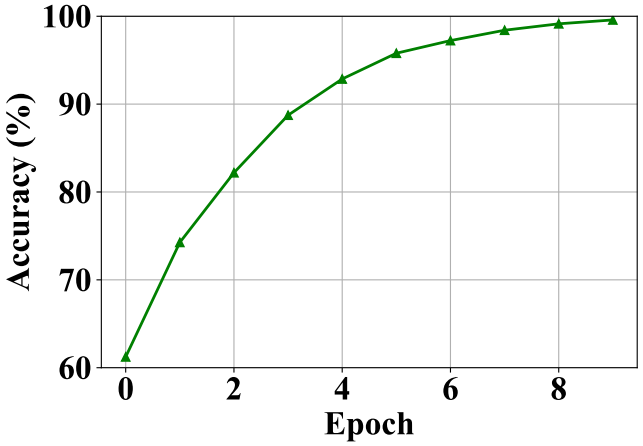


Fig. 8. Training accuracy of the proposed model across epochs, showing improvement and stable convergence during the learning process.

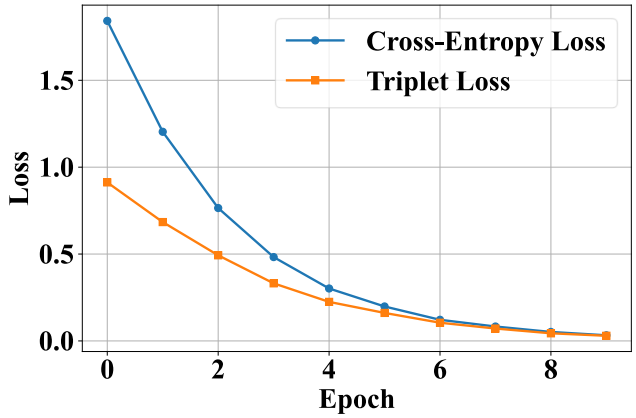


Fig. 9. Training loss curve of the proposed framework showing the reduction of cross-entropy loss for emotion recognition and triplet loss for face identity recognition across epochs.

Fig. 9 illustrates the loss function performance of the proposed framework. Both losses decrease across epochs, with cross-entropy loss reducing sharply during early training. Triplet loss gradually converges as embedding separability improves. The parallel decline of these losses highlights the effectiveness of the joint multi-task learning strategy, where identity and emotion objectives are optimized concurrently. These confirm that the proposed model achieves stable training, balanced optimization across tasks, and robust convergence. This supports its suitability for real-time cognitive assistance applications.

Fig. 10 illustrates the performance of the proposed facial emotion recognition module across seven emotion classes: Angry, Disgust, Fear, Happy, Sad, Surprise, and Neutral. Each emotion class achieves a high number of correct predictions. This indicates high classification accuracy and effective discrimination between emotional states. Misclassifications are minimal and primarily occur between visually similar emotions, such as fear and sadness or anger and disgust. This confirms the reliability and stability of the proposed emotion recognition module, validating its effectiveness for real-world cognitive assistance applications.

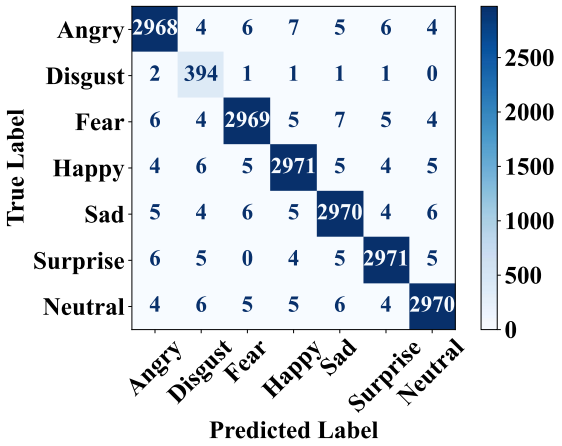


Fig. 10. Confusion matrix of the proposed framework.



Fig. 11. Results of the proposed prosopagnosia assistive system showing face identity and emotion recognition. This predicted identity labels and recognized emotional states.

Fig. 11 illustrates the results of the proposed prosopagnosia assistive system, showing its ability to simultaneously recognize face identity and facial emotion. Each shows an input image processed by the system, with the annotated output displaying the predicted identity label and the recognized emotional state. The examples highlight the system's robustness under varying conditions, including different lighting, facial poses, backgrounds, and emotional intensities. These results confirm the practical applicability of the proposed framework as a real-time cognitive assistance system. This helps to efficiently support individuals with prosopagnosia during everyday interactions.

F. Comparative Analysis

This section presents the comparative analysis of the proposed model, as represented in Table IV. The proposed framework is compared with the baseline models, MLBP-HOG-G, KNN, PNN, SVM, CNN, CLCM, and other approaches. The proposed model attained an accuracy of 99.58% and 99.02% in both modules. This shows that the proposed model attained a better accuracy compared to baseline models, and shows the efficiency in both face identity recognition and face emotion recognition.

TABLE. IV. COMPARATIVE ANALYSIS WITH BASELINE MODELS

Models	Face identity recognition			Face emotion recognition		
	Accuracy (%)	Precision (%)	F1-score (%)	Accuracy (%)	Precision (%)	F1-score (%)
MLBP-HOG-G	87.42	86.91	86.99	87.13	86.92	86.96
KNN	88.17	87.64	87.79	86.47	86.12	86.22
PNN	89.54	89.02	89.16	89.92	87.61	87.67
SVM	91.83	91.42	91.51	90.36	90.11	90.17
CNN	94.26	94.08	94.12	93.58	93.34	93.40
CLCM	96.48	96.31	96.35	95.94	95.81	95.85
GAT	98.12	98.01	98.03	97.61	97.49	97.52
Proposed	**99.58**	**99.08**	**99.09**	**99.02**	**99.14**	**99.21**

G. Discussion

This study presented a unified framework for face identity and facial emotion recognition aimed at addressing the real-world challenges faced by individuals with prosopagnosia. The experimental results demonstrate that integrating face detection and alignment with EfficientNet-B3-based identity feature extraction and landmark-driven HoGAN for emotion recognition leads to highly accurate performance. The joint learning strategy enables the model to exploit information between identity and emotion tasks. The study utilized the VGGFace2 and FER2013 benchmark datasets. They may introduce biases related to demographic distribution, illumination conditions, and expression diversity. VGGFace2 primarily contains celebrity images, which may not fully represent real-world population variability. FER2013 consists of low-resolution grayscale images that may limit fine-grained emotional representation. These dataset characteristics influence model generalization in practical deployments. To mitigate this, data augmentation and joint learning were employed to improve robustness. The strong separation observed in cosine similarity distributions indicates that the proposed system is well-suited for real-time assistive applications.

Although the proposed framework achieves very high performance across multiple evaluation metrics, it is important to critically consider potential overfitting effects. The high accuracy values may partially result from training and evaluation on benchmark datasets. To mitigate overfitting, regularization strategies including dropout, joint loss optimization, and validation monitoring were employed during training. The stable convergence behavior observed in training accuracy and loss curves suggests balanced learning. However, the proposed framework acknowledged some limitations. The study depends on publicly available datasets, which do not fully capture the diversity of real-world interactions experienced by prosopagnosia patients. Future work will focus on addressing these limitations by incorporating multimodal information such as speech, contextual cues, and physiological. This provides a foundation for intelligent assistive technologies for advancing cognitive support systems for prosopagnosia patients.

V. CONCLUSION

This study successfully developed a unified DL framework for face identity and facial emotion recognition to support individuals with prosopagnosia in real-world social interactions. The proposed model employed a dual module framework that simultaneously performs face identity recognition and face emotion recognition. The proposed system effectively captured both identity-specific and expression-related facial cues. The joint learning strategy enabled optimization of identity and emotion tasks, leading to high recognition accuracy, stable convergence, and strong generalization performance. The experimental results demonstrated that the proposed approach outperforms conventional task-specific models while maintaining real-time efficiency. The framework provides a scalable solution for cognitive assistance, offering an efficient identity and emotion framework to support the daily social interactions of individuals with prosopagnosia. In the future, we will extend the proposed framework under different real-world conditions, ensuring robust deployment in cognitive, practical assistive environments.

STATEMENTS AND DECLARATIONS

Author contributions: Both authors contributed to the conception of the problem setting, conceptualization, methodology, implementation, testing and writing the final manuscript.

Funding: No funding was received for conducting this study.

Availability of data and materials:

https://github.com/ox-vgg/vgg_face2
https://datarepository.wolframcloud.com/resources/FER-2013

Conflict of interest: The authors declare that they have no conflict of interest.

Ethical approval: The research is original, and the authors of this manuscript created all the figures and tables.

Consent to participate: Not applicable.

REFERENCES

[1] S. Byrne, M. and Porter, "Rehabilitation and intervention of developmental and acquired prosopagnosia: A systematic review," Neuropsychological Rehabilitation, vol. 35, pp. 1-44, Jan 2025.

[2] K. A. Josephs, and K. A. Josephs Jr, "Prosopagnosia: face blindness and its association with neurological disorders," Brain Communications, vol. 6, pp. fcae002, Jan 2024.

[3] E. Nørkær, S. Gobbo, T. Roald, and R. Starrfelt, "Disentangling developmental prosopagnosia: A scoping review of terms, tools and topics," Cortex, vol. 176, pp. 161-193, Jul 2024.

[4] X. Xu, X. He, W. Ren, and X. Zhao, "The network structure of autistic traits, executive function, prosopagnosia and social anxiety," Research in Autism, vol. 131, pp. 202815, Mar 2026.

[5] J. Lowes, L. M. McGregor, P. J. Hancock, B. Duchaine, and A. K. Bobak, "This condition impacts every aspect of my life: A survey to understand the experience of living with developmental prosopagnosia," PloS one, vol. 20, pp. e0322469, Apr. 2025.

[6] B. K. Devisetty, A. Goyal, A. Mishra, M. W. Nijim, D. Hicks, and G. Toscano, "A Hybrid Regression-Based Network Model for Continuous Face Recognition and Authentication," International Journal of Advanced Computer Science & Applications, vol. 15, p30, Oct. 2024.

[7] E. J. Burns, E. Gaunt, B. Kidane, L. Hunter, and J. Pulford, "A new approach to diagnosing and researching developmental prosopagnosia: Excluded cases are impaired too," Behavior research methods, vol. 55, pp.4291-4314, Dec. 2023.

[8] T. Halder, K. Ludwig, and T. Schenk, "Binocular rivalry reveals differential face processing in congenital prosopagnosia," Scientific Reports, vol. 14, pp. 6687, Mar. 6687.

[9] J. J. Barton, and F. E. I. L. Moritz, "Foundations of prosopagnosia: The three classic Austro-German reports," Cortex, vol. 193, pp. 1-15, Dec 2025.

[10] S. Djouab, A. Albonico, S. C. Yeung, M. Malaspina, A. Mogard, R. Wahlberg, and J. J. Barton, "Search for face identity or expression: Set size effects in developmental prosopagnosia," Journal of Cognitive Neuroscience, vol. 32, pp. 889-905. May 2020.

[11] H. T. Wang, K. M. Chuang, T. Rawat, J. L. Lyu, M. Ali, and S. H. L. Chien, "Subjective face recognition ability is linked to objective face memory and face authenticity judgment: validation of the Traditional Chinese version of the 20-Item Prosopagnosia Index," Brain Sciences, vol. 15, pp.1186, Oct. 2025.

[12] R. Fry, X. Li, T. C. Evans, M. Esterman, J. Tanaka, and J. DeGutis, "Investigating the influence of autism spectrum traits on face processing mechanisms in developmental prosopagnosia," Journal of autism and developmental disorders, vol. 53, pp.4787-4808, Dec 2023.

[13] L. Bell, B. Duchaine, and T. Susilo, "Dissociations between face identity and face expression processing in developmental prosopagnosia," Cognition, vol. 238, pp. 105469, Sep 2023.

[14] K. Minemoto, and Y. Ueda, "Face identity and facial expression representations with adaptation paradigms: New directions for potential applications," Frontiers in psychology, vol. 13, pp. 988497, Dec 2022.

[15] B. Q. Z. Leong, A. M. Hussain Ismail, and A. J. Estudillo, "Persistent task-specific impairment of holistic face processing in acquired prosopagnosia," Scientific Reports, vol. 15, pp. 43115, Dec 2025.

[16] M. A. Hamzah, "Advancing personal identity verification by integrating facial recognition through deep learning algorithms," International Journal of Information Technology, vol. 16, pp. 4381-4386, Oct 2024.

[17] W. Y. Hsu, and Y. C. Chen, "Multi-attribute feature-aware network for facial expression recognition," ACM Transactions on Multimedia Computing, Communications and Applications, vol. 21, pp. 1-20, July 2025.

[18] A. Kazemian, I. Oruc, and J. J. Barton, "Scanning faces: a deep learning approach to studying eye movements in prosopagnosia," Frontiers in Neurology, vol. 16, pp.1616509, Sep 2025.

[19] V. Manippa, and A. Palmisano, M. Ventura, and D. Rivolta, "The neural correlates of developmental prosopagnosia: Twenty-five years on," Brain Sciences, vol. 13, pp.1399, Sep 2023.

[20] S. Albalawi, L. Alamri, J. Atut, S. Albalawi, and R. Haddaddi, "MAHYA: Facial Recognition-Based Pilgrim Identification System for Enhanced Health Monitoring and Assistance," International Journal of Advanced Computer Science & Applications, vol. 16, pp. 928- 941, Mar 2025.

[21] V. Singh, and S. Prasad, "Speech emotion recognition system using gender dependent convolution neural network," Procedia Computer Science, vol. 218, pp.2533-2540, Jan 2023.

[22] J. Ma, X. Wang, and Y. Li, "Specific Deficits in Facial Recognition in Children Aged 7–14 Years With Developmental Prosopagnosia," Journal of Autism and Developmental Disorders, pp.1-14, Nov 2025.

[23] P. Naga, S. D. Marri, and R. Borreo, "Facial emotion recognition methods, datasets and technologies: A literature survey," Materials Today: Proceedings, vol. 80, pp. 2824-2828, Jan 2023.

[24] A. Zhalgas, B. Amirgaliyev, and A. Sovet, "Robust Face Recognition Under Challenging Conditions: A Comprehensive Review of Deep Learning Methods and Challenges," Applied Sciences, vol. 15, pp. 9390, Aug 2025.

[25] A. Volfart, C. Michel, and B. Rossion, "A simple behavioral evaluation test of human face identity recognition with natural images validated with the case of prosopagnosia PS," Scientific Reports, vol. 15, pp. 43149, Dec 2025.

[26] E. I. Olivares, A. S. Urraca, A. Lage-Castellanos, and J. Iglesias, "Different and common brain signals of altered neurocognitive mechanisms for unfamiliar face processing in acquired and developmental prosopagnosia," Cortex, vol. 134, pp. 92-113, Jan 2021.

[27] X. Li, "The Application of Face Recognition Model Based on MLBP-HOG-G Algorithm in Smart Classroom," International Journal of Advanced Computer Science & Applications, vol. 16, p724, Mar 2025

[28] W. H. Jain, B. G. Jhong, and M. Y. Chen, "A Social Assistance System for Augmented Reality Technology to Redound Face Blindness with 3D Face Recognition," Electronics, vol. 14, pp. 1244, Mar 2025.

[29] M. Mukhiddinov, and J. Cho, "Smart glass system using deep learning for the blind and visually impaired," Electronics, vol. 10, pp.2756, Nov 2021.

[30] M. Tsantani, K. L. Gray, and R. Cook, "New evidence of impaired expression recognition in developmental prosopagnosia," Cortex, vol. 154, pp. 15-26, Sep 2022.

[31] H. T. Wang, J. L. Lyu, and S. H. L. Chien, "Dynamic Emotion Recognition and Expression Imitation in Neurotypical Adults and Their Associations with Autistic Traits," Sensors, vol. 24, pp. 8133, Dec 2024.

[32] F. K. Bardak, and F. Temurtaş, "Regional Brain Analysis and Machine Learning Techniques for Classifying Familiar and Unfamiliar Faces Using EEG," Arabian Journal for Science and Engineering, vol. 50, pp. 1-26, Jan 2025.

[33] M. C. Gursesli, S. Lombardi, M. Duradoni, L. Bocchi, A. Guazzini, and A. Lanata, "Facial emotion recognition (FER) through custom lightweight CNN model: performance evaluation in public datasets," IEEE Access, vol. 12, pp. 45543-45559, Mar 2024.

[34] F. M. Talaat, "Real-time facial emotion recognition system among children with autism based on deep learning and IoT," Neural Computing and Applications, vol. 35, pp. 12717-12728, June 2023.

[35] M. Mukhiddinov, O. Djuraev, F. Akhmedov, A. Mukhamadiyev, and J. Cho, "Masked face emotion recognition based on facial landmarks and deep learning approaches for visually impaired people,". Sensors, vol. 23, pp. 1080, Jan 2023.

[36] VGGFace2 dataset link: https://github.com/ox-vgg/vgg_face2.

[37] FER2013 dataset link: https://datarepository.wolframcloud.com/resources/FER-2013

[38] M. Gu, X. Liu, and J. Feng, "Classroom face detection algorithm based on improved MTCNN," Signal, Image and Video Processing, vol. 16, pp.1355-1362, Jul 2022.

[39] M. U. Naveed, M. M. Iqbal, S. Majeed, F. Ali, and Q. G. K. Safi, "Lung Cancer Classification through Transfer Learning and Deep Feature Extraction using EfficientNetB3," Journal of Computing & Biomedical Informatics, vol. 9, Sep 2025.

[40] W. Khan, T. Kumar, C. Zhang, K.Raj, A. M. Roy, and B. Luo, "SQL and NoSQL database software architecture performance analysis and assessments—a systematic literature review," Big Data and Cognitive Computing, vol. 7, pp. 97, May 2023.

[41] J. Wang, C. He, and Z. Long, "Establishing a machine learning model for predicting nutritional risk through facial feature recognition," Frontiers in Nutrition, vol. 10, pp. 1219193, Sep 2023.

[42] L. He, L. Bai, X.Yang, H. Du, and J. Liang, "High-order graph attention network," Information Sciences, vol. 630, pp. 222-234, June 2023.

Comparative Analysis of Neural Network Architectures for Classifying Depressive Content in Social Networks

Yntymak Abdrazakh[1], Rita Ismailova[2], Nurseit Zhunissov[3]*,
Arypzhan Aben[4], Anuarbek Amanov[5], Aigerim Baimakhanova[6]*
Khoja Akhmet Yassawi International Kazakh-Turkish University, Turkistan, Kazakhstan[1, 3, 4, 5, 6]
Kyrgyz-Turkish Manas University, Bishkek, Kyrgyzstan[2]

***Abstract*—Depression-related language on social media provides measurable signals for population-level mental-health research, yet model selection remains sensitive to evaluation protocol, domain shift, class imbalance, and computational constraints. This study benchmarks CNN, LSTM, and transformer encoders (BERT, RoBERTa, DistilBERT, and MentalBERT) for binary depression-indicative versus control classification on a unified corpus of 19,800 English posts/comments aggregated from three platforms (Reddit, Twitter, and Facebook) under a consistent preprocessing pipeline. We report two complementary evaluation protocols: (1) a fixed-split single-run baseline for a comparable snapshot, and (2) a five-seed repeated-run protocol with statistical testing (effect sizes and multiple-comparison correction) to quantify variability and reduce sensitivity to initialization effects. Under repeated-run reporting, MentalBERT achieves the best overall performance (F1 = 0.918 ± 0.005; AUC = 0.962 ± 0.002), while CNN/LSTM baselines show lower robustness under cross-platform transfer. Cross-domain experiments reveal a consistent performance drop relative to in-domain evaluation, confirming non-trivial platform shift and motivating robustness-aware reporting for deployment-oriented settings. In addition to predictive metrics, we report training time, inference latency, and derived throughput to support practical model selection for use cases such as moderation pipelines and screening/triage dashboards.**

***Keywords*—Depression detection; social media text; natural language processing; cross-platform evaluation; robustness; statistical significance testing; transformer-based models; CNN; LSTM; BERT; MentalBERT**

I. Introduction

In modern society, depression has become an increasingly pressing public-health challenge, influencing not only individual well-being but also broader social and economic outcomes. According to the World Health Organization, depression affects approximately 332 million people worldwide (about 4% of the global population) and is a common mental disorder that can substantially impair daily functioning and, in severe cases, increase suicide risk [1]. In parallel, the rapid growth of digital footprints on social media platforms (e.g., Twitter, Reddit, Facebook) provides large-scale textual evidence where users may express emotional distress and related indicators, making text-based analysis a promising direction for population-level risk monitoring and research support. Prior work shows that linguistic signals in social-media text are associated with depression-related outcomes and can support NLP-based screening and monitoring in research settings [2], [3].

In recent years, there has been substantial progress in automated assessment of a person's psychological state through text analysis. Natural Language Processing (NLP), combined with machine-learning methods, provides computational tools for detecting depression-related signals from user-generated statements and other online textual traces [4]. Recent surveys and systematic reviews describe the evolution of this field from feature-engineered classifiers toward deep representation learning, including neural architectures and transformer-based approaches for text classification [5], [6]. In this context, NLP-based methods can support research-oriented screening and monitoring of depression and anxiety indicators; however, model outputs should be interpreted as risk signals rather than clinical diagnoses.

Early computational mental-health studies on online platforms often relied on bag-of-words or lexicon-based representations combined with linear classifiers (e.g., SVM, Naïve Bayes, logistic regression) as strong baselines for depression-related text classification. In community and forum settings, such approaches were evaluated on large-scale resources built from users' self-reported signals (e.g., self-reported diagnosis–based forum resources), which helped establish benchmark tasks for depression detection from language [7]. However, feature-engineered pipelines can struggle with subtle semantics, pragmatic cues, and longer-range context in noisy user-generated text, motivating deep representation learning. This shift promoted architectures that learn hierarchical and sequential representations directly from text, including CNN- and LSTM-based models that became strong baselines for sentence- and post-level classification tasks [8],[9]. More recently, transformer-based encoders and ensemble strategies have further improved contextual modeling for depression detection across social platforms, reinforcing the move toward attention-based architectures in modern benchmarks [10].

Transformer-based architectures have substantially advanced NLP by enabling global context modeling via self-attention mechanisms [11]. Building on this framework, BERT introduced deep bidirectional contextual representations and achieved strong performance across a wide range of language understanding and classification tasks, including text

*Corresponding author.

classification settings relevant to affective and mental-health NLP [12].

The present study provides a comparative assessment of CNN, LSTM, and BERT models in the context of depression detection from text-based data. All models were trained under identical conditions to ensure comparability. The analysis focuses on key performance metrics such as Accuracy, Precision, Recall, F1-score, and AUC, aiming to determine which architecture demonstrates the highest reliability in identifying depressive language patterns.

Despite the strong performance of CNN and LSTM baselines, recent work increasingly favors transformer-based encoders, particularly BERT and its variants, for mental-health–related text classification, due to their ability to model long-range context and bidirectional dependencies [12], [13]. More recently, several works have explored fine-tuning large language models for depression detection on social-media benchmarks and reported strong performance under their experimental settings [14]. In broader benchmarking and comparative studies, transformer-based encoders are frequently reported to be competitive on mental-health NLP tasks, including depression and anxiety detection [15]. However, statistical significance testing is sometimes omitted or misapplied in NLP benchmarking, which can lead to overconfident conclusions [16]. Practical deployment must still consider computational requirements, inference latency, and memory footprint compared to lighter neural architectures [17].

Despite substantial progress in neural text classification for mental health applications, the question of how different deep learning architectures compare under controlled and deployment-relevant conditions remains insufficiently explored. Existing studies often focus on a single model family, report results on a single dataset, or rely on one-off train–test splits without accounting for variability induced by random initialization and training dynamics. As noted in prior surveys and comparative reviews, comprehensive analyses that systematically contrast CNN, LSTM, and transformer-based models in depression-related text classification remain relatively scarce [15]. As a result, model selection is frequently based on isolated performance snapshots rather than on robust, reproducible evidence, complicating the identification of architectures that achieve a balanced trade-off between predictive accuracy and computational efficiency in real-world systems.

The purpose of this study is to conduct an empirical comparison of CNN, LSTM, and BERT architectures for binary classification of social media texts with respect to depressive versus non-depressive language. In addition to standard quality metrics (Accuracy, Precision, Recall, F1-score, and AUC), we quantify run-to-run variability under repeated training, report effect sizes with multiple-comparison control, and evaluate cross-platform transfer to better reflect deployment-oriented conditions.

Deployment motivation. The intended deployment setting is a post-level text classifier integrated into a decision-support workflow (e.g., a research screening dashboard or a content-moderation triage pipeline) where the model flags potentially depressive posts for human review. In such settings, the objective is not clinical diagnosis but reliable risk-sensitive prioritization under platform shift, bounded compute, and real-time throughput constraints.

In summary, this work (1) performs a controlled comparison of CNN, LSTM, and multiple transformer encoders under identical preprocessing and evaluation conditions; (2) evaluates both a single-run baseline and a repeated-run protocol with statistical testing to account for performance variability; and (3) complements predictive metrics with robustness (cross-platform transfer) and efficiency indicators (training time and inference latency) to support deployment-oriented model selection.

This study hypothesises that transformer-based encoders will outperform lightweight CNN/LSTM baselines on depression-indicative text classification because they model non-local context and subtle semantics. Although this direction is broadly expected, its magnitude, stability across random seeds, and robustness under cross-platform shift are not guaranteed and are often underreported in single-split comparisons. Therefore, the main novelty of this work is not architectural invention but a deployment-relevant evaluation design: we jointly analyze baseline accuracy, repeated-run stability, cross-platform degradation, and efficiency indicators under a unified preprocessing and training protocol.

The remainder of the study is organized as follows. Section II reviews related work and highlights methodological gaps in prior comparative studies. Section III describes the datasets, preprocessing pipeline, evaluation protocols, and statistical testing. Section IV reports experimental results, including baseline comparisons, transformer variants, and cross-platform transfer. Section V discusses deployment implications, limitations, and future research directions. Section VI concludes the study.

II. Related Works

Research on depression detection from text has evolved from feature-engineered machine-learning pipelines toward deep representation learning. Early studies relied on linear classifiers and handcrafted lexical and psycholinguistic features, but such approaches often struggled to capture compositional meaning and longer-range context in mental-health–related language, especially in noisy, user-generated social-media posts where context and pragmatics are hard to capture [4]. As deep learning became standard, neural baselines such as CNN and LSTM gained popularity due to their ability to learn task-relevant representations directly from data. CNN-based models are effective at extracting local n-gram patterns for sentence- and post-level classification [8], while recurrent architectures such as LSTM better capture sequential dependencies that may reflect evolving emotional framing in text [9].

Critical synthesis and gap. Many prior depression-detection comparisons are difficult to reconcile because they (1) use different label definitions (self-report vs symptom proxy), (2) mix platform-specific preprocessing and splits that may leak user timelines, (3) report a single train–test split without seed variability, and (4) omit efficiency reporting beyond point accuracy. Consequently, it is often unclear whether reported gains reflect model capability, data preprocessing choices, or favorable randomness. Domain-adaptive transformers such as

MentalBERT are frequently reported to outperform general-purpose BERT in mental-health language, but the magnitude varies across datasets. In our controlled benchmark, MentalBERT improves mean F1 by approximately 0.052 over standard BERT under Protocol B (Table IV), supporting the practical value of domain-adaptive pretraining when computational budgets permit. In addition, statistical testing protocols are often underspecified in NLP comparisons [16].

Comparative studies and systematic reviews often report that transformer-based encoders are competitive or superior to classical baselines on mental-health text classification benchmarks, although results depend on dataset construction and evaluation protocol [15]. At the same time, practical deployment must account for computational and memory costs of attention-based models, which motivates efficiency-oriented transformer variants and compact encoders when latency or footprint is constrained [17]. In addition, domain-adaptive pretraining (e.g., MentalBERT) aims to improve sensitivity to mental-health–specific language patterns and implicit distress markers. More recently, several works also explored fine-tuning large language models for depression detection, reporting strong performance on selected social-media benchmarks [14].

A complementary research direction focuses on domain-adaptive pretraining. MentalBERT is a representative example of a transformer pretrained on mental-health–related corpora (including content from mental-health communities), to increase sensitivity to psychologically salient cues and implicit distress markers. Such domain adaptation has been reported to improve performance over general-purpose encoders in settings where the target language includes community-specific vocabulary, self-referential statements, and non-standard phrasing. Beyond domain adaptation, integrative reviews emphasize that affective expression and linguistic markers in social-media text are consistently associated with mental-health–related outcomes, supporting emotion-aware feature analysis in computational mental-health research [18].

Datasets and evaluation campaigns have also expanded substantially. Classic corpora such as RSDD and platform-specific datasets enabled large-scale depression-related text classification and benchmarking [7], [19]. More recent resources emphasize richer supervision (e.g., DSM-oriented labeling), multi-task objectives, and evaluation designs that better reflect temporal dynamics and domain shift. For example, MentalHelp provides a multi-task dataset for mental-health objectives in social media [20], while ReDSM5 introduces depression detection labels aligned with DSM-5-oriented criteria on Reddit content [21]. In addition, user-level and multimodal datasets have been proposed to capture changes in behavior and language over time, enabling evaluation beyond isolated posts [22]. Community-driven shared tasks such as CLPsych encourage standardized protocols that emphasize robustness and longitudinal modeling in realistic scenarios [23]. Expanding beyond English and beyond a single platform is also an active direction; for instance, Arabic mental-health resources such as CARMA support cross-lingual and cross-cultural studies [24].

Overall, prior work indicates that realistic deployment-oriented evaluation should consider not only in-domain accuracy but also robustness under repeated runs, cross-platform transfer, and computational constraints. However, controlled comparisons that jointly analyze predictive quality, repeated-run stability, cross-platform robustness, and practical efficiency indicators under a unified pipeline remain limited [17]. Moreover, many studies report single-split results without repeated-run variability, which complicates robust model selection for deployment-oriented settings. This motivates the comparative and deployment-oriented analysis performed in the present study, as well as the consideration of early-risk prediction setups such as CLEF eRisk, which emphasize longitudinal evidence and early detection under constrained information [25].

III. Materials and Methods

A. Data

To conduct a comparative analysis, we relied on publicly available research corpora originating from multiple social media platforms, which helps capture variation in writing style, message length, and the way depressive states may be expressed in different online communities. The following datasets were used:

1) Reddit Self-reported Depression Diagnosis (RSDD): A user-level corpus derived from Reddit timelines where users self-report a depression diagnosis, paired with matched control users. The original resource covers posts from approximately 9,000 diagnosed users (with a larger set of matched controls) [7]. For the present experiments, we derive a post-level binary classification subset by sampling posts from the corresponding user timelines under a fixed protocol and applying filtering and normalization. Because the source dataset is user-level, we enforce user-disjoint train/validation/test splits to prevent leakage, ensuring that posts from the same user appear in only one split.

2) Twitter-based depression dataset (Twitter-STMHD): A user-level benchmark resource for mental-health analysis on Twitter [19]. From this dataset, we derive a post-level experimental subset of approximately 6,500 English tweets for binary depression-indicative versus control classification after filtering and normalization. As Twitter-STMHD is provided at the user level, we construct the post-level subset by sampling tweets from user timelines while enforcing user-disjoint splits to prevent user overlap across train, validation, and test.

3) Facebook comments corpus: Following Islam et al. [26], we use a Facebook comments corpus derived from publicly accessible pages and labeled into two categories (depression-indicative vs non-depressive) under the authors' labeling procedure. The source study reports 7,145 labeled comments (4,149 depression-indicative; 2,996 non-depressive). After applying our preprocessing and filtering pipeline (e.g., removing empty/duplicate entries and non-English fragments), we retain 4,300 instances for controlled experiments while preserving the original label mapping. In addition, to keep the merged corpus prevalence consistent with the overall non-depressive majority observed across sources (Table I), we subsampled the retained Facebook data without changing label definitions. Due to platform policy and data-sharing

constraints, we report only aggregated results and do not redistribute raw Facebook text.

Overall, the unified dataset comprises 19,800 post-level instances spanning all three platforms (9,000 Reddit posts, 6,500 tweets, and 4,300 Facebook comments). We use a fixed stratified split into training, validation, and test sets (80%/10%/10%). For user-level sources (RSDD and Twitter-STMHD), splits are user-disjoint to avoid leakage across a user's timeline; for Facebook, comments are split at the instance level while preserving class balance.

TABLE I. PLATFORM-WISE CLASS DISTRIBUTION AFTER PREPROCESSING

Platform	Total (N)	Depressive, n (%)	Non-depressive, n (%)
Reddit	9,000	3,600 (40.0%)	5,400 (60.0%)
Twitter	6,500	2,400 (36.9%)	4,100 (63.1%)
Facebook	4,300	1,800 (41.9%)	2,500 (58.1%)
Total (all platforms)	**19,800**	**7,800 (39.4%)**	**12,000 (60.6%)**

Across platforms, depressive posts are less frequent overall (Table I). We tested whether class proportions differ across platforms using a chi-square test of independence and found statistically significant, but small, differences ($\chi^2(2) = 28.96$, $p = 5.14e\text{-}07$; Cramér's $V = 0.038$). These supports treating platform shift as a meaningful factor when interpreting cross-platform transfer results.

The merged corpus exhibited a moderate skew toward non-depressive posts. To mitigate potential bias during training without distorting evaluation, we applied a conservative balancing strategy after performing the train/validation/test split and only on the training split. Specifically, we used mild random under sampling of the majority class and semantic-preserving text augmentation for the minority (depressive) class, while keeping the validation and test splits unchanged. This approach increases exposure to depressive patterns during learning while preserving a realistic evaluation setting and preventing leakage from synthetic text into evaluation.

All datasets used in this study are publicly available research corpora or publicly accessible social-media text collections described by prior work and used under their respective terms and platform constraints. In our local processing, we removed or masked user handles, mentions, URLs, and other explicit identifiers where present, and we used the data strictly for aggregated modeling and evaluation rather than any individual-level inference.

B. Text Preprocessing

Social-media text is typically noisy and informal, containing emojis, URLs, mentions, hashtags, non-standard spelling, and grammatical deviations. To reduce noise while preserving emotionally informative patterns, we applied a multi-stage preprocessing pipeline designed for user-generated posts. The process included six interrelated stages (Fig. 1) that transform raw posts (raw post text) into model-ready inputs while preserving linguistically and affectively informative cues.

1) Cleaning: We removed purely technical tokens (URLs, @mentions, and obvious HTML artifacts) and normalized whitespace. Importantly, we preserved punctuation patterns that may convey affect (e.g., ellipses "...", repeated exclamation marks "!!!", and expressive repetition) when they functioned as emotional markers rather than boilerplate. Emoji handling was conservative to avoid discarding affective cues.

2) Tokenization: To ensure comparability across model families, we used word-level tokenization for CNN/LSTM baselines and subword tokenization for transformer encoders using the official pretrained tokenizers (e.g., WordPiece for BERT). This design prevents tokenization mismatch from confounding architecture comparisons.

3) Lemmatization: Lemmatization was applied conservatively with part-of-speech constraints to avoid semantic distortion in colloquial expressions. We avoided aggressive stemming that may alter psychologically relevant wording.

4) Normalization: Text was lowercased to reduce sparsity; however, we avoided aggressive stop-word removal because function words, negations, and stylistic markers can be informative in depression-related language, as emphasized in methodological reviews of mental-health prediction from social media [4]. Normalization was kept light to preserve expressive spelling patterns that may indicate emotional intensity.

5) Padding/truncation: All inputs were padded/truncated to a fixed length of 128 tokens. When truncation was required, we applied a consistent policy across models by retaining the most informative part of the sequence under a fixed-length constraint.

6) Model input: For transformer-based models, we constructed input IDs and attention masks following the standard input formatting of the corresponding pretrained tokenizer, with truncation/padding to a fixed maximum length. For CNN and LSTM, the same cleaned texts were mapped into index sequences using the baseline vocabulary and embeddings pipeline. We fixed the maximum sequence length and padding strategy consistently across transformer models to ensure comparable computational budgets.

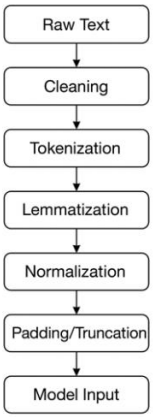


Fig. 1. Algorithmic pipeline for preprocessing textual data before neural classification.

C. Model Architectures

The dataset exhibited moderate class imbalance between depressive and non-depressive posts, which can negatively affect model generalization. Initially, the Synthetic Minority Over-sampling Technique (SMOTE) was considered for balancing the dataset [27]. SMOTE operates by interpolating minority samples in a continuous feature space; however, for sequence-based neural models that require token sequences as inputs, such interpolation does not directly yield new valid text instances. Therefore, we adopted training-only resampling and semantic-preserving text augmentation rather than relying on synthetic oversampling for the main experiments [27], [28], [29].

To mitigate class imbalance without distorting evaluation, we applied a conservative balancing strategy only on the training portion of the data after performing the train/validation/test partitioning. Specifically, we used mild random undersampling of the majority class and semantic-preserving text augmentation for the minority (depressive) class, while keeping the validation and test splits unchanged. This prevents leakage, preserves realistic prevalence during evaluation, and reduces the risk of over-interpreting a single favorable run. A systematic ablation study comparing multiple augmentation families is beyond the scope of this work and is left for future research.

To classify depressive content, we implemented and compared three widely adopted neural architectures: a convolutional model for capturing local n-gram patterns (CNN) [8], a recurrent model designed to model sequential dependencies (LSTM) [9], and a transformer-based architecture relying on bidirectional contextual encoding (BERT) [12]. Fig. 2 summarizes the compared model families at a high level.

1) Convolutional neural network: A model that captures local n-gram patterns using convolutional filters and pooling. Following common sentence-level CNN setups, we used convolution kernels of sizes 3, 4, and 5 with GlobalMaxPooling, and applied dropout (0.5) for regularization to mitigate overfitting in short-text settings. We additionally align this configuration with depression-oriented CNN pipelines reported in prior studies [30].

2) Long short-term memory: is a recurrent architecture capable of capturing temporal dependencies and context in sequences of words. The model included one bidirectional LSTM layer with a latent state dimension of 128 and an output fully connected layer.

3) Bidirectional encoder representations from transformers: A transformer-based architecture that produces bidirectional contextual representations of tokens using self-attention. In our experiments, we used the pre-trained BERT-base-uncased model and fine-tuned it on the prepared corpus for depression classification, following the standard fine-tuning protocol for BERT-style encoders.

To ensure a fair comparison, we standardized the training pipeline across models and tuned hyperparameters within controlled ranges. For transformer-based models (BERT, RoBERTa, DistilBERT, MentalBERT), we fine-tuned pretrained encoders using AdamW with a linear learning-rate scheduler and searched learning rates in {1e−5, 2e−5, 3e−5, 5e−5} and batch sizes in {16, 32}, with 2–5 epochs and early stopping based on validation loss (patience 2–3). For CNN and LSTM, we tuned the hidden size and dropout rate and selected learning rates via Bayesian optimization to balance convergence stability and generalization. Unless explicitly stated, we kept the maximum sequence length fixed at 128 tokens and used the same data split and preprocessing pipeline across all baseline experiments. This setting follows common practice in transformer fine-tuning for sentence-level classification and provides a strong efficiency–accuracy trade-off, since self-attention has quadratic complexity in sequence length. In particular, the original BERT training strategy used a sequence length of 128 for the majority of training steps to reduce computational cost, increasing to 512 only for a small portion of training.

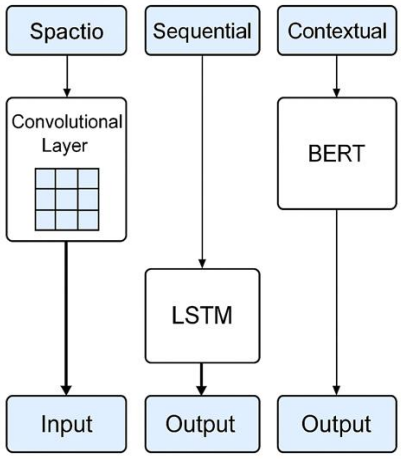


Fig. 2. Comparison of neural network architectures for text classification.

D. Evaluation Metrics

We evaluate all models using standard binary classification metrics: Accuracy, Precision, Recall, F1-score, and AUC-ROC. Together, these metrics characterize overall correctness, false-alarm behavior, and the ability to identify depressive content without missing critical cases. In mental health screening settings, Recall is particularly important because false negatives may lead to missed at-risk users, while Precision helps limit unnecessary false alarms. AUC-ROC is additionally reported as a threshold-independent indicator of class separability [31].

- Accuracy (Classification accuracy).

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Accuracy represents the proportion of correctly classified samples among all observations. It provides a general measure of model performance but may be misleading for imbalanced datasets.

- Precision (Positive Predictive Value)

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

Precision indicates the proportion of true positive predictions among all predicted positives. High precision is essential when minimizing false alarms is critical.

- Recall (Sensitivity)

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

Recall measures the model's ability to identify all relevant positive instances. In mental health detection tasks, high recall is vital to avoid missing cases that require attention.

- F1-score (Harmonic mean of precision and recall)

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

F1-score balances precision and recall, providing a single measure of model effectiveness. It is particularly useful when both false positives and false negatives are costly.

- AUC (Area Under the ROC Curve)

$$AUC = \int_0^1 TPR(FPR)d(FPR) \quad (5)$$

where:

$TPR = \frac{TP}{TP+FN}$ – True Positive Rate (sensitivity),

$FPR = \frac{FP}{FP+TN}$ – False Positive Rate.

AUC measures the area under the Receiver Operating Characteristic (ROC) curve, showing the model's ability to discriminate between classes. A higher AUC indicates stronger separability.

E. Evaluation Protocols

We report results under two complementary protocols. Protocol A (single-run baseline) reports performance on a fixed held-out split using one training run per model configuration (see Table III and Fig. 3–5). Protocol B (repeated runs) evaluates robustness by training each configuration five times with different random seeds while keeping the same split and preprocessing pipeline fixed; results are summarized as mean ± standard deviation in Table IV. Repeated-seed reporting is important because fine-tuning outcomes can vary due to initialization, minibatch order, and early-stopping dynamics. Therefore, we report seed-averaged performance to reduce the risk of over-interpreting a single favorable run and to provide a more stable basis for model comparison [32], [33]. This separation prevents mixing single-run snapshots with stability-oriented reporting and improves reproducibility.

F. Statistical Analysis

To assess whether observed performance differences between models were statistically meaningful, we tested model comparisons on Protocol B per-run F1 scores (n = 5 seeds per model). We first applied one-way ANOVA over per-run F1 to test the global null hypothesis of equal means. For planned pairwise localization, we then performed paired t-tests on seed-matched runs comparing BERT against each lightweight baseline (CNN and LSTM). To control family-wise error under these two planned comparisons, we applied Holm's step-down adjustment. Alongside p-values, we report paired effect sizes using Cohen's d_z, computed as the mean of paired differences divided by the standard deviation of paired differences. As a non-parametric sensitivity check for small n, we additionally report Wilcoxon signed-rank tests [34], but we interpret them cautiously because n = 5 offers limited power.

IV. Experimental Results

A. Data Preparation

For the experiments, we used the combined corpus described in the "Data" subsection of MATERIALS AND METHODS. It consists of curated subsets from Reddit, Twitter, and Facebook datasets and totals approximately 19,800 texts after preprocessing and deduplication. The detailed dataset sources, subset sizes, and selection criteria are provided in the main "Data" description to ensure clarity and reproducibility.

B. Model Training

Table II summarizes the training setup and computational characteristics of the baseline architectures under identical preprocessing and a fixed train/test split. CNN and LSTM were optimized using Adam, while transformer-based encoders were fine-tuned using AdamW with a linear learning-rate scheduler, following standard practice for BERT-style models. All models used binary cross-entropy loss for the binary classification objective. All experiments were conducted in Google Colab using an NVIDIA Tesla T4 GPU. We recorded the runtime environment (Python, CUDA, PyTorch/TensorFlow, NumPy/SciPy, scikit-learn, and Hugging Face Transformers) and the full configuration (random seeds, batch size, learning rate, early stopping, and hyperparameter ranges) to support reproducibility.

TABLE II. Training Setup and Computational Characteristics of the Baseline Architectures

Model	Framework	Key Parameters	Training Time (GPU)
CNN	TensorFlow/Keras	128 filters, kernel={3,4,5}, dropout=0.3	~0.3 h
LSTM	TensorFlow/Keras	128 units, dropout=0.3	~0.4 h
BERT (base-uncased)	Hugging Face Transformers	lr=2e-5, batch=16, epochs=3	~3.4 h

C. Results and Comparison

After training, each model's performance was evaluated using standard classification metrics (Accuracy, Precision, Recall, F1-score, and AUC). The results of calculating metrics for each architecture are shown in Table III.

TABLE III. Comparison of Classification Metrics Across Baseline Models

Metric	CNN	LSTM	BERT
Accuracy	0.820	0.870	**0.897**
Precision	0.800	0.860	**0.887**
Recall	0.790	0.880	**0.892**
F1-score	0.795	0.870	**0.889**
AUC	0.850	0.910	**0.943**

Analysis of the table shows that all three architectures have demonstrated stable results, but performance differs across architectures in a consistent manner.

The CNN model showed satisfactory Accuracy (0.820) and AUC (0.850) values, which confirms its ability to isolate local text patterns and n-grams. However, the lower Recall (0.790) and F1-score (0.795) scores indicate that CNN has limited capabilities in recognizing contextual dependencies characteristic of natural language.

The LSTM model, due to its recurrent structure, demonstrated a deeper understanding of sequential dependencies and improved all metrics compared to CNN. In particular, Recall increased to 0.880, and AUC increased to 0.910, which indicates a higher ability of the model to detect depressive texts without a significant increase in false positives.

The BERT model showed the highest results, reaching the maximum values Accuracy = 0.897, Recall = 0.892, F1 = 0.889 and AUC = 0.943. This result is explained by the use of the self-attention mechanism, which allows the model to take into account the context of the entire sentence, not just neighboring words. This is especially important when analyzing emotionally laden statements, where the meaning may depend on complex contextual relationships.

A high Recall value (0.892) indicates the minimum number of missed cases of depressive texts, which is of key importance for monitoring the psycho-emotional state of users on social networks. Concurrently, high Precision (0.887) indicates a low level of false alarms, which makes the BERT model the most balanced and reliable solution among the tested architectures.

BERT surpasses the traditional CNN and LSTM architectures in all major metrics, especially in F1 and AUC, which confirms its high ability for contextual text analysis.

LSTM remains an acceptable option for scenarios with limited computing resources, providing a good balance of accuracy and speed.

CNN, despite its relatively low completeness (Recall), can be used for basic or mobile applications where speed and ease of implementation are critical.

Thus, the BERT model has demonstrated the best balance between accuracy, completeness, and the ability to distinguish classes, which makes it preferable for practical systems for analyzing depressive content on social networks.

Fig. 3–5 provide complementary visual evidence for the baseline comparison in Table III, including a metric profile (Fig. 3), ROC curves across thresholds (Fig. 4), and confusion matrices that summarize error types (Fig. 5).

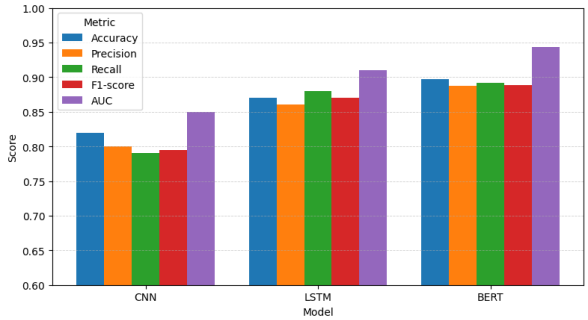


Fig. 3. Comparison of classification metrics across models.

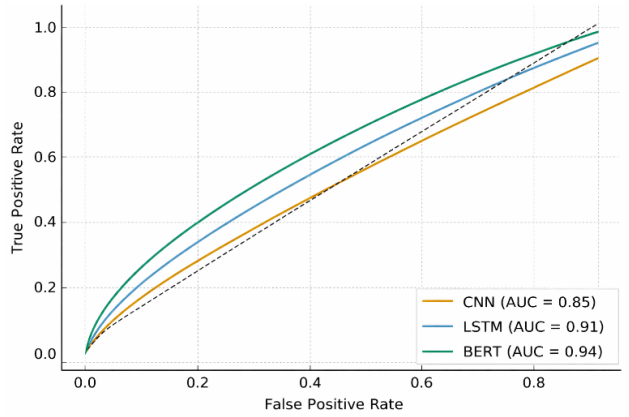


Fig. 4. ROC curves for each architecture.

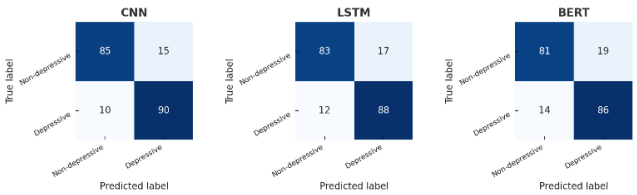


Fig. 5. Confusion matrices illustrating classification performance.

D. Statistical Analysis

To assess whether the observed differences in predictive quality are statistically meaningful, we performed significance testing on per-run F1 scores obtained under Protocol B (five independent runs with different random seeds, n = 5 per model). We first applied one-way ANOVA across the three baseline models (CNN, LSTM, BERT) and obtained $F(2, 12) = 33.73$, $p = 1.19e-05$, indicating that at least one model differs in mean F1 under repeated training.

To localize pairwise differences while limiting multiple comparisons, we performed planned post-hoc paired t-tests comparing BERT to each lightweight baseline using seed-matched runs, reporting Holm-adjusted p-values and paired effect sizes (Cohen's d_z). BERT outperforms CNN ($p = 0.004$; Holm-adjusted $p = 0.008$; $d_z = 2.64$) and outperforms LSTM ($p = 0.017$; Holm-adjusted $p = 0.017$; $d_z = 1.76$). In addition, LSTM outperforms CNN in these runs ($p = 0.003$; $d_z = 3.00$). Overall, the results suggest that the performance advantages of contextual modeling are unlikely to be explained solely by random initialization under the evaluated setting.

E. Summary of Findings

In the baseline comparison, transformer-based encoding (BERT) achieved the most favorable overall metric profile, including a strong F1-score and AUC, indicating improved class separability in depression-related text.

LSTM provided a competitive non-transformer baseline and generally outperformed CNN, consistent with improved modeling of sequential context and longer dependencies.

CNN remained computationally efficient but showed weaker performance on metrics that are sensitive to subtle semantics, suggesting limitations for nuanced mental-health language.

Statistical testing on repeated runs (Protocol B) indicates that the performance differences involving BERT are unlikely to be due to random variation alone, supporting the robustness of the observed ranking.

F. Enabling and Testing New Transformers: RoBERTa, DistilBERT, MentalBERT

To extend the baseline comparison beyond CNN/LSTM/BERT, we evaluated additional transformer-family encoders that represent different trade-offs in pretraining objectives, parameter efficiency, and domain adaptation. Specifically, we considered RoBERTa, which improves upon BERT-style pretraining by removing the next-sentence prediction objective and leveraging larger training corpora [36]; DistilBERT, a compressed variant designed to reduce computation with minimal loss in accuracy [37]; and MentalBERT, a domain-adapted transformer pretrained on mental-health–related corpora, aimed at improving sensitivity to psychologically salient cues [13]. Importantly, MentalBERT retains the BERT-base architecture and WordPiece tokenization; thus, improvements are attributable to domain-adaptive pretraining rather than tokenization changes. A detailed embedding-space similarity analysis (e.g., representation drift between BERT and MentalBERT) is left for future work.

Results are summarized in Table IV under Protocol B, where each model configuration is trained five times with different random seeds on the same fixed split, and the reported values correspond to mean ± standard deviation across runs. For clarity, Table III presents Protocol A as a single-run baseline on the same split; therefore, Table III provides a point-performance snapshot rather than variability estimates. Accordingly, mean ± standard deviation is reported only for Protocol B (Table IV), while Protocol A (Table III) is presented as a single-run baseline snapshot. This two-protocol design complements the single-run snapshot (Protocol A) with variability-aware evaluation (Protocol B), helping characterize stability and reduce sensitivity to initialization effects.

All classification metrics in Tables III and IV are reported in fractional form (0-1). Percent values are used only where explicitly indicated (e.g., Table V). Table IV presents a comparative view of predictive quality (Accuracy, Precision, Recall, F1-score, AUC) together with practical deployment indicators (training time, inference latency, and model size). Overall, RoBERTa demonstrates competitive performance among general-purpose transformer variants, consistent with stronger pretraining on larger corpora. DistilBERT typically shows a small reduction in predictive quality while offering noticeable gains in computational efficiency, making it a reasonable candidate for resource-constrained scenarios. MentalBERT achieves strong performance in this setting, which aligns with the expected benefit of domain-adaptive pretraining for mental-health language, where subtle cues, slang, and indirect phrasing can be critical for correct classification.

Batching throughput note. Let ℓ denote the batch=1 inference latency (ms/sample) reported in Table IV. For a batch size b, the achievable throughput lies between a conservative lower bound of $1000/\ell$ samples/s (no batching speedup) and an optimistic upper bound of $b \cdot 1000/\ell$ samples/s (ideal batching), with actual throughput depending on GPU memory bandwidth, kernel fusion, and framework overheads. Using the Table IV latencies, the corresponding ranges are: BERT (baseline): b=1 59.5; b=8 59.5–476.2; b=16 59.5–952.4; b=32 59.5–1904.8 samples/s. RoBERTa-base: b=1 54.9; b=8 54.9–439.6; b=16 54.9–879.1; b=32 54.9–1758.2 samples/s. DistilBERT: b=1 106.4; b=8 106.4–851.1; b=16 106.4–1702.1; b=32 106.4–3404.3 samples/s. MentalBERT: b=1 50.8; b=8 50.8–406.1; b=16 50.8–812.2; b=32 50.8–1624.4 samples/s.

Efficiency note (throughput and FLOPs). Using the reported batch-1 inference latency, the corresponding batch-1 throughput is approximately 59.5 samples/s (BERT), 54.9 samples/s (RoBERTa), 106.4 samples/s (DistilBERT), and 50.8 samples/s (MentalBERT). Throughput typically increases under batching up to hardware saturation, but the exact gain is device- and implementation-dependent; therefore, we report latency and derived batch-1 throughput as comparable baselines. We also report approximate per-sample forward-pass compute for transformers at sequence length 128 using standard FLOP accounting: a 12-layer BERT/RoBERTa/MentalBERT encoder is on the order of ~22 GFLOPs per sample, while DistilBERT (6 layers) is ~11 GFLOPs. These estimates help explain why DistilBERT provides a favorable efficiency–accuracy trade-off, whereas domain-adaptive models may be preferred when accuracy dominates and compute budgets allow. At small batch sizes, runtime can also be influenced by memory bandwidth and kernel launch overheads; hence, FLOP estimates should be interpreted as approximate compute proxies rather than exact predictors of latency.

G. Qualitative Error Analysis (False Positive / False Negative)

We performed a qualitative inspection of misclassified examples to identify common sources of false positives (FP) and false negatives (FN) across architectures. A recurring pattern for FP errors is the presence of negatively valenced language that expresses temporary frustration or situational stress without clear indicators of persistent depressive symptoms. In such cases, strong sentiment words dominate the surface signal, while the broader self-referential and persistent framing that is more typical of depressive discourse is absent.

TABLE IV. PERFORMANCE COMPARISON OF TRANSFORMER-BASED MODELS FOR DEPRESSIVE CONTENT CLASSIFICATION

Model	Accuracy	Precision	Recall	F1-score	AUC	Training time (hrs)	Inference latency (ms/sample)	Model size (MB)
BERT (baseline)	0.897 ± 0.004	0.887 ± 0.004	0.892 ± 0.004	0.889 ± 0.004	0.943 ± 0.003	3.4 ± 0.2	16.8 ± 0.5	418
RoBERTa-base	0.912 ± 0.003	0.904 ± 0.003	0.917 ± 0.003	0.910 ± 0.003	0.956 ± 0.002	3.8 ± 0.2	18.2 ± 0.5	473
DistilBERT	0.884 ± 0.006	0.875 ± 0.006	0.881 ± 0.006	0.878 ± 0.006	0.938 ± 0.003	2.1 ± 0.1	9.4 ± 0.3	265
MentalBERT	0.920 ± 0.005	0.913 ± 0.005	0.924 ± 0.005	0.918 ± 0.005	0.962 ± 0.002	4.0 ± 0.2	19.7 ± 0.6	475

False negatives often arise from indirect, understated, or context-dependent expressions of distress, including implicit hopelessness, short, fragmented statements, and multi-clause constructions where the depressive cue is subtle. Another common source of errors is negation and contrast (e.g., "I'm not sad anymore"), where surface-level sentiment may be misleading. In these scenarios, classical architectures such as CNN and LSTM more frequently fail to resolve long-range dependencies or pragmatic intent, whereas transformer-based models tend to be more resilient due to contextual encoding—although pragmatic ambiguity, figurative language, and complex discourse structure remain challenging even for contextual encoders.

Overall, the error analysis suggests that robust depression-related text classification benefits from strong contextual modeling, but additional improvements may require explicit handling of negation, pragmatics, and user-level temporal context, especially when transferring across platforms and writing styles.

H. Cross-Validation and Cross-Platform Evaluation of Model Robustness

To assess generalization and stability beyond a single dataset, we evaluated model robustness using two complementary settings: within-dataset cross-validation and cross-platform transfer. Within-dataset cross-validation estimates how stable model performance is when training and testing are performed on different folds of the same corpus. Cross-platform transfer evaluates domain shift by training on one platform-specific dataset and testing on another, which better reflects real-world variability in vocabulary, style, and topic distribution.

Cross-platform transfer results in Table V indicate a consistent performance drop for all models relative to in-domain evaluation, highlighting a non-trivial domain shift between Reddit, Twitter, and Facebook. For within-domain validation, we conduct 5-fold stratified cross-validation on the merged Reddit+Twitter corpus and report the mean F1-score across folds. For cross-platform transfer, we (1) train on Reddit and evaluate on Twitter and (2) train on Twitter and evaluate on Reddit without target-domain fine-tuning. Additionally, to assess transfer to Facebook under a different discourse style, we train on the combined Reddit+Twitter corpus and evaluate on Facebook without collecting new Facebook data. Table V reports F1-scores for each setting.

TABLE V. CROSS-VALIDATION AND CROSS-PLATFORM PERFORMANCE OF NEURAL ARCHITECTURES (MEAN F1-SCORE, %)

Model	5-Fold CV (Reddit+Twitter) F1 (%)	Reddit→Twitter F1 (%)	Twitter→Reddit F1 (%)	Reddit+Twitter→Facebook F1 (%)
CNN	82.8	79.3	80.1	78.6
LSTM	86.7	82.6	83.4	82.0
BERT	92.4	86.8	87.5	85.9
RoBERTa	93.3	88.1	88.9	87.4
DistilBERT	90.4	84.9	85.7	84.1
MentalBERT	**94.0**	**89.3**	**90.1**	**88.8**

I. Ethical Considerations and Data Protection

This study uses research corpora described in prior work and accessed under their respective terms (public releases where available, or research-accessible benchmarks subject to platform and sharing constraints). We follow established internet-research ethics guidance and health-focused social-media research protocols that emphasize contextual integrity, risk assessment, and harm minimization when working with online traces in sensitive domains such as mental health. Our analysis is conducted at an aggregated level for methodological research rather than individual-level profiling or intervention. Where present, we remove or mask explicit identifiers (e.g., user handles, mentions, and URLs) during local processing, and we avoid reporting verbatim user-generated text; when illustrative examples are necessary, they are paraphrased to reduce re-identification risk and unintended disclosure [38], [39], [40].

Because mental-health–related language can be sensitive, we emphasize that model outputs should not be interpreted as clinical diagnoses. Potential harms include false positives and false negatives, which may lead to inappropriate concern or missed cases if deployed without safeguards. Therefore, any practical use should include human oversight, careful threshold calibration, and evaluation on the target domain and language before deployment [40]. We also recommend additional validation across platforms and demographic contexts to reduce bias and improve robustness.

We acknowledge that language models may inherit demographic, cultural, or platform-specific biases present in the underlying corpora. Therefore, fairness-sensitive evaluation and cross-domain validation are necessary before operational use. Any deployment in mental-health–related settings should incorporate human oversight, transparent reporting, and risk-mitigation procedures appropriate for sensitive decisions.

V. DISCUSSION

A. Main Findings and Interpretation

This study compared baseline neural architectures (CNN, LSTM, and BERT) [35] and several transformer variants (RoBERTa, DistilBERT, and MentalBERT) for classifying depression-related language in social-media text. Across the baseline comparison (Table III), BERT achieved the strongest overall balance in predictive quality, particularly in F1-score and AUC. This result is consistent with the advantage of contextual self-attention, which supports modeling non-local dependencies

and subtle semantic cues that frequently appear in user-generated mental-health narratives.

CNN showed the weakest performance among the three baselines. While convolutional filters can capture discriminative local patterns, the model is limited in representing long-range dependencies and context interactions that are important for distinguishing transient negative sentiment from depression-related expression. LSTM provided an intermediate solution: its recurrent structure better captures sequential context than CNN, improving Recall and AUC, yet it remained below transformer-based encoders in the main metrics. Importantly, these findings reinforce the practical view that model choice should be aligned with the intended deployment constraints: CNN and LSTM remain attractive when compute and latency budgets are tight, whereas transformers are preferable when higher screening sensitivity and overall separability are required.

The robustness-oriented experiments further clarify trade-offs among transformer variants (Tables III–IV). RoBERTa showed strong predictive performance as a general-purpose transformer, while DistilBERT provided a favorable efficiency profile with reduced inference latency and smaller model size at a moderate performance cost. MentalBERT achieved the highest quality among the tested variants in our setting, which is consistent with the expected benefit of domain-adaptive pretraining for mental-health language, where indirect phrasing, self-referential statements, and psychological terminology can be informative. At the same time, performance differences across variants should be interpreted together with computational indicators: in deployment-oriented scenarios, the "best" model depends on whether the priority is maximum predictive quality, minimal latency, or memory footprint. A plausible explanation is that domain-adaptive pretraining exposes the encoder to mental-health–specific vocabulary, self-referential patterns, and community discourse that are underrepresented in general-domain corpora. This can improve sensitivity to subtle distress cues that are difficult to capture with shallow n-gram features or purely sequential baselines.

The cross-validation and cross-platform results (Table V) highlight the impact of domain shift. Performance typically decreases when transferring across platforms, reflecting differences in writing style, slang, message length, and topic distribution. For example, for BERT, the within-domain cross-validation F1 is 92.4%, while transfer yields 86.8% (Reddit→Twitter) and 87.5% (Twitter→Reddit), corresponding to drops of 4.9 – 5.6 F1 points; transfer to Facebook is 85.9% (drop 6.5 points). The transformer family generally exhibited stronger robustness under transfer than CNN/LSTM, suggesting that contextual representations can reduce sensitivity to lexical variability. However, non-trivial domain shift remains a challenge, and platform-specific adaptation, threshold calibration, or additional domain-relevant pretraining may be necessary for stable real-world performance. These results suggest that strong in-domain performance alone is insufficient for deployment-oriented use cases and that robustness under domain shift should be treated as a first-class evaluation criterion.

B. Limitations

Several limitations should be noted. First, our experiments use post-level samples derived from user-level public corpora, and specific sampling, filtering, and deduplication decisions may affect absolute performance values across platforms. Second, labels in social-media datasets represent operational definitions from prior studies and are imperfect proxies for clinical diagnosis; they may contain label noise and self-report bias, and model outputs should be interpreted as research-oriented risk signals rather than diagnostic conclusions. Third, class imbalance in real deployments may differ from both the merged corpus and our fixed evaluation split; while we applied balancing only on the training split, we did not conduct a full probability calibration study (e.g., reliability curves or ECE), which is important for threshold-based triage settings. Fourth, CNN and LSTM baselines were implemented as standard lightweight architectures and are not strictly capacity-matched to transformer encoders; stronger non-transformer baselines (deeper CNNs, larger BiLSTMs, or modern efficient transformers) are a useful extension. Fifth, platform policies and data-sharing constraints limit redistribution of raw text and may restrict independent instance-level verification; accordingly, we report results at an aggregated level. Sixth, the unified corpus is English-only, and cross-lingual generalization remains an important direction for future work. Finally, although repeated-run reporting improves robustness, the number of runs ($n = 5$) is modest; future studies could extend variability analysis across additional seeds, report calibration under prevalence shift, and explore efficient deployment techniques such as quantization, pruning, and distillation under realistic batching and hardware constraints.

VI. Conclusion

This study presented a controlled comparison of CNN, LSTM, and transformer encoders for depression-related text classification on a unified multi-platform social-media corpus. Using a single-run baseline together with repeated-run evaluation and statistical testing, we found that transformer models, especially a domain-adapted encoder, deliver the strongest predictive performance and more stable behavior under cross-platform transfer. At the same time, transfer experiments showed a consistent degradation across platforms, indicating that deployment-oriented evaluation must go beyond in-domain accuracy and explicitly account for domain shift and robustness. Finally, our efficiency analysis highlights practical trade-offs between quality and computational cost, enabling informed model choice for real-world research pipelines. Future work will extend evaluation to additional languages and platforms, incorporate longitudinal user-level signals, and study calibration and fairness-sensitive safeguards for responsible use.

Acknowledgment

The authors thank colleagues and collaborators for helpful feedback on earlier drafts and the research community for releasing public resources that enable reproducible computational mental-health research. No external funding was received for this study.

REFERENCES

[1] World Health Organization, "Depressive disorder (depression)," Fact sheet, Aug. 29, 2025. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/depression (accessed Jan. 2026).

[2] M. De Choudhury, M. Gamon, S. Counts, and E. Horvitz, "Predicting Depression via Social Media," Proc. Int. AAAI Conf. Web and Social Media (ICWSM), pp. 128–137, 2013, doi: 10.1609/icwsm.v7i1.14432.

[3] A.-M. Bucur, I. R. Podină, and L. P. Dinu, "A psychologically informed part-of-speech analysis of depression in social media," arXiv:2108.00279, 2021, doi: 10.48550/arXiv.2108.00279.

[4] G. Chancellor and M. De Choudhury, "Methods in predictive techniques for mental health status on social media: A critical review," npj Digital Medicine, vol. 3, no. 1, pp. 1–11, 2020, doi: 10.1038/s41746-020-0253-0.

[5] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, and P. S. Yu, "A survey on text classification: From shallow to deep learning," arXiv:2008.00364, 2020, doi: 10.48550/arXiv.2008.00364.

[6] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep Learning–based Text Classification: A Comprehensive Review," ACM Computing Surveys, vol. 54, no. 3, pp. 1–40, 2021, doi: 10.1145/3439726.

[7] A. Yates, A. Cohan, and N. Goharian, "Depression and Self-Harm Risk Assessment in Online Forums," in Proc. EMNLP 2017, pp. 2968–2978, 2017, doi: 10.18653/v1/D17-1321.

[8] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in Proc. EMNLP 2014, pp. 1746–1751, 2014.

[9] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," IEEE Trans. Neural Netw. Learn. Syst., vol. 28, no. 10, pp. 2222–2232, Oct. 2017, doi: 10.1109/TNNLS.2016.2582924.

[10] I. Tavchioski, M. Robnik-Šikonja, and S. Pollak, "Detection of depression on social networks using transformers and ensembles," arXiv:2305.05325, 2023, doi: 10.48550/arXiv.2305.05325

[11] A. Vaswani et al., "Attention Is All You Need," in Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 30, pp. 5998–6008, 2017, doi: 10.48550/arXiv.1706.03762.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT 2019, pp. 4171–4186, 2019, doi: 10.18653/v1/N19-1423.

[13] S. Ji et al., "MentalBERT: Publicly available pretrained language models for mental healthcare," arXiv:2110.15621, 2021, doi: 10.48550/arXiv.2110.15621.

[14] S. Munir, S. A. Gillani, M. S. A. Baig, M. A. Saleem, and H. Siddiqui, "Advancing Depression Detection on Social Media Platforms through Fine-Tuned Large Language Models," arXiv:2409.14794, 2024, doi: 10.48550/arXiv.2409.14794.

[15] R. W. Leiva, A. Martín, and A. González, "Deep learning for depression detection from text: A comparative analysis," Expert Systems with Applications, vol. 183, p. 115376, 2021, doi: 10.1016/j.eswa.2021.115376.

[16] R. Dror, G. Baumer, S. Shlomov, and R. Reichart, "The Hitchhiker's Guide to Testing Statistical Significance in Natural Language Processing," in Proc. ACL 2018, pp. 1383–1392, doi: 10.18653/v1/P18-1128.

[17] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient Transformers: A Survey," ACM Computing Surveys, vol. 55, no. 6, Art. no. 109, 2022, doi: 10.1145/3530811.

[18] S. C. Guntuku, D. B. Yaden, M. Kern, L. H. Ungar, and H. A. Schwartz, "Detecting depression and mental illness on social media: An integrative review," Current Opinion in Behavioral Sciences, vol. 18, pp. 43–49, 2017, doi: 10.1016/j.cobeha.2017.07.005.

[19] Suhavi et al., "Twitter-STMHD: An extensive user-level database of multiple mental health disorders," in Proc. ICWSM, vol. 16, pp. 1182–1191, 2022. [Online]. Available: https://ojs.aaai.org/index.php/ICWSM/article/view/19368

[20] N. Raihan, S. S. C. Puspo, S. Farabi, A.-M. Bucur, T. Ranasinghe, and M. Zampieri, "MentalHelp: A Multi-Task Dataset for Mental Health in Social Media," in Proc. LREC-COLING 2024, Torino, Italy, May 2024, pp. 11196–11203.

[21] E. Bao, A. Pérez, and J. Parapar, "ReDSM5: A Reddit Dataset for DSM-5 Depression Detection," arXiv:2508.03399, Aug. 2025. [Online]. Available: https://arxiv.org/abs/2508.03399

[22] B. Wang, Y. Sun, Y. Zhao, and B. Qin, "Beyond Snapshots: A Multimodal User-Level Dataset for Depression Detection in Dynamic Social Media Streams," in Proc. ACM Multimedia (MM '25), Oct. 2025, doi: 10.1145/3746027.3758236.

[23] T. Tseriotou et al., "Overview of the CLPsych 2025 Shared Task: Capturing Mental Health Dynamics from Social Media Timelines," in Proc. CLPsych 2025, May 2025, pp. 193–217.

[24] S. Mankarious and A. Zirikly, "CARMA: Comprehensive Automatically-annotated Reddit Mental Health Dataset for Arabic," arXiv:2511.03102, Nov. 2025. [Online]. Available: https://arxiv.org/abs/2511.03102

[25] "eRisk: Early Risk Prediction on the Internet – CLEF 2025," CLEF 2025 Labs. [Online]. Available: https://clef2025.clef-initiative.eu/index.php?page=Pages/Labs/eRisk.html (accessed Dec. 16, 2025).

[26] M. R. Islam, M. A. Kabir, A. Ahmed, A. R. M. Kamal, H. Wang, and A. Ulhaq, "Depression detection from social network data using machine learning techniques," Health Information Science and Systems, vol. 6, no. 1, Art. no. 8, 2018, doi: 10.1007/s13755-018-0046-0.

[27] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, Learning from Imbalanced Data Sets. Springer, Cham, 2018, doi: 10.1007/978-3-319-98074-4.

[28] S. Y. Feng et al., "A Survey of Data Augmentation Approaches for NLP," in Findings of ACL-IJCNLP 2021, pp. 968–988, 2021, doi: 10.18653/v1/2021.findings-acl.84.

[29] E. Ma, "nlpaug: NLP augmentation library," GitHub repository, 2019. [Online]. Available: https://github.com/makcedward/nlpaug (accessed Dec. 2025).

[30] H. E. Tay, M. K. Lim, and C. Y. Chong, "SERCNN: Stacked Embedding Recurrent Convolutional Neural Network in Detecting Depression on Twitter," arXiv:2207.14535, 2022, doi: 10.48550/arXiv.2207.14535.

[31] M. Hossin and M. N. Sulaiman, "A Review on Evaluation Metrics for Data Classification Evaluations," Int. J. Data Mining & Knowledge Management Process, vol. 5, no. 2, pp. 1–11, Mar. 2015, doi: 10.5121/ijdkp.2015.5201.

[32] N. Reimers and I. Gurevych, "Reporting Score Distributions Makes a Difference: Performance Study of LSTM-networks for Sequence Tagging," in Proc. EMNLP 2017, pp. 338–348, 2017, doi: 10.18653/v1/D17-1035.

[33] J. Dodge et al., "Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping," arXiv:2002.06305, 2020. [Online]. Available: https://arxiv.org/abs/2002.06305

[34] M. Hollander, D. A. Wolfe, and E. Chicken, Nonparametric Statistical Methods, 3rd ed. Hoboken, NJ, USA: Wiley, 2013.

[35] R. Dror, S. Shlomov, and R. Reichart, "Deep Dominance - How to Properly Compare Deep Neural Models," in Proc. ACL 2019, pp. 2773–2785, 2019, doi: 10.18653/v1/P19-1266.

[36] Y. Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," arXiv:1907.11692, 2019. [Online]. Available: https://arxiv.org/abs/1907.11692

[37] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," arXiv:1910.01108, 2019. [Online]. Available: https://arxiv.org/abs/1910.01108

[38] A. S. Franzke, A. Bechmann, C. M. Ess, and M. Zimmer (Eds.), Internet Research: Ethical Guidelines 3.0. Association of Internet Researchers (AoIR), 2019.

[39] L. Townsend and C. Wallace, Social Media Research: A Guide to Ethics. University of Aberdeen, 2016.

[40] A. Benton, G. Coppersmith, and M. Dredze, "Ethical Research Protocols for Social Media Health Research," in Proc. EthNLP (ACL Workshop), Valencia, Spain, 2017, pp. 94–102, doi: 10.18653/v1/W17-1612.

Attention-Based Capsule Network with Vision Transformer for Underwater Hyperspectral Image Classification

Thiyagarajan B, Thenmozhi M
Department of Computer Science and Engineering, Puducherry Technological University, Puducherry, India

***Abstract*—The underwater investigations and research remain challenging due to various underwater distortion factors, scattering, and low wavelength absorption. Hyperspectral imaging helps in obtaining detailed information on each underwater object using the spectral reflectance, using 100 to 300 bands. The Hyperspectral imaging in underwater applications contains the 3D hyperspectral cube, which needs a high level of processing that results in high-accuracy classification. Hence, this study proposes the framework of hybrid deep learning techniques that work on segmentation, feature extraction, and classification processes. The Channel Attention Module (CAM) based U-Net architecture is used for the segmentation process to obtain the spectral spatial characteristics based on the Region of Interest (ROI). The CapsNet Feature extraction helps in obtaining the features of various bands, which helps in the classification of object class-wise using the pose-based relationships. The Vision Transformer (ViT) based classification that depends on the capsule vector token, carries out the multi-class classification by obtaining the global attention among the feature vectors and relationship-based long-range ROI feature data. In this way, the proposed model attains 95.3% accuracy using the maximum IoU of 0.88 and 95.2% of the segmentation process, which helps in achieving 0.99AUC for 8 substrate classes of the underwater HSI dataset.**

***Keywords*—Underwater imaging; hyperspectral; U-Net; Vision Transformer; CapsNet; classification; segmentation**

I. Introduction

Marine exploration, which is 70% part of the Earth's surface, remains slow and error-prone using the data collected from sources like AUVs, underwater sensors and cameras, ROVs, etc., which are used in domains like marine ecology, environmental monitoring, seabed mapping, aquaculture inspection, etc. The visibility in the underwater images are degarded due to the factors [1] like turbidity, low light, scattering, and colour distortion, which makes the visual inspection and interpretation a challenging task where the conventional systems completely rely on the RGB and multispectral sensors [2] that contain less spectral information. The distinguishing process of various marine materials [3] becomes a challenging task where the involvement of the advanced digital imaging techniques [4][5] helps to perceive the structures that have different spectral signatures too [6][7].

Hyperspectral imaging helps in the material identification process using the different spectral bands that capture the information using continuous reflectance, which helps in obtaining the discriminative features of spectral and spatial that help the digital image classification techniques [9]. The reviews provide an overview of the role of hyperspectral imaging in detecting coral species, geological substrates, and the chemical composition of seafloor ecosystems, among other applications, using different spectral bands that enhance both visual inspection and digital imaging techniques [10]. In the RGB imaging techniques, most of the marine materials are inspected as similar structures where the differentiation is effective using the hyperspectral signatures that classify the underwater objects and vegetation types [10].

Underwater, the absorption of red and near infrared wavelengths is rapid, which compresses the effectiveness of its spectral range, leading to continuous reflectance and results in poor signal quality at deeper zones in the sea, and also distorted reflected profiles occur due to the suspended particles that produce the spectral signatures inconsistent. The authors in [2] and [6] enlisted the main factors like depth, salinity, turbidity, chlorophyll concentration, and particulate matter illuminations as the reasons for the spectral nonlinearity that leads to the shift of scenes across the domains. In particular, when the annotated underwater data are insufficient, the introduction of 100-300spectral bands increases the risk of overfitting and computation complexity of the system that ends in an imbalanced dataset. In order to deal with deep neural networks, the algorithm must train with large and diverse information to generalise, where the limited samples are labelled, which makes the data discrepancy in the digital image classification process [7].

The characteristics of the underwater structures are non-rigid shapes, varying textures, and occlusion, which are difficult to visually distinguish from the background substrates that complicate the spectral and spatial classification, considering the intra-class variations and inter-class similarities [4][5]. Convolutional Neural Networks (CNN) are a strong contestant encountered in underwater object recognition and tracking [3], coral segmentation [4], seaweed detection [5], etc., where the local spatial features are given more importance with no information on global attentions. On the other hand, machine learning techniques are majorly dependent on the handcrafted features of hyperspectral image characteristics, where the performance is less when handling larger hyperspectral cubes under high noise underwater conditions [10] [11].

The sonar and RGB-based images lack the fine-grained material discrimination information, which cannot replace the high spectral resolutions of the hyperspectral images [1]. In the

existing literature, the involvement of transformer and attention-based models is used for RGB and traditional HSI dataset that depends on the global relationships [3], where it requires large datasets and strong prior training and finds difficulty with specific spectral distortions like absorption-induced band attenuation [1]. In the underwater HSI, the capturing of images despite the movement of seagrass and light absorption by water that affects infrared bands are encountered by hyperspectral acquisition process where the time variation in the hyperspectral cube and the orientation of tripods are done for movement of grass and to combat the IR bands issues the arrays of LED are used [12] however the method stands as the preliminary insights.

Therefore, the underwater HSI classifications need to develop the spectral distortion aware models, which requires the spectral spatial attention mechanism with a high level of feature learning process. Hence, the proposed work uses the channel attention U-Net for segmentation, CapsNet for Feature learning and Vision Transformer for Classification of underwater HSI.

The rest of the study is organized as follows: Section II: A survey on the various segments of underwater HSI, CapsNet feature extraction, and various transformer models; Section III: A detailed workflow of the proposed framework pipeline process; Section IV: Various results and discussions; and Section V: Conclusion and future work.

II. Related Works

A. Survey on Segmentation of Underwater HSI

In [13], the binary segmentation process is carried out using the Keras-U-Net architecture, where the cross-entropy and Jaccard distance are used to address the class imbalance, along with boundary detection, which helps to identify the object by removing the background information. This model failed to differentiate various ecological elements in the marine environment. Channel attention-based U-Net architecture is used for the image enhancement process of underwater image processing that overcomes the degradation effects due to attenuation and scattering effects, which is trained using the LayerNorm and GeLU activation function, showing improved performance by means of Floating Point Operations Per Second (FLOPS) in [14].

The U-Net architecture is designed in light weight manner [15] with no compensation on accuracy using the additional structures like parameter reduction using MRS (Multi-Residual Structure), accuracy improvement using the MSC (Multi-Scale Skip Connection) block, and weight changes using the CBAM (Convolutional Block Attention Mechanism) blocks that helps in fish segmentation process with good trade-off between accuracy and speed. In [15], there is a limitation to feature extraction layers that reduces the proposed lightweight U-Net architecture to work on fine-grained structures. The high-level feature information is fused, and global attentions are provided that help in the extraction of the target-specific segmentation process of aquatic data, which effectively comprises the boundary information, as described in [16], which forms the Semantic Segmentation Network, helping to improve the accuracy of boundary findings. The work in a smaller skewed distribution of classes shows less performance.

TABLE I. Survey on U-Net Segmentation Architectures

Ref.	Year	Model	Metrics
[13]	2025	Keras-U-Net-based binary segmentation using the NAUTEC—UWI Real Dataset	F-Score=94.59% IoU=89.75% Recall=93.38% Precision=95.83%
[14]	2025	Improved U-Net using Channel Attention, GeLU and LayerNorm and tested with the Underwater Imagenet Dataset	Flops reduction by half compared to CycleGAN, UGAN and UGAN-P UCIQE =0.008 (reduced) NIQE=0.019 (Improved)
[15]	2024	Lightweight U-Net using CBAM, MSC, and MRS	Reduction in Volume by 94.20% Reduction in parameters by 94.39% Reduction in floating-point by 51.2% mIOU=94.44% mPA=97.03% FPS=43.62
[16]	2024	Semantic Segmentation using Feature Fusion and Channel Attention Mechanism for Fish Species validated using SUIM dataset and Deep Fish Dataset	mIoU=95.05% mPA=80.37%
[17]	2024	U-Net Architecture using the Spectral global attention followed by Deformable Transformer for Remote Sensing Spectral Applications	Highes mIoU=57.22% Improvement in Accuracy=56.58%
[18]	2023	Attention U-Net for underwater image enhancement tested on UIEB and EUVP datasets	UICM=6.587 UISM=6.839

To segment the hyperspectral images in [17], the work introduces the global spectral attention to obtain the special characteristics of features in the spectrum and semantic information mining is done with the help of a Deformable Transformer that collectively forms the U-Net architecture, which shows a trade-off between high accuracy and parameter reduction, where the spectral redundancy is high due to the inter-band correlations. Image enhancement using a U-Net attention model is introduced in [18] for underwater image improvisations, where the retention of feature fusion and a weighted objective function used for multi-loss are used to show less loss performance that acts on the real-time world applications, but the work involves the downstream tasks (see Table I).

B. Survey on CapsNet-Based Feature Extraction

In the hyperspectral images, the inter-band selection leads to redundancy band selection that reduces the performance in the underwater HSI is addressed in [19] using the CapsNet architecture, where the relative importance of the spectral characteristics is provided using the deep feature extraction blocks consist of dual stream latent variable encoding, where the model suffers from band selection quality due to not mitigating noise. The model for handling the hyperspectral image requires

tackling the effects of non-linearity, which can be done using the Capsule networks that use the encoder and decoder blocks comprised of L2 normalisation, correlation matrix, distance measure and spatial attention mechanism that uses a small number of parameters results in high accuracy when classified but model performance is limited to high computational complexity in [20].

TABLE II. SURVEY ON CAPSNET ARCHITECTURES RELATED TO THE PROPOSED MODEL

Ref.	Year	Model	Metrics
[19]	2025	CapUBS using DLVE and DFEBs that were tested on the dataset from Indian Pines, InPool and Outpool	OA=98.7% AA=98.71% Kappa=98.04%
[20]	2024	SA-CapsNet for 3D spectral images	Accuracy=98.18% Parameters=369,609 Execution time=99.29s
[21]	2024	ShipGeoNET using Capsule networks, R-CNN and ViT Det	Geometric matching
[22]	2024	CapsNet Attention using Self-Attention mechanism for Hyperspectral images	OA=98.15% AA=98.73% Kappa=98.04%

Geometrical extraction methods highly help in the precise localisation, which is proposed in [21] for identifying and detecting the ships in marine environments, where the model excels in exact matching of the ship size onshore using the Radar images. The Vectors are used for the Attention mechanism in [22] for hyperspectral image classification using the CapsNet-based attention module that provides a high level of entity representation that satisfies most of the classification requirements, where the model fails to provide importance for the band weighting mechanism. The 3DCNN [39] models operate with the local receptive fields, this tends to learn the background-dominated convolutional patterns and lead to the overfitting issue of large labelled datasets (see Table II).

C. Survey on Transformer Models

The SpectralUformer model is proposed for UAV-based HSI technology that uses the hybrid attention block for the collection of spectral features and a hybrid cascaded upsampler for aggregation of features for the detection process, but the explanation on spectral normalisation is limited in [23]. To overcome the huge data requirement of the Vision Transformers, in [24], the researchers used an image block that utilises the potential relationships using the Amended Dual block attention mechanism on self-locally and external models, where the block tokenisation is performed to reduce the parameter requirement and provided a high accuracy range, but the work needs improvement in terms of image enhancement.

The hybrid deep learning model focused on the detection of fish species in the deeper sea and uses the ConvMixer and Swin Transformer for segmentation and classification, which helps in underwater exploration despite its bad lighting conditions [25], where the limitation is seen in the segmentation accuracy. The weak signal detection images of small objects in the marine environments is addressed by Hybrid attention transformer that utilises the global attention features and fine-grained features for object detection that reduces the detector performance with less parameter usage in [26] (see Table III).

TABLE III. SURVEY ON TRANSFORMER ARCHITECTURES RELATED TO THE PROPOSED MODEL

Ref.	Year	Model	Metrics
[23]	2024	SpectralUFormer using hybrid attention modules and an upsampler	Accuracy=93.15%
[24]	2024	ADANSE-based ViT on proprietary and standard datasets	Accuracy=90.9%
[25]	2024	ConvMixer and Swin Transformer for Fish classification	Accuracy=94.6%
[26]	2024	Lightweight Hybrid attention transformer for weak and small object detection	Parameter reduction to 28.5M Performance improvement by 6.3%.
[27]	2023	Lightweight Transformer network for Hyperspectral classification	OA=89.03% AA=93.52% Kappa=87.73%

The computational complexity of ViT increases with data requirement, which is handled in [27] using a lightweight architecture that concentrates on position-based and channel-based approaches for hyperspectral image classification process. The feature information is obtained using the convolution blocks and the self-attention mechanism for obtaining the spectral characteristics, where the system fails for complex scenes, as it requires the multi-feature extraction processes. HSI datasets are limited, and there is a need for the large-scale training data, lack of geometric encoding and noisy spectral tokens of the HSI transformer [38] makes it unsuitable for the classification for underwater objects like coral morphology.

From the survey, it is learnt that the existing segmentation model for the Underwater HIS processing lacks global reasoning. The commonly used CNN model does not provide long-range dependencies and the geometric coding using the small sample underwater conditions are present. This motivates the proposed work to concentrate on CAM-U-Net-based segmentation, CapsNet based feature extraction and classification using ViT hierarchical architecture.

III. PROPOSED FRAMEWORK

Based on the challenging factors of underwater exploration due to light absorption, turbidity, saline conditions, etc., compared to different imaging techniques involved in the capturing of information in underwater, Hyperspectral Imaging (HSI) helps a lot to obtain the material specific reflectance information using 100 to 400 spectral bands in the range of 400-900nm that obtains the spectral signatures of underlying objects. Therefore, the proposed model chose the HSI images for underwater image processing, where every information is captured using the specialised optical sensors, illumination systems, and calibration techniques in spite of the underwater distortions.

In the imaging principle, every pixel uses the wavelength to obtain the reflected intensity. Therefore, HSI contains the factors like (x,y, λ) parameters for processing which helps in the identification of the unique underwater materials. A wide aperture optics is accompanied by the optical sensor that obtains the spectral resolution of 1-5nm with a proper waterproof pressure housing, where the sensor captures the full spectral information that will be contained in a one line of pixels at a time in single slit capture. The factors like artificial illumination and illumination geometry optimisation are mandatory to overcome the absorption and scattering issues in underwater image capturing.

To overcome the spectral distortions, various techniques are handled in the calibration module, where the sensor noise is measured using the dark reference, reflectance normalisation using the white reference, alignment of wavelength channels using the spectral calibration and finally the refraction correct distortion using the geometrical calibration method using the formulation of reflectance conversion. Therefore, the final HSI cube information of the underwater imagery contains the number of scanned lines (H), pixels per line (B), and the number of spectral bands (W) that are acquired from one line of spatial pixels, where the full information per pixel is represented by R^(H×W×B). Fig. 1 is the overall proposed framework, where the functioning of each module is discussed below.

A. Underwater HSI Pre-Processing

To provide the spectrally accurate, consistent radiometrics and noise-free spatial data, the preprocessing steps are utilised to overcome the optical distortions mainly due to wavelength-dependent attenuation, absorption, scattering, etc., which helps the data acquired HSI cube become more stable for further segmentation, feature extraction and classification process.

It is well known that only blue-green bands can penetrate deeper, and quick attenuation of red bands requires radiometric and water column compensation due to the presence of spectral imbalance and suppressed reflectance in the HSI cubes data acquired [28]. In the radiometric correction, dark frame response is neglected from the raw sensor information of the HSI cube. The non-uniformity in the illumination is corrected by the normalisation process using the white reference spectrum and division instability. It is known that light attenuation increases with ingoing depths; hence, the reflectance value can be reduced by exponential multiplication of the absorption coefficient of seawater [8] and the estimated optical path length. To maintain the natural colour appearance of the grey world, correction can be applied where the assumption is made that the image average should be neutral grey and spectral band scaling is done using the attenuation rate that re-improved using the weighting mechanism for that particular band.

In the denoising process [29], it is important to preserve the curvature of the spectral signatures amidst the presence of salt and pepper noise, scattering noise and sensor noise, which is done using the 3D Gaussian filtering process that uses the 3D weighted 3D Gaussian kernels of size 3×3×5 that help in maintaining the spectral edges and material-specific peaks. To overcome the instability during the deep learning process due to the presence of illumination differences, scaling of wavelength responses, and convergence, the normalisation is done using the mean intensity of the band and the standard deviation of the wavelength.

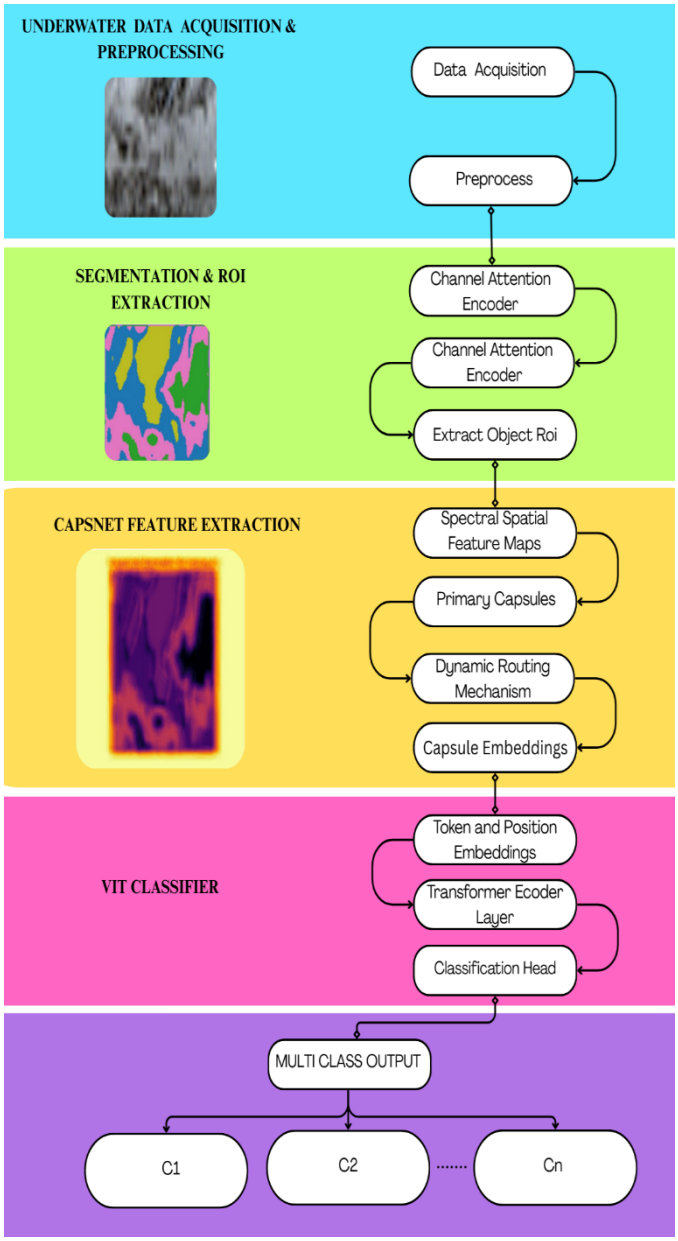


Fig. 1. Proposed framework for underwater HIS.

In contrast, the enhancement process, the band-wise contrast stretching is done using the minimum and maximum of hyperspectral data that contains a low dynamic range. In the next step, Channel Attention (CA) based U-net architecture is employed for processing, hence the sliding window extraction is done to obtain the patch generation based on the patch stride and the stride.

B. Underwater HSI CAM-Based U-Net Segmentation

Despite the preprocessing steps undergone for the HSI cube, the spectral imbalance, distortion, noise presence, etc., are still noticed in it; hence, the proposed framework employs the usage of U-Net architecture for the segmentation process that extracts the desired ROI from the spectral information despite the

background information [30]. It is noticed that the U-Net architecture is designed for RGB images; hence, it is difficult to segment the hundreds of channels present in the 3D HSI cubes. Hence, the proposed framework adopts the Channel attention module in Fig. 2 to improve the recalibration of featuring channels, spectral weight improvisation and spectral-spatial fusion. The proposed CAM-U-Net obtains the class adaptive spectral weights that help in the identification of the underwater scenes that exceed the conventional U-net architecture [31], where the suppression of irrelevant wavelengths is also suppressed during the process.

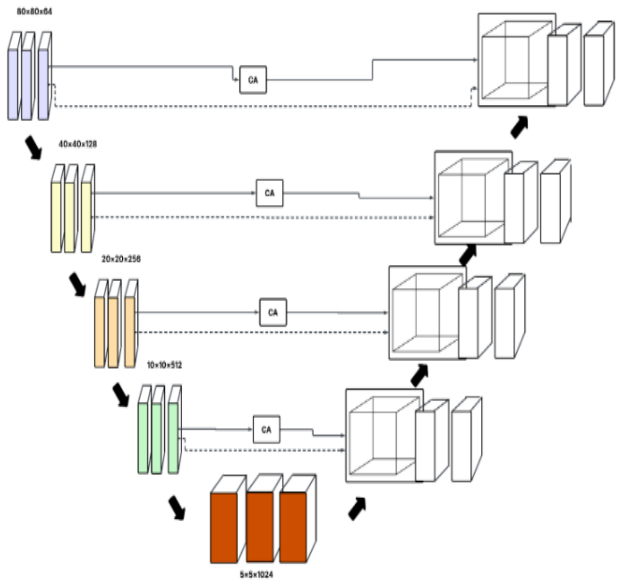


Fig. 2. Channel attention-based U-Net architecture for underwater HIS.

In the encoder of the CAM-U-Net architecture, the spectral spatial feature extraction [32] is done using the convolution blocks and the CAM block. Let us consider F_l as the feature map at the layer l, then the convolution operation using the ReLU activation is given in Eq. (1):

$$F_l = \sigma(W_l * F_{l-1} + b_l) \quad (1)$$

After the convolution is done, the CAM operation is carried out by providing the channel-wise importance weights by learning the feature maps from Eq. (1), where the spectral descriptor for the channel c is given in Eq. (2), which is represented by the Global Average pooling:

$$g_c = \frac{1}{H \times W} \sum_{x=1}^{H} \sum_{y=1}^{W} F_c(x,y) \quad (2)$$

Thus, in Eq. (2), g_c represents the particular channel global spectral activation; therefore, the final spectral representation is given by g [g1, g2……gc]. At the bottleneck layer, the 2-layer MLP is used to obtain the nonlinear channel relationships using the spectral activation and reduction layers ($W1. \in \mathbb{R}^{\frac{c}{r} \times c}$) by Eq. (3):

$$z = \delta(W_1 . g) \quad (3)$$

Using Eq. (3), the channel attention weights are obtained by the sigmoid function (σ) and the expansion layer ($W_{2.} \in \mathbb{R}^{c \times \frac{c}{r}}$), where r is the reduction ratio given in Eq. (4):

$$w = \sigma(W_2 z) \quad (4)$$

Therefore in each channel, the learned importance is obtained by multiplying it with the particular channel feature $F_{c,}$ which is represented in Eq. (5), where the attention weights lie in the range of [0,1].

$$F_c'(x,y) = w_c . F_c(x,y) \quad (5)$$

Thus, the refined feature F' contains the useful spectral bands, where the irrelevant wavelengths are suppressed. The maxpooling operations are done to enlarge the receptive fields of the refined features, which helps in obtaining the deeper features for ROI extraction. The upsampling of the layer l U_l using the transposed convolution operation that is fused with the refined features is given in Eq. (6):

$$S_l = Concat(U_l, F_l') \quad (6)$$

Then the channel attention module is applied to Eq. (7), which produces the fused representation S_l', where the convolution converted to decoder features is given in Eq. (7):

$$O(x,y) = W_{1\times1} * S_o'(x,y) + b \quad (7)$$

Then, the pixel-wise probabilities for Eq. (7), are given in Eq. (8) using the softmax operations:

$$P(x,y,k) = \frac{\exp(O_k(x,y))}{\sum_{j=1}^{k} \exp(O_j(x,y))} \quad (8)$$

Thus, using the final pixel-wise probabilities, the final segmentation map is obtained in Eq. (9):

$$\hat{M}(x,y) = \arg\max_k P(x,y,k) \quad (9)$$

Thus, the proposed framework of CAM-based U-net segmentation provides final segmentation details that contain the CAM amplified informative wavelengths, CAM suppressed noisy channels, edges are recovered using the decoder and skip connections, class-specific channels are obtained, and the U-net helps in obtaining the object size variability.

ROI Extraction: The segmented data need to match the input for the CapsNet feature extraction process; hence, the ROI patch generation is an important task that needs to be carried out to integrate the segmentation and feature extraction process in the proposed framework. Therefore, the bounding box based on the class of interest is obtained using Eq. (10):

$$ROI_{box} = H(x_1 : x_2, y_1 : y_2, :) \quad (10)$$

Then the above desired ROI extractions are resized for the Capsnet input that is given in Eq. (11):

$$X = Resize(ROI_{box}, r \times r \times C) \quad (11)$$

C. Underwater HSI CapsNet-Based Feature Extraction

The targeted underwater objects are obtained from the ROI extraction process using the CAM-based U-net Segmentation process that resolves the issues of mixed pixel contamination, boundary ambiguity, and spatial clutter in underwater scenes

based on the pixel-wise segmentation mask. It is important to extract the features [33] from the ROI that needs to contain high-level features that need to capture the irregular shapes, non-rigid structure, point variations, and depth-based spectral variations that are the characteristics of the underwater objects (see Fig. 3).

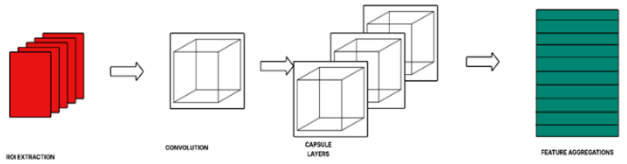


Fig. 3. ROI extraction for CapsNet for underwater HIS.

The learning of low-level features like texture, edges and local spectral gradients is obtained from two convolutional layers before the formation of capsules, which is given in Eq. (12) and Eq. (13):

$$F_1 = \sigma(W_1 * X + b_1) \tag{12}$$

$$F_2 = \sigma(W_2 * F_1 + b_2) \tag{13}$$

Thus, the F_2 contains the capsule formation that contains information about underwater object features. The capsule output [34] for each that is reshaped using the convolution feature maps is given in Eq. (14):

$$s_i = \sum_p W_{ip} . F_2(p) \tag{14}$$

The capsule vector is computed using the squash function that helps in providing the nonlinear vector length given in Eq. (15):

$$\hat{u}_{j|i} = w_{ij} u_i \tag{15}$$

The dynamic routing mechanisms help in mapping the parent node to the child node with the help of the coupling coefficients given in Eq. (16), where the nonlinear capsule outputs are produced in the vector form using the squash function that is represented in Eq. (17):

$$s_j = \sum_i c_{ij} \hat{u}_{j|i} \tag{16}$$

$$v_j = squash(s_j) \tag{17}$$

The transformational relationships help in capturing the coral reefs' curvature, texture, contour shapes, orientation variations based on the water movement, etc., which are represented in the Fcaps shown in Eq. (18), that is, actually the penultimate capsule output, where the softmax is not used, provides the deep features information.

$$F_{caps} = [v_1, v_2, \ldots . v_U] \tag{18}$$

D. Underwater HSI Vision Transformer-Based Classification

The HSI classification for underwater is done using the Vision Transformer that provides the long-range relationships to the spectral-spatial relationship, where the global attentions are provided to the capsule vectors obtained from Eq. (19):

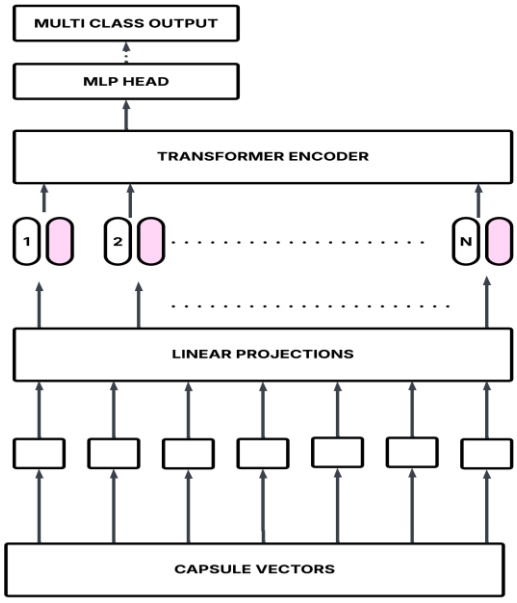


Fig. 4. ViT-based classification for underwater HIS.

$$u_i = squash(s_i) \tag{19}$$

The predictions are shared using the primary capsules for the higher level capsules using the formulation in Eq. (16).

The input of the ViT [35] requires a fixed level of dimension tokens; hence, the capsule vectors in straight act as the tokens, which is followed by the linear transformations [36] that are given in Eq. (20), which contains the learnable projection matrix Wp, and bias b.

$$L_i = W_p v_i + b_p \tag{20}$$

Now the tokenisation is given by T [L1, L2….LU]. Now the positional encoding is added to help the final classification process, which is done using the additions of the CLS token to the patches that are represented in Eq. (21):

$$Z_i^{(0)} = CLS + L_i + PE_i, \ \omega\eta\varepsilon\rho\varepsilon \ \iota = 1,2,,,,,,Y. \tag{21}$$

Now the self-attention is applied in the transformer block, where the neural network contains the query, key, and Value functions that extract the global and long-range dependency of the data. The value function extracts the actual object information present in ROI based features, the Query collects the information where the data contains the object-based information, and the Key collects the information that the query is trying to identify. The Query and Key parameters help to identify the classification pattern highlighted in the Value function using the attention formula given in Eq. (22):

$$A_i = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{D}}\right) . V_i \tag{22}$$

The above attention values are obtained from the multi-heads of the transformer, to which the residual and layer normalisation are carried out. At the end using the CLS token that represents the entire patch is used for classification using the feed-forward neural network. Thus, the ViT helps in

handling the class ambiguity, irregular object shapes, which are common in underwater environments, and feature loss due to scattering.

IV. Results and Discussion

A. Implementation

The proposed framework is tested on the dataset Taxonomically annotated underwater hyperspectral and coral images of coral reef transects [37] that contains the hyperspectral images of underwater using the wavelength (400-750nm) using the 100-300spectral bands under the natural water illumination. The dataset contains the hyperspectral cubes and is annotated with detailed labelling of coral leaf substrates, corals, algae, sponges, sand, and rubble. Table IV provides the attributes of the datasets.

TABLE IV. HIS Dataset Attribute

Parameters	Value
No of Hyperspectral cubes	*120*
RGB images	*240*
Hyperspectral Cube Size	*512×512×(100-300) bands*
Spectral range	*400-750nm*
Mean annotated classes	*8*
Depths of transects	*5-30m*
Spatial resolution	*1-3mm/pixel*
ROI patches	*3730*

B. Data Preprocessing

The dataset contains the HSI cubes of size 512×512×no of bands, undergoes data preprocessing to reduce the noise and spectral distortion, which helps in the stable learning of the deep learning techniques involved in the proposed framework. At the initial stage, to reduce the spectral bias among the wavelengths, radiometric correction is done using the sensor-level dark current frames and white reference calibration for compensation purposes. Exponential absorption model is used for water column attenuation, and the normalisation is carried out using the z-sore among the cross-band illumination differences. The 3D Gaussian filters are used to suppress the shot noise and scattering distortions without disturbing the spectral gradients using the 3×3×5 spatial window filter size. The redundant information of spectral bands is reduced by the use of a 1×1 convolutional filter that compresses the channels to 16, which is followed by contrast stretching for improving the boundary visibility.

C. Segmentation

The encoder stage contains the augmented process of the U-net architecture and the CAM module using the four downsampling blocks, using 3×3 convolution per block, followed by the global average pooling and two MLPs, where the reweighting of the spectral channels is carried out using the CAM block. The connection between the encoder and decoder is maintained using the skip connections that help to maintain the spatial details, which are distorted due to the turbidity present in the underwater environment. The transpose of convolution upsampling takes place in the decoder, and the pixel-wise probabilities are obtained from the softmax operation using a 1×1 convolution operation. Now the mask of resolution 256×256 that is segmented for the classes of Pocillopora, Porites, Acropora, Macroalgae, sponge, and sand.

D. Feature Learning

The segmented masks are resized to a 64×64 bounding box that forms the ROI patches for the CapsNet feature extraction process. The two convolution layers provide the 9×9 kernel filters for extracting the low-level spectral and spatial features that are reshaped to 8-dimensional capsules for primary capsules that extract the high-level feature extraction that is concatenated into 10-digit capsule layers by a dynamic routing mechanism using the coupling coefficients. The features contain the capsule vector encoded geometry, orientation, and spectral reflectance patterns of similar coral taxa.

E. Classification

The capsule vectors act as the tokens that help in the tokenisation process, which are linearly projected for transformations followed by positional encoding. The 8 transformer layers with 6 Multi-Head Self-Attention and GeLU-activated feed-forward neural network are used for classification. The prediction of coral substrate classes is identified using the two-layer MLP using the CLS from the final layer.

This architecture combination ensures that the spatial relationships preserved by the capsules are effectively interpreted through the transformer's global context. By utilizing multi-head attention, the model selectively focuses on the intricate coral textures and geometric patterns that distinguish various substrate types. The feed-forward layers further refine these features, allowing for high-dimensional representations that remain resilient against underwater noise and lighting variations. As the data passes through the final transformer stage, the CLS token aggregates the most salient information required for a definitive categorization. Consequently, the two-layer MLP serves as a precise decision-making head, mapping these complex embeddings into distinct categorical probabilities with high accuracy. This integrated approach significantly enhances the model's ability to differentiate between the subtle morphological variations inherent in various coral species. The trainable parameter counts are estimated to be 15M, with the inference time of 30ms/ROI and Peak GPU memory of 3.5GB.

F. Implications

The visualization of Class Activation Maps (CAM) across the 10-band spectral data provides profound insight into the model's localized focus during the segmentation phase. By leveraging a U-Net architecture integrated with Channel Attention mechanisms, the network effectively filters out underwater backscatter while prioritizing the most discriminative morphological features of the coral substrates. Each feature map serves as a diagnostic window, revealing how the multi-band input—extending from visible light to specific spectral wavelengths—enables the system to bypass the inherent limitations of traditional RGB imagery. This high-dimensional feature extraction ensures that the boundaries of complex, overlapping coral colonies are delineated with high fidelity, significantly reducing false positives in heterogeneous seafloor

environments. Integrating 10-band spectral data with Channel Attention enables the U-Net to isolate critical morphological features while suppressing underwater noise, as demonstrated in the CAM visualizations (Fig. 4), for autonomous benthic mapping in complex seafloor environments.

Fig. 5 is the illustration of segmentation visualisation using the CAM-based U-net architecture for 10 bands, where the different band level shows different feature representation that helps in obtaining the informative bands. The higher bands 7 to 10 show good performance with more stable activations, which help in identifying the class-specific spectral characteristics using the channel attention recalibrations. The Gaussian-like activation regions are noted in the feature bands 3 to 6 that show more reliable reflectance structure for coral and benthic surfaces. In bands 1 and 2, the discriminative performance of the proposed framework has been obtained due to the strong attenuation for shorter wavelengths. Fig. 6 helps in obtaining the ROI extraction, where the mask was generated for band 3, which shows the high activation region, spatially coherent clusters, and segmentation contour overlay. Fig. 7 is the mean activation intensity of the segmentation process, which shows bands 7 to 9 have dominating spatial discriminations, which helps in obtaining the prioritised spectral channels. Table V presents the simulation parameters.

The bands 1, 3, and 10 have low values, indicating less involvement in the segmentation process. This selective prioritization not only streamlines the computational load during feature extraction but also enhances the signal-to-noise ratio by de-emphasizing the blue-dominated backscatter typically found in bands 1 and 3.

Consequently, the segmentation process becomes significantly more robust against the variable turbidity and lighting conditions common in deep-sea environments. Such channel-level optimization is pivotal for ensuring that the subsequent classification stages receive a refined, highly representative input, ultimately leading to superior identification of subtle coral substrate variations. This data-driven channel selection validates the use of multispectral imaging as a superior alternative to conventional three-band systems for complex biological mapping.

TABLE V. SIMULATION PARAMETERS

Parameters	Values
CAM-U-Net Architecture	
Input size	**256×256×16**
Encoder	4
Filter per level	64,128,256,512
Convolution	3×3, stride 1, padding 1
Activation	Relu
Pooling	Maxpooling 2×2
Channel attention reduction ratio	R=8
Decoder upsampling	Transposed convolution
ROI EXTRACTION	
Mask threshold	ARGMAX
Bounding box method	Contour based
Resize	Bilinear interpolation
Roi size	64×64×16
CAPSNET FEATURE EXTRACTOR	
Conv layer 1 and 2	256 filter, 9×9 kernel
Primary capsules	32 capsules × 8d vectors
Class capsules	10d ×16d vectors
Routing iterations	3
Capsule dim output	16d per class capsule
VIT CLASSIFIER	
Token dimension	**128**
No f token from Capsnet	32
Projection layer	16D→128D
Transformer layers	6
Attention heads	8
MLP DIM	256
CLS	Enabled
TRAINING SETUP	
Framework	PYTORCH
Optimiser	ADAM
Learning rate	1E-4
Batch size	16
Dropout	0.1
Hardware	NVIDIA RTX GPU
Regularization	1E-5

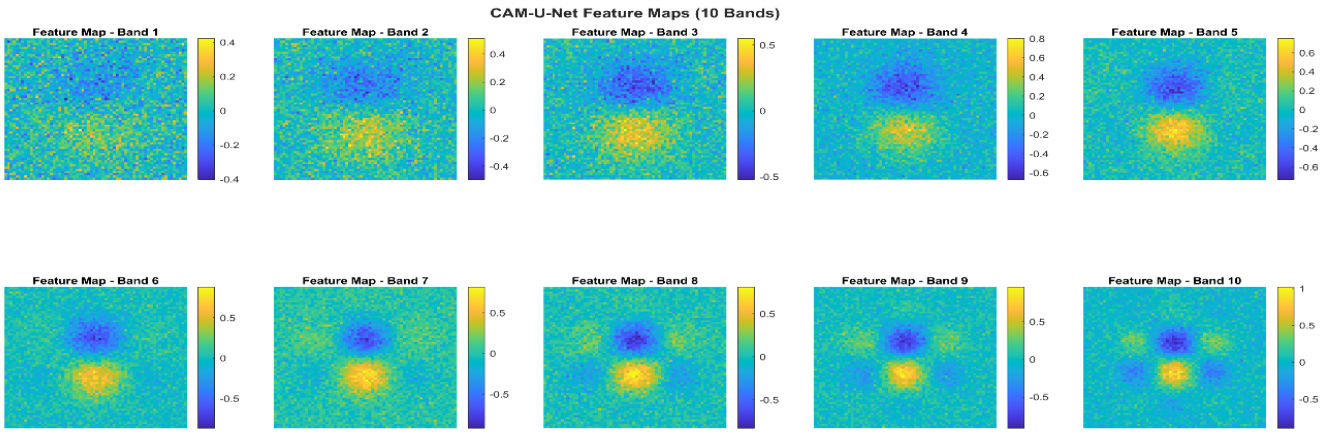


Fig. 5. CAM-based U-Net segmentation visualisation feature map for 10 bands.

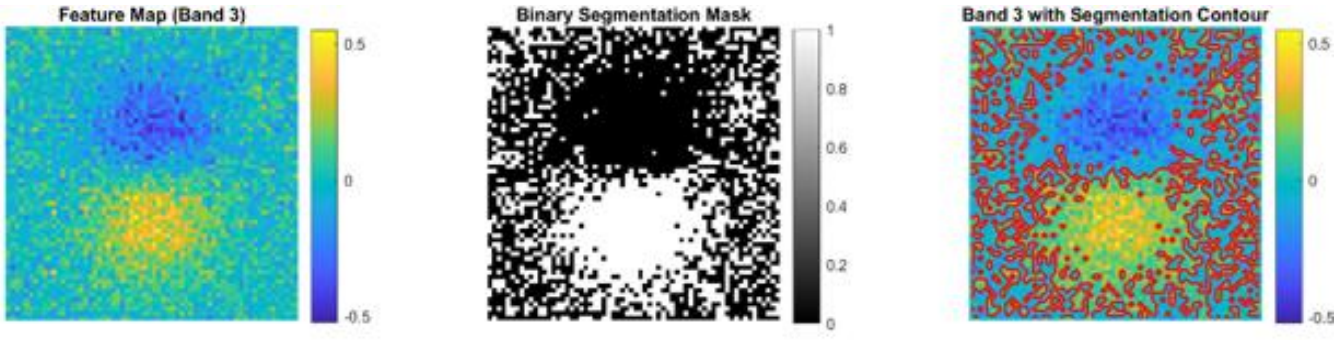


Fig. 6. CAM-based U-Net segmentation visualisation feature map and mask.

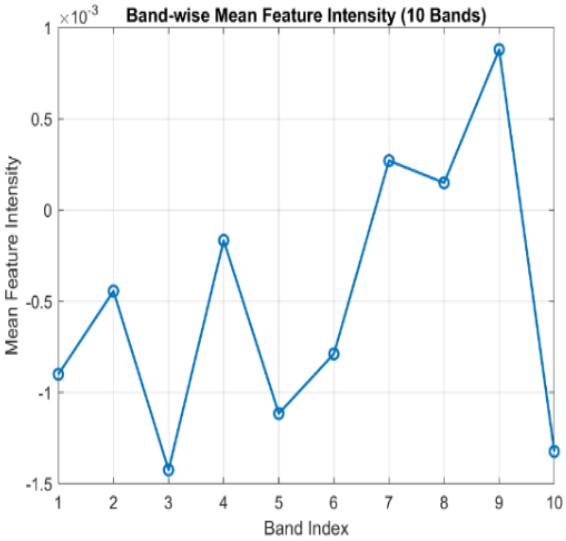


Fig. 7. Bandwise mean feature intensity for 10 bands using CAM-based U-Net segmentation visualisation.

Table VI is the CAM-based U-Net architecture for segmentation of substrate classes using the Intersection Over Union (IoU) to measure the overlapping, precision used for identifying the predicted positive pixels, recall helps to know the capturing most of the object, F scores helps in false negatives and positives and pixel segmentation helps in correctness that are manipulated using True positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN).

The highest IoU is seen in the sand that has different texture patterns and unique spectral reflectance, where the coral class Acropora preserves the good boundary conditions with the IoU of 0.82. In the overall estimation, the strong segmentation performance is noted by the IoU value between 0.72 and 0.88 among the substrate classes. It is important to obtain a false positive rate that is minimal, which is estimated using the precision calculation, showing a range from 0.80 to 0.93. The recall helps in obtaining the features of class regions, where 0.91 is at maximum and helps in calculating the trade-off F-score that shows consistent segmentation performance across different classes of spectral bands. Thus, the ROI patches have sharper boundaries, region consistency, and confident segmentation output, which is indicating highly discriminative and uniform characteristics.

TABLE VI. CAPSNET FEATURE EXTRACTION METRICS

Class	*Capsule Mean Length*	*Capsule Variance*	*Spread*
Acropora coral	0.862	0.041	0.203
Pocillopora coral	0.835	0.049	0.221
Porites coral	0.812	0.052	0.238
Soft coral	0.778	0.067	0.265
Macroalgae	0.902	0.036	0.184
Turf algae	0.791	0.059	0.249
Sponge	0.804	0.055	0.232
Sand	0.918	0.029	0.162
Acropora coral	0.862	0.041	0.203

Table VI is the CapsNet feature extraction performance on the ROI extracted data from the segmentation modules, where the factors like capsule mean length, capsule variance and spread help in the statistical analysis of capsule outputs. To know the presence of a feature, the capsule length can be used from the analysis of the sand, and the macroalgae show the highest identity of spectral-spatial features. To measure the stability of the capsule across various classes of ROI extraction, capsule variance is used, where the higher variance of soft coral has a higher 0.067 because of the greater intra-class variability based on the textures and spectral mixing. The capsule activation among the intra-class dispersion is noted using the spread function, where the soft coral shows a high of 0.265.

Table VI provides a comprehensive statistical analysis of the performance of CapsNet in feature extraction from region-of-interest (ROI) data derived from the segmentation modules. A comprehensive understanding of how well the capsule network reflects spectral–spatial features across several aquatic classes is provided by the analysis, which focuses on capsule mean length, capsule variance, and spread. By displaying the strength, stability, and dispersion of capsule activations, these measures provide deeper interpretation beyond classification accuracy.

Within each class, the feature existence confidence is directly shown by the capsule mean length. Stronger and more reliable activation of class-specific spectral-spatial characteristics is implied by higher values. Because of its

relatively homogeneous texture and low spectral mixing, sand has the highest capsule mean length of 0.918 in the data. In underwater photography, macroalgae have a high mean length of 0.902, which reflects their unique spectral response and unusual spatial patterns.

The statistical analysis in Table VI indicates that CapsNet proficiently captures substantial and interpretable spectral–spatial data, especially for classes exhibiting consistent visual traits. In order to improve global contextual knowledge, complementary mechanisms like attention and transformer-based modules are required, as evidenced by the observed variance and spread in complex biological classes. These results confirm that capsule-based feature modelling is a solid basis for classifying underwater images.

Alongside class-specific statistical results, the capsule-based study underscores CapsNet's capacity to maintain the spatial hierarchies intrinsic to underwater objects. Capsule representations capture both the existence and instantiation properties of features, in contrast to traditional CNN feature maps. This is especially useful in underwater environments where illumination, scale, and orientation vary greatly. Because of this feature, CapsNet can continue to provide discriminative representations even in the face of extreme optical aberrations. Compact capsule distributions with longer mean lengths indicate better separability in the latent space from the standpoint of categorization, which directly promotes trustworthy decision boundaries. The statistical compactness found in substrate-dominant groups like macroalgae and sand suggests that CapsNet can accurately and minimally confuse low-texture regions. On the other hand, the network's capacity to encode part-whole interactions, including branching, folding, or layered structures, is advantageous for biologically rich classes.

Furthermore, the segmentation-driven ROI extraction approach is indirectly validated by the capsule statistics. The segmented regions maintain important object-level information while decreasing irrelevant background noise, as demonstrated by stable capsule activations across several classes. The whole framework's resilience is strengthened by the synergy between segmentation and capsule-based feature extraction.

Global contextual awareness is required to further resolve ambiguities among visually overlapping classes, even while CapsNet excels at local structure preservation. Consequently, a solid empirical foundation for integrating attention mechanisms and transformer-based modelling in subsequent trials is provided by the capsule feature behaviour seen in Table VI.

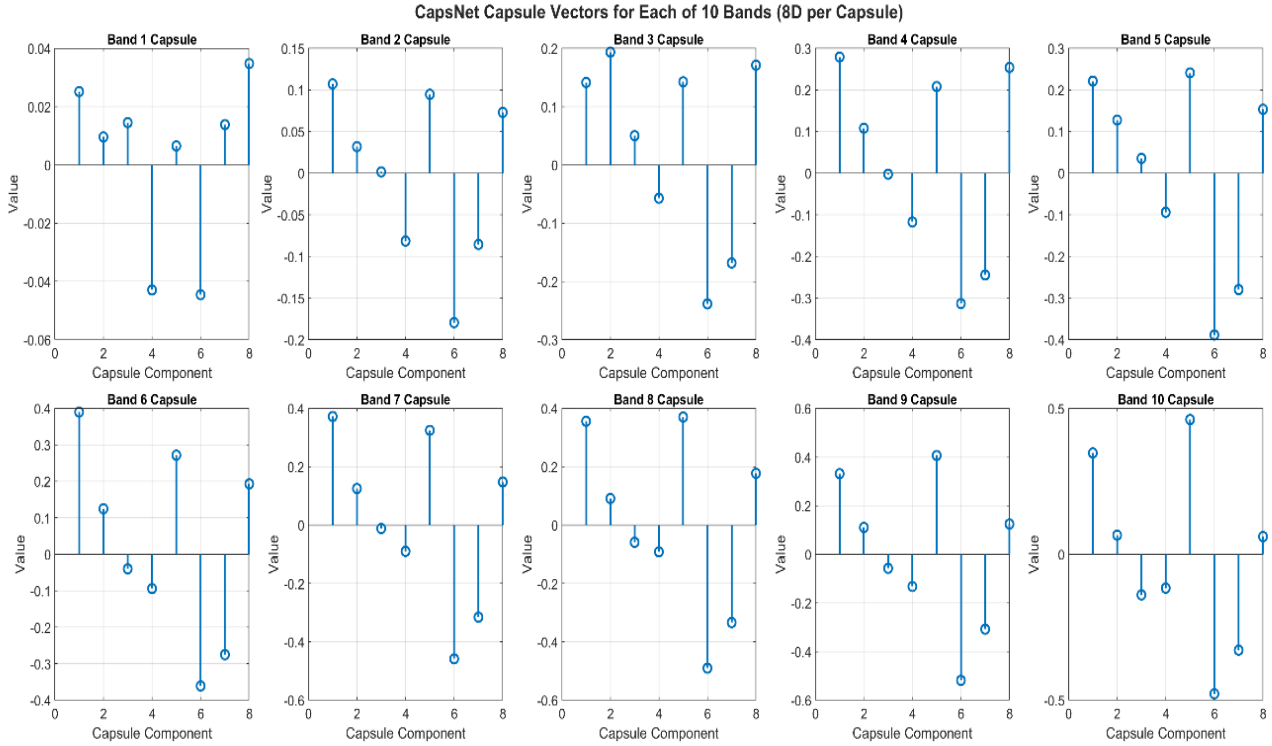


Fig. 8. Capsule vector for 10 bands based on the ROI extracted patches from the segmentation process.

Fig. 8 is the depiction of the CapsNet feature extraction projections using the ROI patches extracted from the segmentation process, which shows the effective multi-band feature abstraction with both positive and negative vectors indicating the spatial relationships based on the pose and magnitude-based features. The band 10 outputs are obtained from the 8-dimensional feature representation helps in exhibiting the spectral spatial characteristics among the bands. In the sensitive underwater imaging, the bands 3,6,7,8, and 10 help in the identification of corals, algae, sand, etc., classes present in the HSI cubes based on the higher vector components that capture the band-specific spatial spectral characteristics. Less dominance is noticed in the spectral bands of 4, 5 and 9 that contain the structured activations with informative spectral information. The shorter wavelengths showed low performance at bands 1 and 2, mainly due to the attenuation factor under the

sea, which showed weak capsule activations. These findings show that the capsule network creates discriminative pose-aware representations by efficiently utilizing informative spectral bands, which directly enhances class separability. In complex underwater sceneries, the predominance of mid-to-long wavelength bands allows for robust recording of object borders and texture variations. Stronger capsule activations in these bands are therefore associated with improved physiologically varied class recognition. The significance of choosing reliable spectral regions underwater is further highlighted by the diminished impact of shorter wavelengths. All things considered, the band-wise capsule behaviour shows that CapsNet is appropriate for spectral-spatial feature learning in difficult underwater hyperspectral situations.

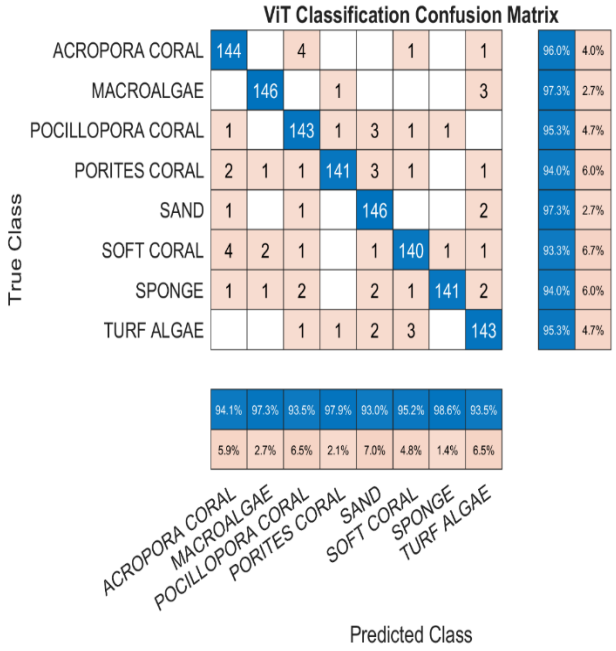


Fig. 9. Confusion matrix from ViT classification.

Fig. 9 is the confusion matrix obtained for the 8 substrate classes of the dataset of underwater HSI, which are preprocessed and segmented using CAM CAM-based U-net architecture and CapsNet-based feature extraction. The confusion matrix produces the cues for true class and predicted class for 8 substrates. The proposed framework highly classifies the sand class with 97% efficiency because of its fine-grained texture patterns that are effectively segmented and feature extracted.

The overall range is between 93 and 98% for all the classes of substrates in the underwater HSI datasets. This shows the preservation of spectral spatial characteristics by the proposed segmentation, features collection by CapsNet feature extraction, and classification, which obtains global attention based on range dependencies among the classes. The misclassification is noted in the coral taxonomy due to the similar structural patterns and spectral overlaps. Based on the confusion matrix, the proposed work plots the accuracy and error curves for 100 epochs, where the accuracy stood on average at 95% with an error percentage of 5%. The proposed method shows consistent learning activity till 60 epochs and saturates after that, as depicted in Fig. 10.

Fig. 11 is the depiction of AUC curves for 8 substrate classes that help to show the proposed framework's performance robustness for true negative and false positive rates. All the classes shown above 0.99 had the highest AUC values of macroalgae due to the distinctive spectral signatures for various hyperspectral bands. Table VII discusses the various ViT classification hybrid models, where the proposed combination framework shows higher performance.

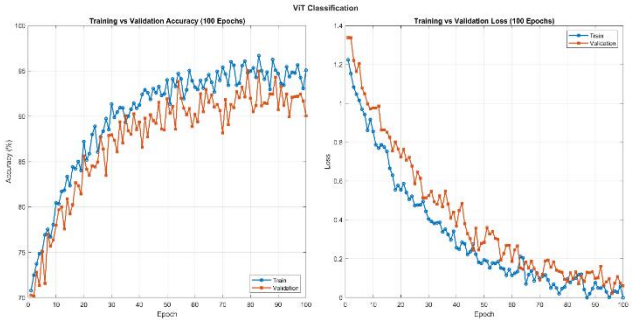


Fig. 10. ViT classification of underwater HSI based on CapsNet feature extraction and CAM-based U-Net segmentation.

The training curves' constant convergence behaviour demonstrates how stable the suggested learning framework is in challenging underwater environments. While the saturation after 60 epochs shows that the model achieves an ideal representation without overfitting, the progressive decrease in error rate during early epochs reveals efficient feature learning from segmented ROI patches. This characteristic is especially crucial for undersea hyperspectral data, as significant spectral redundancy and a lack of labelled samples can otherwise impair generalization.

The robustness of the suggested framework in differentiating between visually and spectrally similar categories is further demonstrated by the good AUC performance across all substrate classes. Reliable underwater scene interpretation depends on the model's capacity to maintain a favourable balance between true positive and false positive rates, which is confirmed by the nearly flawless AUC values. Macroalgae achieves the highest AUC due to its distinctive spectral reflectance and spatial continuity, enabling clearer decision boundaries within the feature space.

Common morphological traits and overlapping spectral responses are the main causes of misclassification within coral taxonomy, particularly between branching and large coral taxa. However, by maintaining spatial hierarchies while concurrently modelling long-range relationships, the network is able to reduce these ambiguities through the merging of capsule-based representations with transformer-driven global context. In difficult situations where local texture alone is insufficient, this dual representation capability enhances differentiation.

The channel attention mechanism suppresses redundant or noise-prone channels and highlights useful spectral regions to further improve feature selection. This selective amplification stabilizes categorization results across epochs and enhances learning efficiency. The Vision Transformer, CapsNet, and channel attention modules work together to acquire complementary information at various representation levels.

Overall, the performance trends show that the suggested hybrid design successfully strikes a compromise between global contextual understanding and local structural awareness. The framework's appropriateness for underwater hyperspectral image classification is validated by its good classification accuracy, robust convergence behaviour, and consistently high AUC values. These findings demonstrate that the suggested approach can function as a trustworthy basis for sophisticated mapping of marine habitats and automated substrate analysis.

Additionally, the integrated design shows tolerance to illumination variability and spectrum disturbance that are frequently found in underwater conditions. More accurate feature discrimination across diverse substrates is made possible by the combined learning of spectral context and spatial structure. For complicated benthic classes, where minute characteristics affect classification results, this is especially advantageous. Strong generalization ability across several ROI samples is shown by the observed performance consistency. As a result, the suggested architecture provides an efficient and well-rounded solution for underwater hyperspectral classification problems.

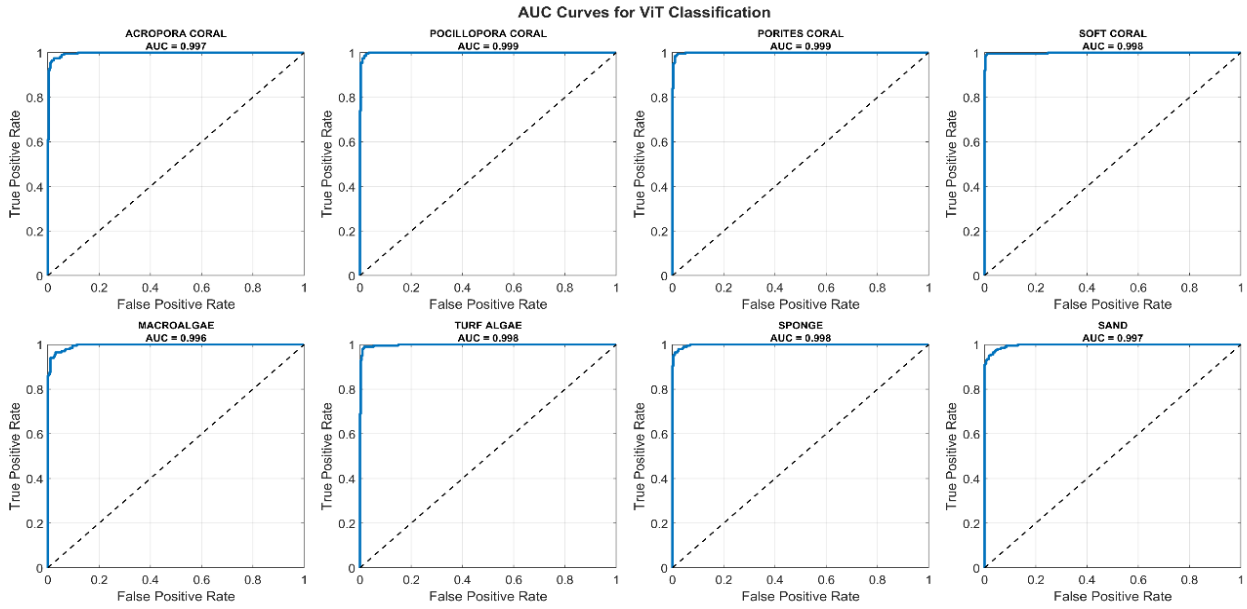


Fig. 11. AUC curves for 8 classes of underwater HSI using ViT classification.

TABLE VII. COMPARATIVE STUDY BASED ON THE PROPOSED MODELS

Ref	Year	Models	Accuracy
[23]	2024	SpectralUFormer using hybrid attention modules and an upsampler	93.15%
[24]	2024	ADANSE-based ViT on proprietary and standard datasets	90.9%
[25]	2024	ConvMixer and Swin Transformer for Fish classification	94.6%
[27]	2023	Lightweight Transformer network for Hyperspectral classification	89.03%
Proposed Model (CAM-based U-Net+Capsnet+ViT)			**95.3%**

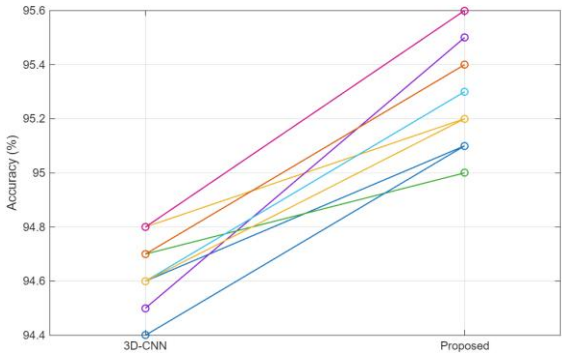


Fig. 12. Paired runs test for 3DCNN with the proposed model.

The statistical analysis is carried out using the paired runs with the 3D CNN [39] and the proposed framework against the measure of accuracy. It is seen that the baseline accuracy range is approximately 94.5% with the proposed accuracy range of 95.5% that shown the improvement per run to 0.4% to 0.9% (see Fig. 12).

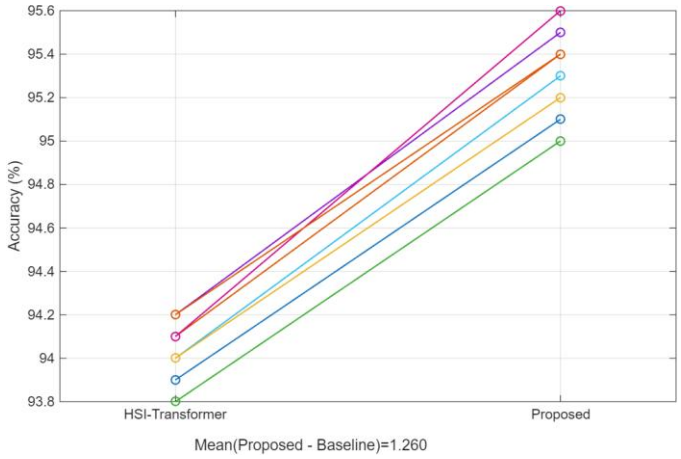


Fig. 13. Paired runs test for the HSI transformer with the proposed model.

Fig. 13 depicts the paired run test for the HIS transformer and the proposed framework. The proposed work shows stable gain using the channel attention-based segmentation, capsule features that provide pose-aware and the context based global transformer with a baseline accuracy range of 94.5% accuracy approximately and a mean improvement of +1.26%.

V. Conclusion

The advanced hybrid deep learning techniques that utilise the CAM-based U-et architecture for segmentation, CapsNet-based Feature Extraction, and ViT classification for Underwater Hyperspectral Image classification are framed and obtained the accuracy of 95.3% with AUC 0.99. The proposed framework obtains the region of interest from the hyperspectral images from 10 bands input by channel attention, which helps the U-net architecture to spatially and spectrally characterise multi-band hyperspectral images. The CapsNet features extract the pose and spatial relationships using the capsules and dynamic routing mechanism from the ROI extraction of segmented bands. The VitClassification uses the capsule vectors for identifying the global relationships using the query, key, and value in the transformer heads that provide range relationships among the features to identify the classes of substrates. The proposed framework achieved pixel accuracy of 95.2% with a maximum IoU of 0.88. In future works, as follows: a) The scalability of the proposed work applies to all hyperspectral image-based applications. b) Band-wise classification enhancement using the advanced generative models to improve the performance of the overall systems, and c) The proposed framework suffers from computational complexity; hence, the lightweight models need to be incorporated.

Conflict of Interest

The authors declare that there is no conflict of interest related to the research.

References

[1] Mittal, Sparsh, Srishti Srivastava, and J. Phani Jayanth. "A survey of deep learning techniques for underwater image classification." IEEE Transactions on Neural Networks and Learning Systems 34, no. 10 (2022): 6968-6982.

[2] Naveen, Palanichamy. "Advancements in underwater imaging through machine learning: Techniques, challenges, and applications." Multimedia Tools and Applications 84, no. 22 (2025): 24839-24858.

[3] Elmezain, Mahmoud, Lyes Saad Saoud, Atif Sultan, Mohamed Heshmat, Lakmal Seneviratne, and Irfan Hussain. "Advancing underwater vision: a survey of deep learning models for underwater object recognition and tracking." IEEE Access (2025).

[4] Li, Ming, Hanqi Zhang, Armin Gruen, and Deren Li. "A survey on underwater coral image segmentation based on deep learning." Geo-spatial Information Science 28, no. 2 (2025): 472-496.

[5] Prabhakar, G., S. Selvaperumal, and B. Baranitharan. "YOLOv5-based Enhanced Underwater Seaweed Detection Using Open-Source Datasets." Recent Patents on Engineering 19, no. 6 (2025): E300524230521.

[6] Aubard, Martin, Ana Madureira, Luís Teixeira, and José Pinto. "Sonar-Based Deep Learning in Underwater Robotics: Overview, Robustness, and Challenges." IEEE Journal of Oceanic Engineering (2025).

[7] Nawaz, Uzma, Mufti Anees-ur-Rahaman, and Zubair Saeed. "A Survey of Deep Learning Approaches for the Monitoring and Classification of Seagrass." Ocean Science Journal 60, no. 2 (2025): 19.

[8] Zhu, Rongxin, Lei Sheng, Kaitao Wu, Azzedine Boukerche, Libo Long, and Qiuling Yang. "Toward Efficient Underwater Visual Perception through Image Enhancement, Compression, and Understanding." ACM Computing Surveys (2025).

[9] Cheng, Ming-Fang, Arvind Mukundan, Riya Karmakar, Muhamed Adil Edavana Valappil, Jumana Jouhar, and Hsiang-Chen Wang. "Modern Trends and Recent Applications of Hyperspectral Imaging: A Review." Technologies 13, no. 5 (2025): 170.

[10] Wang, Zhixin, Peng Xu, Bohan Liu, Yankun Cao, Zhi Liu, and Zhaojun Liu. "Hyperspectral imaging for underwater object detection." Sensor Review 41, no. 2 (2021): 176-191.

[11] Montes-Herrera, Juan C., Emiliano Cimoli, Vonda Cummings, Nicole Hill, Arko Lucieer, and Vanessa Lucieer. "Underwater hyperspectral imaging (UHI): A review of systems and applications for proximal seafloor ecosystem studies." Remote sensing 13, no. 17 (2021): 3451.

[12] Longhi, Valeria, Nives Grasso, Paolo Felice Maschio, Fabio Menna, Arianna Pansini, Andrea Maria Lingua, Filiberto Chiabrando, Erica Nocerino, Giulia Ceccherelli, and Sabrina Rossi. "Underwater hyperspectral imagery for Posidonia oceanica mapping: challenges and preliminary results from the POSEIDON Project." The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 48 (2025): 177-184.

[13] George, Geomol, and Anusuya S. "Pretrained U-Net: in-depth analysis of binary image segmentation in underwater marine environment." Machine Learning for Computational Science and Engineering 1, no. 1 (2025): 4.

[14] Zhu, Sisi, Zaiming Geng, Yingjuan Xie, Zhuo Zhang, Hexiong Yan, Xuan Zhou, Hao Jin, and Xinnan Fan. "New Underwater Image Enhancement Algorithm Based on Improved U-Net." Water 17, no. 6 (2025): 808.

[15] Zhang, Zhenkai, Wanghua Li, and Boon-Chong Seet. "A lightweight underwater fish image semantic segmentation model based on U-Net." IET Image Processing 18, no. 12 (2024): 3143-3155.

[16] He, ZhiQian, LiJie Cao, JiaLu Luo, XiaoQing Xu, JiaYi Tang, JianHao Xu, GengYan Xu, and ZiWen Chen. "UISS-Net: Underwater Image Semantic Segmentation Network for improving boundary segmentation accuracy of underwater images." Aquaculture International 32, no. 5 (2024): 5625-5638.

[17] Zhang, Tianjian, Zhaohui Xue, and Hongjun Su. "Deformable transformer and spectral U-Net for large-scale hyperspectral image semantic segmentation." IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 17 (2024): 20227-20244.

[18] Tang, Pengfei, Liangliang Li, Yuan Xue, Ming Lv, Zhenhong Jia, and Hongbing Ma. "Real-world underwater image enhancement based on attention U-Net." Journal of Marine Science and Engineering 11, no. 3 (2023): 662.

[19] Li, Qi, Junwen Wang, Xingyuan Zu, Dan Chen, Ke Zhang, and Jinghua Li. "CapUBS: Capsule Network-Based Band Selection for Underwater Hyperspectral Imagery." IEEE Transactions on Geoscience and Remote Sensing (2025).

[20] Xiaoxia, Zhang, and Zhang Xia. "Attention-based deep convolutional capsule network for hyperspectral image classification." IEEE Access 12 (2024): 56815-56823.

[21] Yasir, Muhammad, Shanwei Liu, Xu Mingming, Jianhua Wan, Saied Pirasteh, and Kinh Bac Dang. "ShipGeoNet: SAR image-based geometric feature extraction of ships using convolutional neural networks." IEEE Transactions on Geoscience and Remote Sensing 62 (2024): 1-13.

[22] Wang, Nian, Aitao Yang, Zhigao Cui, Yao Ding, Yuanliang Xue, and Yanzhao Su. "Capsule attention network for hyperspectral image classification." Remote Sensing 16, no. 21 (2024): 4001.

[23] Yu, Zijian, Tingyu Xie, Qibing Zhu, Peiyu Dai, Xing Mao, Ni Ren, Xin Zhao, and Xinnian Guo. "Aquatic plants detection in crab ponds using UAV hyperspectral imagery combined with transformer-based semantic segmentation model." Computers and Electronics in Agriculture 227 (2024): 109656.

[24] Manikandan, Dhana Lakshmi, and Sakthivel Murugan Santhanam. "Underwater species classification using deep learning technique." Rev. Română De Informatică și Autom 34 (2024): 7-20.

[25] Pavithra, S. "An efficient approach to detect and segment underwater images using Swin Transformer." Results in Engineering 23 (2024): 102460.

[26] Chen, Gangqi, Zhaoyong Mao, Kai Wang, and Junge Shen. "HTDet: A hybrid transformer-based approach for underwater small object detection." Remote Sensing 15, no. 4 (2023): 1076.

[27] Zhang, Xuming, Yuanchao Su, Lianru Gao, Lorenzo Bruzzone, Xingfa Gu, and Qingjiu Tian. "A lightweight transformer network for hyperspectral image classification." IEEE Transactions on Geoscience and Remote Sensing 61 (2023): 1-17.

[28] Prasenan, Pooja, and C. D. Suriyakala. "A Study of Underwater Image Pre-processing and Techniques." In Computational Vision and Bio-Inspired Computing: Proceedings of ICCVBIC 2021, pp. 313-333. Singapore: Springer Singapore, 2022.

[29] Vijayalakshmi, M., and A. Sasithradevi. "A comprehensive review on deep learning architecture for pre-processing of underwater images." SN Computer Science 5, no. 5 (2024): 472.

[30] Huang, Guoheng, Junwen Zhu, Jiajian Li, Zhuowei Wang, Lianglun Cheng, Lizhi Liu, Haojiang Li, and Jian Zhou. "Channel-attention U-Net: Channel attention mechanism for semantic segmentation of oesophagus and esophageal cancer." IEEE Access 8 (2020): 122798-122810.

[31] Zhao, Peng, Jindi Zhang, Weijia Fang, and Shuiguang Deng. "SCAU-net: spatial-channel attention U-net for gland segmentation." Frontiers in Bioengineering and Biotechnology 8 (2020): 670.

[32] Guo, Changlu, Márton Szemenyei, Yangtao Hu, Wenle Wang, Wei Zhou, and Yugen Yi. "Channel attention residual u-net for retinal vessel segmentation." In ICASSP 2021-2021 IEEE international conference on acoustics, speech and signal processing (ICASSP), pp. 1185-1189. IEEE, 2021.

[33] Jacob, I. Jeena. "Performance evaluation of caps-net based multitask learning architecture for text classification." Journal of Artificial Intelligence 2, no. 01 (2020): 1-10.

[34] SENGUL, Sumeyra Busra, and Ilker Ali OZKAN. "New Architecture in Deep Learning: Capsule Networks (CapsNet)." International Research in Engineering Sciences III (2022): 245-262.

[35] Han, Kai, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang et al. "A survey on vision transformer." IEEE transactions on pattern analysis and machine intelligence 45, no. 1 (2022): 87-110.

[36] Khan, Salman, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. "Transformers in vision: A survey." ACM Computing Surveys (CSUR) 54, no. 10s (2022): 1-41.

[37] Chennu, Arjun; Rashid, Ahmad Rafiuddin; den Haan, Joost; de Beer, Dirk (2020): Taxonomically annotated underwater hyperspectral and color images of coral reef transects from Curaçao [dataset]. PANGAEA, https://doi.org/10.1594/PANGAEA.911300

[38] Hong, Danfeng, Zhu Han, Jing Yao, Lianru Gao, Bing Zhang, Antonio Plaza, and Jocelyn Chanussot. "SpectralFormer: Rethinking hyperspectral image classification with transformers." IEEE Transactions on Geoscience and Remote Sensing 60 (2021): 1-15.

[39] Roy, S. K., G. Krishna, S. R. Dubey, and B. B. Chaudhuri. HybridSN: Exploring 3D–2D CNN Feature Hierarchy for Hyperspectral Image Classification. IEEE Geoscience and Remote Sensing Letters, 17(2), 277–281, 2020.

Effectiveness of Capsule Networks in Detecting Deepfakes Instead of Traditional CNNs

M. C. Weerawardana, T. G. I. Fernando
Department of Computer Science, University of Sri Jayewardenepura, Nugegoda, Sri Lanka

***Abstract*—As artificial intelligence has advanced, computer-generated fake content has become increasingly prevalent. Deepfake is an advanced fake creation generated using deep learning-based technologies, and deepfake images, videos, and voices have spread rapidly. The distinction between real and fake content is very hard for the naked eye, making reliable detection essential. Existing deepfake detection methods have achieved success, but still face limitations in keeping pace with the rapid evolution of deepfake generation techniques. In particular, CNN-based approaches may require a large number of parameters and may not fully capture spatial hierarchies. In this research, we investigate whether Capsule Networks can provide an effective and parameter-efficient alternative for deepfake detection. We proposed four different Capsule Network architectures by altering the size, complexity, and configuration. A comparative analysis is conducted against state-of-the-art Capsule models and CNN models across various datasets, utilizing AUC% and the number of parameters as evaluation criteria. For transparency, we note that some baseline CNN and Capsule models follow the training protocols and datasets reported in their original studies, which may differ across implementations. Our experimental results show that the proposed Capsule Network models achieved over 98% AUC% on the evaluated datasets while using fewer parameters than several CNN-based models. These findings suggest that Capsule Networks exhibit greater efficacy in detecting deepfakes compared to traditional CNN-based methods and represent a promising direction for future research.**

***Keywords*—Deepfake; deep learning; capsule network; convolutional neural network**

I. Introduction

Currently, we are exposed to a new generation fueled by artificial intelligence (AI). Day-by-day, AI generates novel creations that become integrated into the world. AI-related applications and tools play different roles in social media, websites and the Internet. Among many applications in AI, deepfake [2] is a significant and critical technology. Deepfakes involve digitally reenacting and manipulating facial appearances, utilizing deep neural networks to swap faces [1] with someone else's likeness. Deepfakes are generated in a complex process using deep neural networks. At present, lots of deepfake-generated tools like GAN [3], FaceSwap-GAN [4], FaceApp [5], Face2Face [6], and DeepFake [7] are available freely. Deepfake generates fake images, voices, and videos. The speciality of these deepfake-generated forgeries is as real as seen by the naked eye. Deepfake creation tools have been developed with new technologies and release highly accurate and quality outputs. The produced fake information can mislead society by spreading fake news, defaming celebrities and public figures, creating pornographic videos for blackmail, or perpetrating financial fraud [2].

The challenge lies in the fact that the human eye may struggle to discern between real and fake content within deepfakes. So, deepfake detection is a big challenge because synthetic forgeries are deceiving the naked eye. Many researchers have presented different approaches to detecting deepfakes during the last few years. These methods can be categorized into several technologies like feature-based [8], artifacts [9], inconsistencies of physiological signals [10], [21] and deep convolutional approaches [11], [12], [13]. These existing deepfake detection methods have achieved success but come with significant limitations. The key limitation in existing deepfake detection research is the insufficient examination of cross-dataset generalization. Many prior studies report performance primarily under intra-dataset train-test splits, which may lead to overly optimistic estimates of real-world effectiveness. In practical scenarios, however, detection must handle previously unseen manipulation techniques, varying compression levels, and diverse data distributions. The existing detection methods have failed to respond to the new technologies with the rapid evolution of deepfake creation, leaving a gap where deepfakes outpace detection techniques. Additionally, there are no flawless detection methods, particularly in Convolutional Neural Network (CNN)-based approaches. To avoid this situation, we moved to a new way rather than traditional methods.

The increasing realism of deepfakes poses significant challenges to digital security and misinformation control. Traditional CNN-based detection methods often struggle to capture subtle structural inconsistencies in manipulated content. Motivated by the need for more robust detection methods, this study explores the potential of Capsule Networks, which are better suited to model spatial and hierarchical relationships, to improve deepfake detection performance. This experimental protocol enables a rigorous analysis of robustness to domain shift.

The main contribution of this study is the development and comparative evaluation of multiple Capsule Networks for deepfake video detection, demonstrating that Capsule Networks are more effective than traditional CNN-based models. We conduct a comparative analysis of the performance of the proposed Capsule Network models against state-of-the-art Capsule models, and traditional CNN-based models across various datasets utilizing AUC% and the number of parameters as the criteria for comparison. So, to achieve this target, we utilized multiple Capsule Network models for the experiment,

altering several parameters, changing size, complexity, and configuration, etc. For the comparison, we selected five existing CNN methods. They were MesoNet [15], Head Pose [17], Visual Artifact [8], XceptionNet [16], and Multitask-learning [19] architectures, which were fine-tuned on the 140K Real and Fake Faces Kaggle dataset [20]. These models have been implemented on several technologies. We used five different datasets: UADFV [17], DF-TIMIT [21], FaceForensics++ [22], DFDC [23][24], and Celeb-DF [25]. To address the generalization problem, we conduct systematic cross-dataset evaluations by training the proposed models on a single dataset and assessing the performance across multiple unseen datasets.

Our study is structured as follows: in the first part, we present a brief introduction about deepfakes, along with their background, problem, motivation, and contribution, and outline the structure of our study. The second part represents a literature summary on deepfake detection methods and Capsule Networks. Next, we detail our methodology, including our proposed Capsule models and datasets used. In section four, we explore our implementations and detail the results obtained throughout the experiment. Lastly, we summarize the conclusions drawn.

II. Related Work

Under the related works, we cover the existing best deepfake detection methods and explore the improvements and unique advantages of the Capsule Network models on deepfake detection.

A. Deepfake Detection Methods

Recently, deepfake is a more concerned phenomena due to the rapid spread of various deepfake creations [2]. Deepfake generation and detection are developing in parallel. From 2018 onwards, there has been increased socialization of forgery arts, applications, fake video creations, and deepfake generation tools. At the same time, financial frauds, revenge on politicians and popular people, pornographic videos of celebrities and blackmails, etc. started to increase gradually [2]. With the growth of misleading information, attention has been drawn to finding deepfake detection methods. Recently, a considerable number of research publications regarding deepfake detection have been available.

Numerous researchers have proposed various approaches for deepfake detection, including feature-based [8], artifact analysis [9], inconsistencies in physiological signals [10], [21], and deep convolutional methods [11], [12], [13]. Deep convolutional approaches may be quite reliable because they do not depend on feature limits. XceptionNet architecture [16] has been successfully used in many approaches [22], [26], [27]. In 2020, Kumar et al. [28] presented a successful model with an accuracy of 99.2% on the CelebDF dataset using XceptionNet architecture as a feature extractor. Several papers have presented deepfake video detection methods using Recurrent neural networks [30], [32] or LSTMs [31], [33]. Lima et al. [29] combined several 2D and 3D convolutional layers in their model to leverage the temporal dimension. Very recently, Google has presented a watermark-based detection tool called SynthID [34] to identify AI-created images. But our opinion is that the Watermark technique is not very successful because the sources of image creation are varied. This method is not valid for widespread usage because it is restricted only to limited resources like Google Cloud and some specific customers, such as Vertex AI [34].

Deep learning methods, machine learning methods, computer vision methods, multimodal and scalable techniques can be used for deepfake detection. The problem lies in deepfake creation, which continuously evolves with new technologies and advances day-by-day. Therefore, the state-of-the-art deepfake detection methods may fail in the face of new technical advances. Recently, a few researchers in the domain of deepfakes have explored the use of Capsule Networks.

B. Capsule Network Models and Their Advantages

According to the literature, Capsule Networks are not a novel concept in the field of deepfake detection. Hinton et al. [35] first introduced the concept of the Capsule Networks in 2011. Then again, Hinton et al. [36] introduced the training and propagation of capsules using the routing by agreement algorithm in 2017. Capsule Networks are composed of vectorial units that encode information in multiple dimensions, including pose information, unlike scalar units in Convolutional Neural Networks. Fig. 1 presents the general architecture of the Capsule Network employed in this work. The model processes the input through a primary capsule layer that encodes local features as vector representations. These capsules are then aggregated in a higher-level capsule layer using a routing mechanism to model hierarchical relationships. The resulting capsule outputs are used in the decision stage, and the final predictions are obtained at the output layer.

Through experiments, it has been shown that Capsule Networks could improve upon the limitations of traditional CNN methods in detecting deepfakes. Hinton et al. [35] have found some drawbacks in the use of a pooling layer in a CNN architecture. Mainly, the authors [35] mentioned that information like the position of an object is lost in the pooling layer. In the CNN model, neural networks strongly consider key elements like the eyes of a face rather than its position because the pooling layer deletes other information like position. But position is very important to increase the accuracy of the method. On the other hand, vectors help to collect more information about the object, like position, color, texture, and skewness, etc. and help to get more accurate predictions. In 2018, Shahroudnejad et al. [37] identified an additional drawback: the need for more training data when employing CNN architectures with a pooling layer. Also, the authors in [37] described that CNNs require longer duration training data and Capsule Networks need a wider range of training data. Recent research has also explored transformer-based and hybrid architectures for deepfake detection. Transformer backbones such as Swin Transformer have been proposed to capture global context and spatial consistency, improving robustness across datasets in deepfake video classification [44]. Furthermore, hybrid models integrating CNNs with vision transformers have been developed to leverage local feature extraction and global attention mechanisms for enhanced detection efficacy [45]. However, these approaches often require large-scale training data and substantial computational resources, which can limit their practical development.

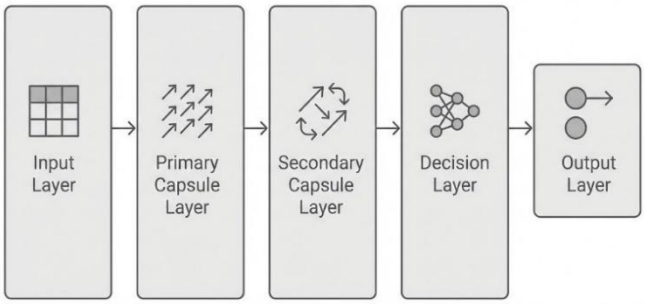


Fig. 1. Common architecture of the capsule network.

Jindong et al. [38] have presented the advantages and disadvantages of the Capsule Network architecture. Advantages are the ability to generalize, a small number of parameters needed, various transforms applied to the image and resistance to attacks. According to the authors [38], the disadvantages are that the capsules have limited learning capabilities and sometimes Capsule Networks show poorer assessment than traditional CNNs on large datasets like ImageNet [39].

Recently, Nguyen et al. [40] have improved a Capsule-Forensics Network [40], [41] for deepfake video detection. This method has shown higher performance compared to the state-of-the-art methods while utilizing less training data, a smaller number of parameters and lower computational cost, etc. [40], [41]. These superior advancements across multiple domains have motivated them to turn towards the Capsule Networks.

C. Improvements of the Capsule-Forensics Architecture

In 2019, Nguyen et al. [40] improved the Capsule-Forensics architecture for deepfake video detection. In the following paragraph, we briefly described the configuration and shortcomings of the Capsule-Forensics architecture.

Capsule-Forensics architecture has been configured along with a part of the pre-trained VGG-19 model [14], which has been used to extract the features and has been transformed into primary capsules. The authors [40] have included a 'stat pooling layer' to calculate the mean and standard deviation for filters. The Capsule-Forensics architecture has utilized from 3 to 10 primary capsules, each with 8 features, to extract relevant features. Next, the architecture was configured along with a dynamic routing algorithm. Here, the dynamic routing algorithm performs two iterations, calculating the sum of output vectors from the primary capsules. The authors [40] have used two output capsules (for deepfake and real with 4 x 1 vectors). The resulting vector has passed through the SoftMax layer. The authors [40] used an average function to predict the probability values of deepfakes. Capsule-Forensics architecture has used 3.9 million parameters for the whole network. Nguyen et al. [40] showed 93.11% accuracy on binary classification with FaceForencis++ dataset. Tolosana et al. [42] utilized the same architecture and demonstrated a 99.52% AUC on the FaceForensics++ dataset and 82.46% AUC on the Celeb-DF dataset. This architecture has exhibited superior performance by utilizing fewer parameters, a smaller amount of training data and lower computational costs compared to state-of-the-art methods. Hence, many research domains have started incorporating Capsule Networks, particularly within the field of deepfakes.

But we identified that the Capsule-Forensics architecture [40], [41] has several shortcomings. The authors [40], [41] and [42] utilized a small number of primary capsules, typically ranging from 3 to 10. It appears that their configuration might be considered weak since primary capsules hold significant importance in computing the values of secondary capsules. Additionally, the authors of [40] and [42] employed only 2 iterations for the dynamic routing algorithm. A higher number of iterations can significantly impact the accuracy of the final prediction. The Capsule-Forensics architecture also incorporated the 'statistical pooling layer.' According to the principles of Capsule Networks, the pooling layer's function involves discarding a considerable amount of valuable information, potentially leading to a decrease in the performance of the capsules. We believe that this architecture has the potential to achieve higher performance by addressing these existing shortcomings and implementing necessary improvements.

III. Methodology

In the research carried out in this study, our primary objective was to implement multiple Capsule Network architectures for detecting deepfake videos, addressing the existing shortcomings of the Capsule-Forensics architecture. We aimed to compare their performances and evaluate the effectiveness of modifications introduced to address the earlier discussed shortcomings. To achieve this, we centered our attention on the Capsule-Forensics architecture introduced in 2019. While retaining the fundamental elements of the Capsule-Forensics network, we made essential modifications necessary to implement diverse Capsule Network architectures for deepfake detection. We altered the size, complexity, and configuration of each Capsule architecture from its original specifications. Furthermore, we conducted a comparative analysis of the performance of Capsule Network models and CNN models for deepfake detection. Our objective was to demonstrate that Capsule Networks possess greater efficacy in detecting deepfakes compared to traditional CNN methods.

We configured four different Capsule Network models for the analysis, elaborated upon in detail in the following section. During experimentation, the DFDC dataset was utilized for training the Capsule models. To compare the performance, we employed five state-of-the-art CNN models: specifically, the MesoNet [15], Head Pose [18], Visual Artifact [17], XceptionNet [16], and Multitask-learning [19] architectures. All these models are feature-based deepfake video detection methods. They were initially pre-trained on the ImageNet dataset [39], and their output layers were subsequently replaced with fully connected decision layers for the purposes of this study. Initially, we fine-tuned and retrained these models using the 140K Real and Fake Faces Kaggle dataset during the training phase. To conduct a comparative analysis, we assessed the performance of the mentioned CNN models across five datasets: UADFV, DF-TIMIT, FaceForensics++, DFDC, and Celeb-DF dataset. Subsequently, we analyzed the results and compared the performance of each Capsule model with that of the state-of-the-art CNN models in detecting deepfake videos. The Area Under Curve (AUC) values are presented in the results and discussion section for comparison analysis.

A. Proposed Capsule Network Architecture

Our primary objective was to demonstrate that Capsule Networks possess greater efficacy in detecting deepfakes compared to traditional CNN methods. For this task, we need to implement a better Capsule Network architecture and to show better performance instead of the state-of-the-art CNN methods. We tried to rectify the few shortcomings present in the state-of-the-art Capsule models for detecting deepfake videos. We started with the state-of-the-art Capsule-Forensic architecture [40], enhancing identified shortcomings, and subsequently assessed various Capsule Network architectures listed below to enhance their performance. While keeping the fundamental elements of the Capsule-Forensics architecture utilized in [40] as the basis, we made some necessary modifications to our Capsule Network architectures described as follows.

We modified their size, complexity, and configuration as necessary. We require an existing pretrained model to extract the necessary features for the primary capsules in our Capsule models. We chose the VGG-19 CNN architecture [14] as a pretrained model. In the model configuration, the larger models were set up with the initial 8 convolutional layers and totaled around 1.6 million parameters. The smaller models were composed of the initial 3 convolutional layers, resulting in a total of around 0.4 million parameters. The authors of [40] and [42] utilized a limited number of primary capsules, typically ranging from 3 to 10. Their configuration appears weak because primary capsules play a crucial role in computing output values for the secondary capsules. Secondary capsules are of great importance to predict the correct values. We assumed that a large number of primary capsules may eliminate noise and prevent poor predictions. Thus, we conducted experiments with varying numbers of primary capsules, starting from 1,800 primary capsules.

Initially, we used two secondary capsules for each class. Later, we conducted tests with different quantities of secondary capsules for each class. In most instances, there were 16 capsules of 16x1 dimensions used. The fully connected decision layer was used to distinguish real and deepfakes. The number of iterations in the dynamic routing algorithm plays a role in determining the weights assigned to the secondary capsules. The better weights of the secondary capsules ensure better accuracy of the predictions. The authors of [40] and [42] have used only 2 iterations in their research. Instead of using only 2 iterations, we implemented a broader range of iterations in the dynamic routing algorithm, employing a minimum of 3 and a maximum of 5 iterations. Finally, we obtained a comparative analysis of each architecture's performance in comparison to the state-of-the-art methods. Below is the list of all the implemented Capsule Network models used in our experiment. Fig. 2 shows a graphical representation of the architectural configuration of each model.

- Capsule Network model 1: This model was configured with 1,800 primary capsules, each consisting of 8 dimensions, and 2 secondary capsules, each comprising 16 dimensions.
- Capsule Network model 2: This model was designed with 7,200 primary capsules, each having a length of 8, and 10 secondary capsules, each with a length of 16. Additionally, a fully connected layer was incorporated as the decision maker.
- Capsule Network model 3: This specific Capsule Network model comprised 1,568 primary capsules, each having a length of 8, and 16 secondary capsules, each with a length of 16. During the training process, the dynamic routing algorithm utilized 3 and 5 iterations. Additionally, a fully connected layer was included for decision-making purposes.
- Capsule Network model 4: This model shares similarities with Capsule Network model 3, except for model 4, which includes 8 secondary capsules instead of 16.

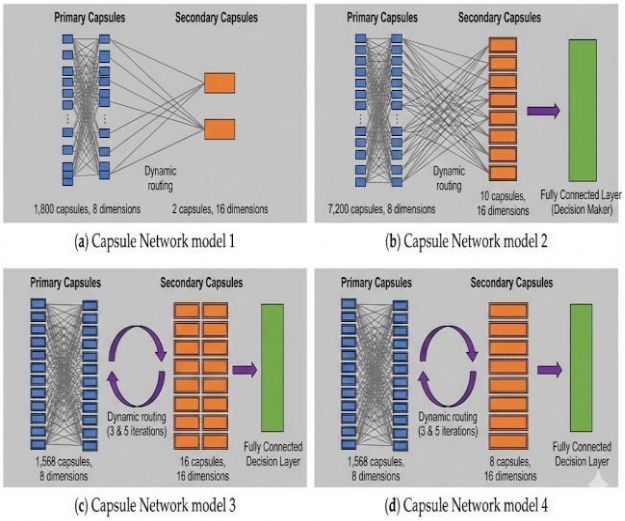


Fig. 2. Proposed capsule network architecture.

B. Dataset Used

When selecting a dataset for deepfake detection, several crucial characteristics should be considered, including short video length (10-15 seconds), focus on close-up human faces, confirmation of consent from actors involved, the total number of videos available, and the overall quality of the video content. These factors are most affected in determining the accuracy and reliability of a deepfake detection method. Currently, many open-source free deepfake datasets are available. We chose five distinct datasets that were very suitable for our experiment: UADFV, DF-TIMIT, FaceForensics++, DFDC, and Celeb-DF. The dataset contained both real and deepfake video clips, each lasting around 5 to 10 seconds in duration. We need to break them down into individual frames for further analysis and processing. Each frame is fed into the detection architecture. To create frames, we selected multiple video clips from both real and deepfake sources found within the DFDC, UADFV, DF-TIMIT, FaceForensics++, and Celeb-DF datasets.

When selecting video clips, we specifically targeted those that provided better focus on the face region. This consideration was crucial since deepfakes primarily manipulate and alter facial features. We employed "Windows Movie Maker" to split the video clip into individual frames. The first 256 frames of each video clip were extracted and resized to 299x299. Thus, we aimed to exclusively extract the facial regions from each frame. For our experiment, after splitting the video, we adopted an

appropriate preprocessing pipeline as follows to extract the necessary features from each frame. Initially, we performed face detection, followed by detecting facial landmarks, eliminating the background, and ultimately resizing the image as part of our preprocessing pipeline. These steps were repeated for all 256 frames within each of the selected video clips. Here, we utilized DLib-ml [43] for detecting the face region and identifying facial landmarks. DLib-ml is a machine learning based toolkit designed for face detection, developed by King in 2009 [43]. We applied DLib-ml for every frame, extracting 68 landmarks from each frame. The resulting landmarks were then mapped onto a standard 3D facial landmark model consisting of 68 points.

We used two datasets for the training process. The 140K Real and Fake Faces Kaggle dataset was used for the CNN models, while the Deepfake Detection Challenge (DFDC) dataset served as the dataset for the Capsule Network models. The DFDC dataset was chosen for training the Capsule Network models because its video-based deepfake samples capture complex spatial relationships, pose variations, and manipulation artifacts that align with the Capsule Network's ability to model hierarchical part–whole representations; therefore, DFDC enables the Capsule Network to exploit its architectural advantages more fully, while the 140K Real and Fake Faces Kaggle dataset remains more suitable for CNNs that perform effectively on static image classification tasks. Below are the six distinct datasets we selected for our research:

- UADFV dataset: This dataset includes 49 original videos collected from YouTube and 49 fake videos generated using deep neural networks.
- DF-TIMIT dataset: This dataset includes 640 real and fake videos. Deepfake videos were generated using FaceSwap-GAN. This dataset is divided into the DF-TIMIT-LQ and DF-TIMIT-HQ datasets.
- FaceForensics++ (FF++) dataset: This dataset consists of 1,000 videos from both real and fake sources. Real videos were collected from YouTube, and Deepfake videos were generated using FaceSwap.
- Deepfake Detection Challenge (DFDC) dataset: This dataset encompasses 4,113 fake videos, which were generated using 1,131 real videos. DFDC used 66 individuals of diverse age, gender, and culture.
- Celeb-DF dataset: This dataset contains 590 real videos and 5,639 fake videos, which were generated using FaceSwap technology.
- 140K Real and Fake Faces Kaggle dataset: This dataset consists of 140,000 human face images, including 70,000 real faces and 70,000 fabricated faces, which were generated using StyleGAN. This dataset is segmented into three subsets: 100,000 images for training, 20,000 for testing, and 20,000 for validation purposes.

C. Model Training

Our primary purpose was to evaluate the effectiveness of both Capsule Network models and modified versions thereof in comparison to state-of-the-art CNN methods regarding their performance. We conducted repeated training and testing of the models with the intention of achieving higher accuracy.

In the training process, we utilized the DFDC dataset for the Capsule models. We selected 10^{-3} as the learning rate value, maintaining a learning rate decay factor within the range 0.5 and 0.8. For the model training, we opted for 15 epochs, utilized the Adam optimizer, set a batch size of 16, and employed 3 and 5 iterations for the dynamic routing algorithm as the required parameters.

We selected five established state-of-the-art CNN models, which were previously pretrained on the ImageNet [39] dataset. During our experiment, we fine-tuned these models in accordance with our specific requirement of classifying real and fake videos. Here, we trained the CNN models on the 140K Real and Fake Faces Kaggle dataset, with a fixed learning rate of $5x10^{-4}$, a learning rate decay factor set to 0.92, running for 10 epochs and utilizing the Adam optimizer with a batch size of 64. Also, we selected Binary Cross-Entropy as the loss function.

In scripting the programs, we utilized Jupyter notebooks with Python. Google Collaboratory was instrumental in executing all our notebooks. In managing the extensive dataset, we opted to partition it into three distinct categories to facilitate training, testing and validation. Specifically, we allocated 100,000 images for the training set and reserved 20,000 images each for the testing and validation sets. These images were drawn from the 140K Real and Fake Faces Kaggle dataset and the DFDC dataset. Saved models were evaluated with five distinct datasets and compared their performance using the Area Under the Curve (AUC%) values.

IV. Results and Discussion

We used Windows Movie Maker to split the video clip into the first 256 frames. Each frame was processed using the "DLib-ml" tool to detect the face region and facial landmarks, marking 68 landmarks of each frame. The resultant landmarks were subsequently mapped onto a standard 3D facial landmark model consisting of 68 points. Fig. 3 shows the detected facial landmarks identified by the "DLib-ml" tool. In Fig. 3, the original real image is shown in Image (A). Image (B) displays the facial landmarks (indicated by red dots) of the central face region of the real face, while Image (C) displays the facial landmarks (indicated by blue dots) of the entire face region of the real face. All frames have been resized to 299x299. The first 256 frames of every video clip underwent this process.



Image (A) – Original Image (B) – Red dots Image (C) – Blue Dots

Fig. 3. Facial landmarks detection.

A. Performance of the Different Deepfake Video Detection Methods

In this research, our objective is to conduct a comparative analysis of the performance of the proposed Capsule Network models against state-of-the-art Capsule models and CNN models across various datasets. We aim to utilize AUC% and the number of parameters as the criteria for comparison.

Firstly, we determined the superior performance of the Capsule Networks over traditional CNN models. Secondly, we evaluated whether our Capsule Network models could achieve comparable or superior performance compared to state-of-the-art Capsule Network (CapsNet) models [41], [42] and CNN models. In this experiment, AUC% values were employed for evaluating the performance of each model. Table I showcases the results obtained from the four proposed Capsule Network models alongside the selected state-of-the-art CNN methods, evaluated across five distinct datasets. The proposed Capsule Network model 1, as depicted in Table I, attained the highest AUC% score of 99.55% on the FaceForensics++ dataset, outperforming existing CNN-based approaches and other models. Table I reveals that all detection methods, apart from Capsule Network models, encounter the most significant challenge in detecting deepfakes within the Celeb-DF dataset. Their performance rates with this dataset have remained below 70% to date. However, the proposed Capsule models outperform others, specifically on Celeb-DF. The following Fig. 4 enhances the clarity of the results in Table I.

Fig. 4 shows the grouped bar chart illustrating the performance of proposed Capsule Network models and state-of-the-art CNN models across different datasets. The graph shows the AUC% values of each model on the y-axis and the datasets on the x-axis. Fig. 4 clearly visualizes the higher AUC% values of the proposed Capsule models 1 and 2 across the five different datasets.

The Capsule Network models 3 and 4 were not tested on the UADFV, DF-TIMIT and DFDC datasets. The Capsule Network models 3 and 4 were designed specifically for comparison to the state-of-the-art CapsNet models and other proposed models. Therefore, models 3 and 4 were solely tested on the FaceForensics++ and Celeb-DF datasets, as shown in Table II.

Moreover, the difficulty levels in detecting fakes using the aforementioned methods vary across different datasets, such as DFDC and Celeb-DF. This suggests that the contents of recently developed datasets are still difficult to detect due to the varying collection of deepfake videos. Among the CNN-based models listed in Table I, XceptionNet exhibited the best performance across all datasets, attaining a notable accuracy of 99.0% with the FaceForensic++ dataset. Nonetheless, the proposed Capsule models we tested exhibited impressive AUC% scores, surpassing 99% on the FaceForrensics++ dataset, with some even exceeding 99.5%.

Based on Table I, it's evident that the proposed Capsule Network models achieve competitive performance over the existing CNN models across the datasets evaluated in this study.

Table II shows the AUC% values of the proposed Capsule models and state-of-the-art Capsule models on both the FaceForensics++ and Celeb-DF datasets. We evaluated the performance using AUC% and the number of parameters. According to Table II, it is noticeable that the majority of Capsule Network models, including both our experimental models and state-of-the-art models, have reached an AUC% score exceeding 95%. Capsule Network model 1 has achieved an AUC% score of 99.55%, surpassing the AUC% scores of existing Capsule Network approaches on the FaceForensics++ dataset. Further, except for Capsule Network model 1, the proposed Capsule Network models showed better performance compared to state-of-the-art Capsule Network architectures on the Celeb-DF dataset.

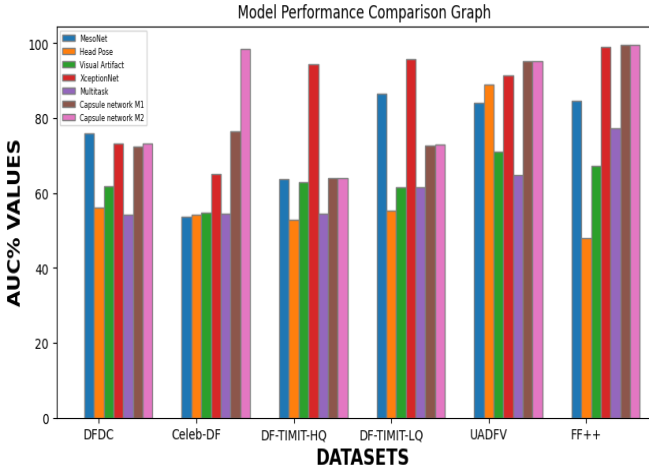


Fig. 4. Comparative performance of proposed capsule network models and CNN models across different datasets.

TABLE I. COMPARATIVE ANALYSIS BETWEEN THE PROPOSED CAPSULE NETWORK MODELS AND STATE-OF-THE-ART CNN-BASED MODELS ACROSS MULTIPLE DATASETS FOR PERFORMANCE EVALUATION

Model	Dataset					
	DFDC	*Celeb-DF*	*DF-TIMIT*		*UADFV*	*FF++*
			HQ	*LQ*		
MesoNet [15]	76.0	53.7	63.8	86.5	84.2	84.5
Head Pose [18]	56.0	54.3	52.8	55.3	89.0	47.9
Visual Artifact [17]	61.8	54.9	63.0	61.5	71.0	67.2
XceptionNet [16]	73.2	65.1	94.3	95.8	91.3	99.0
Multitask [19]	54.1	54.5	54.6	61.6	64.9	77.2
Capsule Network model 1 (Our model)	72.5	76.43	63.9	72.8	95.1	99.55
Capsule Network model 2 (Our model)	73.1	98.4	64.0	72.9	95.2	99.51

TABLE II. COMPARATIVE ANALYSIS BETWEEN THE PROPOSED CAPSULE NETWORK MODELS AND THE STATE-OF-THE-ART CAPSULE NETWORK MODELS ON THE FACEFORENSICS++ AND CELEB-DF DATASETS

Model	AUC %		No. of parameters
	FaceForensics++	Celeb-DF	
Nguyen et al- CapsNet [41]	93.11	57.50	3.9 M
Tolosana et al- CapsuleNet [42]	99.52	82.46	3.9 M
Capsule network model 1	99.55	76.43	5 M
Capsule network model 2	99.51	98.40	16 M
Capsule network model 3: 3 iterations	98.40	97.76	6 M
Capsule network model 3: 5 iterations	99.00	99.00	6 M
Capsule network model 4	98.31	97.46	4 M

Tolosna et al. [42] introduced an architecture that attained 82.46% AUC% on the Celeb-DF dataset and 99.52% on the FaceForensics++ dataset. In contrast, our proposed Capsule model 3 demonstrated the best performance of 99% on the Celeb-DF dataset and 99% on the FaceForensics++ dataset.

Tolosna et al. [42] introduced an architecture that attained 82.46% AUC% on the Celeb-DF dataset and 99.52% on the FaceForensics++ dataset. In contrast, our proposed Capsule model 3 demonstrated the best performance of 99% on the Celeb-DF dataset and 99.00% on the FaceForensics++ dataset.

Fig. 5 illustrates the grouped bar chart showing the AUC% values of proposed Capsule Network models and state-of-the-art Capsule Network models on the FaceForensics++ and Celeb-DF datasets. The graph shows the AUC% values of each model on the y-axis and the datasets on the x-axis.

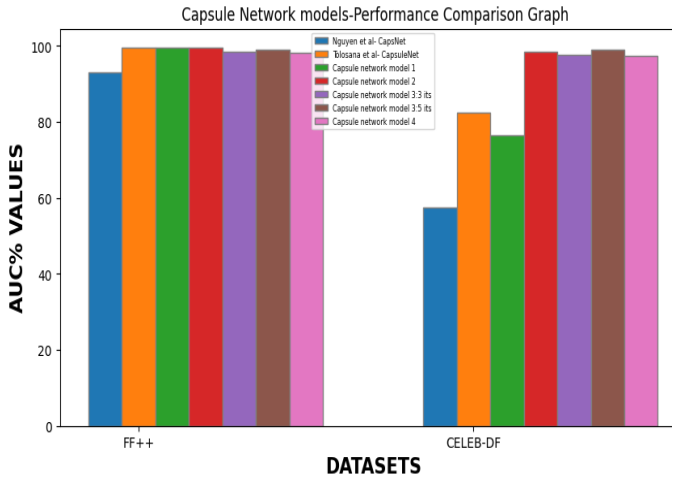


Fig. 5. Comparative performance of proposed capsule network models and state-of-the-art capsule network models on the faceforensics++ and celeb-DF datasets.

Fig. 5 shows a clear comparison of the performance of each model on each dataset. According to Fig. 5, the AUC% values of our proposed Capsule models display higher values than state-of-the-art Capsule models on both the FaceForensics++ and Celeb-DF datasets.

Additionally, we experimented with several iterations of the dynamic routing algorithm. Increasing the number of iterations in the Capsule Network can lead to higher performance. For instance, in Capsule Network model 3, transitioning from 3 iterations to 5 iterations within the same architecture enhanced the performance from 98.40 to 99.00 on the faceForensics++ dataset and from 97.76% to 99.00% on the Celeb-DF dataset, respectively. Nevertheless, a higher number of iterations also leads to increased prediction time, rendering it unsuitable for real-time tasks.

In addition, we conducted an experiment that demonstrates the superiority of Capsule Networks over Convolutional Networks. For instance, Capsule Network models 3 and 4 utilized 6 million and 4 million parameters, respectively, whereas the CNN models employed over 10 million parameters. Despite processing fewer parameters, the Capsule Network models achieved results with over 97% AUC%, surpassing the performance of the CNN models. Through experimentation, we found that maintaining over 1.9 million parameters in each Capsule Network model allowed them to achieve superior performance compared to CNN models.

The following Fig. 6 shows the line chart comparing the number of parameters used for the Capsule models and CNN models. The graph will illustrate the model complexity and resource requirements between the two types of models. Capsule Networks achieve consistently high AUC scores across all parameter sizes, demonstrating strong performance even with fewer parameters. In contrast, CNN models show a gradual improvement in AUC as the number of parameters increases, indicating a higher dependence on model size. Notably, CNNs require substantially more parameters to approach the performance level of Capsule Networks. These results highlight the superior parameter efficiency and robustness of Capsule Networks compared to conventional CNN architectures.

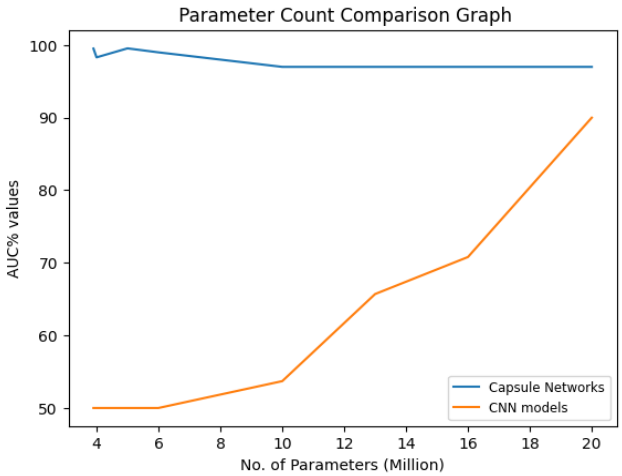


Fig. 6. Comparison of the number of parameters used for both capsule network models and CNN models.

The reported results are presented as single-run AUC values obtained under a fixed and consistent experimental protocol. Although additional statistical validation, such as repeated trials or confidence interval estimation, would further strengthen the robustness assessment, the observed performance trends remain

stable across five cross-dataset evaluations. All models were tested on the same five unseen datasets to assess generalization; Capsule Networks were trained on the DFDC dataset, while CNN models were trained on the 140K Real and Fake Faces Kaggle dataset due to experimental design considerations. Consequently, the observed performance improvements reflect empirical results under the current experimental setup rather than a strict isolation of architectural superiority or generalization capability. Future work will include controlled experiments and extended statistical analysis to more rigorously evaluate architectural generalization and performance stability.

We encountered several challenges during the model training process, particularly in terms of the amount of time it takes to train the models. Achieving higher accuracy requires more than 100,000 data points for training, which may be time-consuming. The time required for successful training can vary depending on factors such as hardware resources, computational cost, size of the dataset, number of training epochs, and type of neural network architecture of the detection model. In our experiment, we used Google Collab online for training and for the evaluation process. We used Windows Movie Maker for the frame extraction. While this tool is not conventional for research, the frame extraction procedure was consistent across all videos, ensuring that the experimental results are reproducible. For future studies, the same extraction procedure can be implemented using standard tools such as OpenCV, which would provide fully reproducible frame sequences.

V. Conclusion

Deepfake-generated fake images, voices, and videos are rapidly spreading across various social media platforms and websites. The deepfake creation tools are highly recommended for their output accuracy, as the synthetic forgeries they produce can appear indistinguishable to the naked eye. Detecting deepfakes poses a significant challenge due to the high accuracy of these fabricated creations.

Through this study, we aimed to present a novel and robust approach to deepfake detection distinct from existing state-of-the-art methods. We identified that Capsule Networks have more effective power in detecting deepfakes instead of traditional CNN-based methods. According to our results, despite having fewer parameters, the Capsule Network models achieved results with over 98% AUC%, showing stronger performance. Our proposed Capsule Networks exhibited reduced performance degradation compared to traditional CNNs and existing Capsule (CapsNet) models, indicating improved generalization capability. We emphasize the following points and present the summarization of the research as follows.

- For the model training, we employed 15 epochs for Capsule models and 10 epochs for CNN models. However, the proposed Capsule Network models outperformed the CNN models, completing the training process in a shorter period.
- Capsule Networks can retain more information compared to traditional CNNs, even with less training data.
- Capsule Networks demonstrated stronger performance under the given experimental configuration.
- As the number of iterations of the dynamic routing algorithm increases, the performance of the Capsule models also improves, even with a smaller number of parameters.
- Existing CNN-based deepfake detection methods have shown lower accuracy, often falling below approximately 65%, primarily due to the generalizability problem. Capsule Networks, however, offer a solution that avoids the generalizability problem.

Our results show that Capsule Networks exhibit greater efficacy in detecting deepfakes compared to traditional CNN-based methods. Therefore, Capsule Networks are better suited for deepfake detection due to their capacity for generalization, utilization of fewer parameters, and the ability to handle image rotation, transformation, and dimensionality. These qualities make Capsule Networks an ideal choice for deepfakes detection. Novel methods within Capsule Networks for deepfake detection should be further improved in the near future. These findings will be a significant step forward for the new researchers in the deepfake domain.

References

[1] S. Haysom, "People are using face-swapping tech to add Nicolas Cage to random movies and what is 2018.", January 2018. [Online]. Available: https://mashable.com/2023/09/15/nicolas-cage-face-swapping deepfakes/?europe=true.

[2] M. C. Weerawardana and T. G. I. Fernando, "Deepfakes detection methods: a literature survey," *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, 2021, pp. 76-81, doi: 10.1109/ICIAfS52090.2021.9606067.

[3] B. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, Courville, A. Bengio, "Generative Adversarial Networks", 2014.

[4] Shaoanlu, "Faceswap-GAN," 2018. [Online]. Available: https://github.com/shaoanlu/faceswap-GAN.

[5] L. Guilloux, "Fakeapp," March 2019. [Online]. Available: https://www.fakeapp.org/

[6] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2face: Real-Time face capture and reenactment of RGB videos," *In CVPR*, pp. 2387–2395, June 2016.

[7] ModelIncubator, "Deepfake project - nonofficial project based on original deepfakes thread," 2018. [Online]. Available: http://www. github.com/deepfakes/faceswap

[8] F. Matern, C. Riess, and M. Stamminger, "Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations," *in 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW),* pp. 83-92, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/ 8638330/

[9] X. Zhang, S. Karaman, and S. Chang, "Detecting and simulating artifacts in GAN fake images," *preprint arXiv: 1907.06515*, 2019.

[10] D. Bitouk, N. Kumar, S. Dhillon, P. Belhumeur, and S. K. Nayar, "Face swapping: automatically replacing faces in photographs," *ACM Trans. Graph,* vol. 27, pp. 1–8, August 2008.

[11] D. Cozzolino and L. Verdoliva, "Noiseprint: a cnn-based camera model fingerprint," *arXiv preprint arXiv: 1808.08396*, 2018.

[12] K. L. Du and M. N. S. Swamy, "Neural networks and statistical learning," *Springer London*, London, 2019.

[13] T. D. Nhu, I. Na, and S. H. Kim, "Forensics face detection from gans using convolutional neural network," 2018.

[14] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: a survey of face manipulation and fake detection," *Information Fusion*, vol. 64, pp. 131–148, 2020.

[15] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "MesoNet: a compact facial video forgery detection network," *in 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE,* 2018, pp. 1–7. [Online]. Available: https://ieeexplore.ieee.org/document/8630761/

[16] F. Chollet, "Xception: deep learning with depthwise separable convolutions," *in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Honolulu, HI, USA, 2017 pp. 1800-1807.

[17] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent Head Poses," *in ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* vol. 2019-May, IEEE, 2019, pp. 8261–8265. [Online]. Available: https://ieeexplore.ieee.org/document/8683164/

[18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017, pp. 4700-4708.

[19] H. H. Nguyen, F. Fang, J. Yamagishi, and I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos," 2019. [Online]. Available: http://arxiv.org/abs/1906.06876

[20] "Real and fake detection Kaggle dataset." [Online]. Available: https://www.kaggle.com/xhlulu/140k-real-and-fake-face

[21] P. Korshunov and S. Marcel, "DeepFakes: a new threat to face recognition? assessment and detection," pp. 1–5, 2018. [Online]. Available: http://arxiv.org/abs/1812.08685

[22] B. A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and ¨ M. Niebner, "FaceForensics++: learning to detect manipulated facial images," *in 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019, pp. 1-11. [Online]. Available: http://arxiv.org/abs/1901.08971

[23] Kaggle, "Join the deepfake detection challenge (DFDC)," December 2019. [Online]. Available: https://deepfakedetectionchallenge.ai/

[24] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The DeepFake detection challenge dataset," 2020. [Online]. Available: http://arxiv.org/abs/2006.07397

[25] Y. Li, X. Yang, P. Sun, H. Qi, and S. Lyu, "Celeb-DF: a large-scale challenging dataset for DeepFake forensics," *in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020 pp. 3204-3213. [Online]. Available: http://arxiv.org/abs/1909.12962

[26] H. Dang, F. Liu, J. Stehouwer, X. Liu, and A. Jain, "On the detection of digital face manipulation," *in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

[27] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang, and C. C. Ferrer, "The DeepFake detection challenge (DFDC) dataset", 2020.

[28] BA. Kumar and A. Bhavsar, "Detecting deepfakes with metric learning," arXiv preprint arXiv:2003.08645, 2020.

[29] O. de Lima, S. Franklin, S. Basu, B. Karwoski, and A. George, "Deepfake detection using spatiotemporal convolutional networks".

[30] E. Sabir, J. Cheng, A. Jaiswal, W. AbdAlmageed, I. Masi, and P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos," *in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops,* 2019.

[31] S. Tariq, S. Lee, Simon S. Woo, "A convolutional LSTM based residual network for deepfake video detection", 2020.

[32] D. Guera and E. Delp, "Deepfake video detection using recurrent neural networks," *in 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Auckland, New Zealand, 2018, pp. 1-6.

[33] S. Dan-Cristian, B. Ionescu, "Deepfake video detection with facial features and long-short term memory deep networks"

[34] "Watermark tool to identify AI created images." [Online]. Available: https://www.geeksforgeeks.org/google-launches-watermark-tool-to-identify-ai-created-images/

[35] G. E. Hinton, A. Krizhevsky, and S. D. Wang, "Transforming auto-encoders," *In International Conference on Artificial Neural Networks*, Springer, 2011, pp. 44-51.

[36] E. Hinton, S. S. Nicholas, and F. Geoffrey, "Dynamic routing between capsules," *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, 2017.

[37] B. A. Shahroudnejad, P. Afshar, K. N. Plataniotis and A. Mohammadi, "Improved explainability of capsule networks: relevance path by agreement," *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, Anaheim, CA, USA, 2018, pp. 549-553, doi: 10.1109/GlobalSIP.2018.8646474.

[38] V. Tresp, J. Gu1, B. Wu, "Effective and efficient vote attack on capsule networks," *International Conference on Learning Representations (ICLR),* 2021.

[39] J. Deng, W. Dong, R. Socher, L. Li, K. Li and L. Fei-Fei, "ImageNet: a large-scale hierarchical image database," *IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009, pp.248-255.

[40] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Use of a capsule network to detect fake images and videos," October, 2019

[41] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-Forensics networks for deepfake detection," *Advances in Computer Vision and Pattern Recognition,* Springer, Cham, 2022.

[42] R. Tolosana, S. Romero-Tapiador, J. Fierrez, and R. Vera-Rodriguez, "DeepFakes evolution: analysis of facial regions and fake detection performance," 2020.

[43] D. E. King, "Dlib-ml: a machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755-1758, December 2009, doi/10.5555/1577069.1755843.

[44] L. Y. Gong, X. J. Li, and P. H. J. Chong, "Swin-Fake: A consistency learning transformer-based deepfake video detector," *Electronics*, 13, vol. 15, pp. 3045, 2024.

[45] A. H. Soudy, O. Sayed, H. Tag-Elser, "Deepfake detection using convolutional transformer and convolutional neural networks," *Journal of Neural Computing and Applications*, vol. 36, pp. 19759-19775, 2024.

Machine-Learning–Assisted Probabilistic Wind Assessment at Sechura, Peru

Ubaldo Yancachajlla Tito[1], Celso Antonio Sanga Quiroz[2], Edilberto Velarde Coaquira[3], Germán Belizario Quispe[4]
Escuela de Posgrado, Universidad Nacional de San Agustin de Arequipa, Arequipa, Peru[1]
Escuela Profesional de Ingeniería Mecánica, Universidad Nacional de San Agustin de Arequipa, Arequipa, Peru[2]
Escuela Profesional de Ingeniería Agrícola, Universidad Nacional del Altiplano, Puno, Peru[3, 4]

***Abstract*—Accurate characterization of wind resources is essential for reliable energy yield estimation and wind farm planning, particularly in regions with limited long-term measurements. This study presents a machine-learning–assisted probabilistic wind assessment in Sechura, Peru, based on multi-year hourly wind data obtained from the NASA POWER database. A representative Typical Meteorological Year (TMY) was constructed to preserve seasonal and diurnal variability while enabling standardized annual energy production (AEP) calculations. Wind speed distributions were modeled using empirical distributions, kernel density estimation (KDE), the Weibull distribution, and Gaussian mixture models (GMM). Statistical evaluation indicates that KDE and GMM reduce the annual RMSE by more than 50% compared to the Weibull model, achieving coefficients of determination above 0.98. Annual energy production is estimated at approximately 1.88 GWh, with differences below 0.3% among probabilistic models. The corresponding capacity factor is approximately 0.25 for a utility-scale wind turbine. The results demonstrate that advanced probabilistic models substantially improve wind speed representation while having a limited impact on integrated annual energy estimates, highlighting the importance of model selection for variability and seasonal analysis rather than for annual yield estimation.**

***Keywords*—Wind resource assessment; probabilistic modeling; machine learning; kernel density estimation; Gaussian mixture model; annual energy production; capacity factor**

I. Introduction

Reliable characterization of the wind resource is a fundamental prerequisite for accurate annual energy production (AEP) estimation, optimal site selection, and robust planning of wind-power projects. Traditionally, wind-speed variability has been modeled using parametric probability density functions, with the two-parameter Weibull distribution being the most commonly employed due to its mathematical simplicity and the availability of closed-form expressions for wind power density and related energy metrics. However, numerous studies have shown that the Weibull distribution can present significant shortcomings when representing wind regimes characterized by skewness, intermittency, strong seasonality, or multimodal behavior. Comparative studies at regional and global scales have demonstrated that no single parametric model can universally capture the complexity of wind-speed distributions across different climatic contexts, particularly in coastal areas and complex terrain [1], [2]. Such limitations may lead to biased estimates of wind power density and, consequently, of AEP [3]–[5].

In response to these limitations, more flexible statistical approaches have been developed and applied. Kernel density estimation (KDE) has emerged as a prominent nonparametric alternative, capable of reconstructing the probability density function directly from observed data without imposing a predefined functional form. Several studies report that KDE systematically outperforms Weibull models in goodness-of-fit metrics and distributional distances, especially when advanced bandwidth-selection strategies and bias-correction techniques are applied [6]–[8].

At the same time, mixture models offer an effective parametric compromise by representing complex distributions as weighted combinations of simpler component distributions. Weibull mixtures and Gaussian mixture models have shown superior capability to capture wind multimodality and to provide more realistic uncertainty estimates for energy production [9]–[15]. These approaches have been extensively used both for wind-resource assessment and for analyses of variability and energy-related risk.

In regions lacking long in-situ measurement records, atmospheric reanalysis products have become an established and reliable data source for wind-resource evaluation. In particular, the NASA MERRA-2 reanalysis has been extensively validated and demonstrated to provide surface and near-surface wind fields suitable for energy applications [6], [16]. Platforms such as NASA POWER facilitate access to these data and have been widely used in studies of wind potential and turbine selection [17]– [22].

This study combines a regional case study (Sechura, Peru) with a differentiated methodological approach. Sechura is a coastal area with high wind potential that has been little explored using advanced methods. Recent studies indicate that wind energy in Peru remains underdeveloped and its potential is not yet fully exploited [23]. In this context, our research provides original value by applying advanced probabilistic techniques, assisted by machine-learning approaches, to characterize the local wind resource. Specifically, we employ an unsupervised Gaussian mixture model (GMM) to identify wind regimes in Sechura, which, to the best of our knowledge, has not been widely documented in the local literature.

Within this context, the present work develops a machine-learning–assisted probabilistic assessment of the wind resource for Sechura, Piura (Peru), using hourly MERRA-2 reanalysis data. We systematically compare empirical distributions, kernel density estimation, the Weibull distribution, and mixture

models, evaluating their statistical performance and their impact on wind power density estimation, annual energy production, and capacity factor. The objective is to provide a rigorous methodological perspective to improve wind-resource characterization in arid coastal environments, where wind variability and multimodality are critical factors.

II. METHODOLOGY

A. Study Area and Data

The study area is located in Sechura, Province of Sechura, Piura, Peru, on the northern Pacific coast of South America (latitude −5.99°, longitude −81.08°). The region is characterized by arid climatic conditions, low surface roughness, and relatively flat terrain, making it suitable for wind resource assessment. The local wind regime is influenced by southeast trade winds and land–sea breeze circulations, which contribute to pronounced seasonal and diurnal variability.

Hourly wind speed data at 50 m above ground level were obtained from the NASA POWER database, based on the MERRA-2 reanalysis [16]. The dataset spans the period from December 2014 to January 2026 and includes wind speed, wind direction, and surface pressure. The average elevation of the corresponding grid cell is approximately 31.65 m above sea level.

Due to the absence of long-term in situ wind measurements in Sechura, MERRA-2 reanalysis data were used as the primary data source. Reanalysis products combine historical global observations with numerical atmospheric models through data assimilation techniques, providing temporally consistent, long-term wind datasets with global coverage. Such datasets are widely employed in preliminary wind resource assessments, particularly in regions lacking ground-based measurements.

Nevertheless, previous studies have reported that reanalysis data may present biases in coastal environments, including potential underestimation of wind speeds due to spatial resolution and surface parameterization limitations [24]. Therefore, the results presented here should be interpreted as a probabilistic assessment based on large-scale atmospheric reanalysis data. Future work should incorporate local meteorological mast measurements to validate and calibrate the probabilistic models developed in this study.

The NASA POWER wind-speed data used in this study correspond to a nominal measurement height of 50 m above ground level. For consistency, no vertical extrapolation or wind-shear correction was applied to adjust the wind speeds to the turbine hub height. The selected Gamesa G58/850 turbine typically operates at hub heights slightly above this level (approximately 55–65 m), and the study area is characterized by flat terrain and low surface roughness. Under these conditions, the expected wind-speed difference between 50 m and the hub height is limited. Nevertheless, the use of 50 m wind speeds directly in the energy calculations represents a simplifying assumption. A more detailed site-specific assessment could apply logarithmic or power-law vertical wind profiles to explicitly account for wind shear effects.

Wind-speed and surface-pressure data were obtained from the NASA POWER reanalysis and updated through January 2026. Hourly series were downloaded manually from the NASA POWER portal and then processed using a Python pipeline that performs chronological concatenation of hourly records and basic quality control: nonphysical values (e.g., negative wind speeds) are clipped or removed, and remaining gaps are handled during aggregation. Derived quantities (e.g., air density) are computed from the cleaned time series, and a Typical Meteorological Year (TMY) was constructed by averaging all available observations for each calendar hour. This workflow supports straightforward manual updates as new data becomes available and thus facilitates frequent refreshes of the resource assessment.

B. Typical Meteorological Year Construction

To enable standardized AEP calculations while preserving long-term variability, a Typical Meteorological Year (TMY) was constructed from the multi-year dataset. For each calendar hour defined by month, day, and hour, the representative wind speed was computed as the arithmetic mean of all available observations corresponding to that time slot across the full record. Leap-day observations (29 February) were excluded to maintain a total of 8760 hourly values.

This procedure reduces the influence of anomalous years while retaining essential seasonal and diurnal variability, and is a widely accepted practice in wind and solar resource assessments [17], [20].

The Typical Meteorological Year (TMY) was constructed by averaging the multi-year hourly wind-speed series for each hour of the calendar year, rather than applying the Finkelstein–Schafer (FS) method, which selects representative months based on statistical goodness-of-fit criteria [25]. The hourly averaging approach produces a continuous and temporally homogeneous climatological year, avoiding potential discontinuities that may arise when concatenating months from different years.

Although the FS technique is widely applied in solar energy studies, the present wind-focused analysis prioritizes the temporal smoothness and long-term mean representativeness of the multi-year hourly average. This averaged TMY is appropriate for the comparative probabilistic assessment conducted in this study. Nevertheless, future research could evaluate a TMY constructed using the FS methodology to quantify potential differences in estimated energy production.

C. Probabilistic Wind Speed Models

1) Empirical distribution: The empirical probability density function (PDF) was obtained directly from the TMY wind speed data using normalized histograms and serves as the reference distribution for model comparison.

2) Kernel Density Estimation (KDE): KDE was employed as a nonparametric method to estimate the wind-speed PDF. Several bandwidth-selection strategies were considered, including normal-scale rules and cross-validation-based methods, which have demonstrated superior performance in wind-energy applications [6], [7], [8], [27].

The KDE of wind speed v is defined as

$$f_{KDE}(v) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{v - v_i}{h}\right) \quad (1)$$

where, n is the sample size, h is the bandwidth, and $K(\cdot)$ is a Gaussian kernel. Two bandwidth selection methods were considered: a normal-scale rule (KDE-NS) and least-squares cross-validation (KDE-LSCV), the latter being shown to provide superior performance in wind energy applications [4].

3) Weibull distribution: The two-parameter Weibull distribution was used as the parametric reference model. Shape and scale parameters were estimated via maximum likelihood following standard practice in the literature [1], [2], [5], [28].

The Weibull PDF is given by

$$f_W(v) = \frac{k}{c}\left(\frac{v}{c}\right)^{k-1} \exp\left[-\left(\frac{v}{c}\right)^{k}\right] \quad (2)$$

where, k is the shape parameter and c is the scale parameter. Maximum likelihood estimation was used to estimate the parameters. Despite its widespread use, the Weibull distribution may inadequately represent multimodal or highly variable wind regimes [1], [2].

4) Gaussian Mixture Model (GMM): To capture possible multimodal behaviour, mixture models were applied, including Weibull mixtures and Gaussian mixture models (GMMs). These models represent the overall PDF as a weighted sum of individual component distributions and have demonstrated superior capability to describe complex wind regimes and to quantify energetic uncertainty [9]–[15].

The GMM represents the wind speed PDF as a weighted sum of Gaussian components:

$$f_{GMM}(v) = \sum_{j=1}^{M} \pi_j \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left[-\frac{(v-\mu_j)^2}{2\sigma_j^2}\right] \quad (3)$$

where, M is the number of components, π_j are the mixing weights, and μ_j, σ_j are the mean and standard deviation of each component. A three-component GMM was selected based on model parsimony and interpretability, consistent with previous wind energy studies using mixture models [9].

Although the applied techniques correspond to classical statistical learning rather than deep learning, both kernel density estimation and Gaussian mixture models are formally categorized as unsupervised machine-learning methods.

The use of the term 'Machine-Learning–Assisted' is justified because the analysis incorporates an unsupervised machine-learning algorithm within the probabilistic modeling framework. Specifically, we employ a Gaussian Mixture Model (GMM) [26], which assumes that the wind-speed distribution arises from a weighted combination of multiple Gaussian components. Model parameters are estimated through the expectation–maximization (EM) algorithm, enabling the automatic identification of latent wind regimes without prior labeling. This unsupervised clustering approach captures multimodal characteristics of the wind-speed distribution that cannot be adequately represented by single-parameter models such as the Weibull distribution. Therefore, although the framework does not rely on supervised predictive learning, the integration of automated data-driven clustering techniques supports the use of the term 'machine-learning–assisted' in this study.

The optimal number of mixture components was determined using statistical information criteria. Models with varying numbers of clusters were fitted, and both the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) were computed for each configuration. The criteria exhibited a minimum (or clear elbow point) around k=3, indicating that three components provide an adequate balance between model goodness-of-fit and complexity. In addition, the selection of three clusters is physically consistent with the local wind-resource characteristics, where it is plausible to distinguish low-, medium-, and high-wind regimes. This approach avoids overfitting while ensuring a parsimonious yet flexible probabilistic representation of the wind-speed distribution.

To avoid heuristic selection of the number of mixture components, models with different values of k were evaluated using the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). Both criteria reached their minimum at k=3, indicating that three components provide the optimal trade-off between model fit and complexity. The large ΔAIC relative to the Weibull model further supports this selection.

D. Statistical Performance Metrics

Model performance was assessed using a combination of goodness-of-fit metrics and distributional distance measures commonly adopted in wind-speed probabilistic modeling. These include the root mean square error (RMSE), mean absolute error (MAE), the Kolmogorov–Smirnov (KS) statistic, the Kullback–Leibler (KL) divergence, the Hellinger distance, and the one-dimensional Wasserstein distance, which together provide a comprehensive evaluation of both pointwise errors and global distributional discrepancies [3], [4].

The R^2, RMSE and MAE were computed as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(f_{\text{emp},i} - f_{\text{model},i}\right)^2}{\sum_{i=1}^{n}\left(f_{\text{emp},i} - \overline{f_{\text{emp}}}\right)^2} \quad (4)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(f_{emp,i} - f_{model,i}\right)^2} \quad (5)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|f_{emp,i} - f_{model,i}\right| \quad (6)$$

The Kolmogorov–Smirnov statistic was used to quantify the maximum absolute deviation between the empirical and modeled cumulative distribution functions (CDFs):

$$KS = \sup_{v} \left| F_{emp}(v) - F_{model}(v) \right| \tag{7}$$

Kullback–Leibler divergence

$$D_{KL}(f_{emp} \| f_{model}) = \sum_{i} f_{emp,i} \ln \left(\frac{f_{emp,i}}{f_{model,i}} \right) \tag{8}$$

Hellinger distance

$$H = \frac{1}{\sqrt{2}} \sqrt{\sum_{i} \left(\sqrt{f_{emp,i}} - \sqrt{f_{model,i}} \right)^2} \tag{9}$$

Wasserstein distance (1D, discrete form)

$$W = \sum_{i} \left| F_{emp}(v_i) - F_{model}(v_i) \right| \Delta v \tag{10}$$

E. Wind Power Density Estimation

Wind power density (WPD) was computed from each modeled PDF, incorporating air density estimated from surface pressure data. Annual energy production and capacity factor were calculated by mapping wind speeds to the power curve of a representative commercial wind turbine and integrating the resulting hourly power time series according to standard procedures [17]– [22].

Wind power density (WPD) was calculated from each modeled PDF as:

$$WPD = \frac{1}{2} \rho \int_{0}^{\infty} v^3 f(v) dv \tag{11}$$

where, ρ is air density, computed from surface pressure data under standard atmospheric assumptions.

F. Energy Production and Capacity Factor

Hourly electrical power output was obtained by mapping wind speeds to a commercial-scale wind turbine power curve. The annual energy production (AEP) was computed as:

$$E_{annual} = \sum_{i=1}^{8760} P_i \Delta t, \qquad \Delta t = 1\,\text{h} \tag{12}$$

The mean annual power was calculated as:

$$\bar{P} = \frac{E_{annual}}{8760} \tag{13}$$

The capacity factor (CF) was defined as:

$$CF = \frac{\bar{P}}{P_{rated}} \tag{14}$$

Seasonal energy production and capacity factors were computed analogously by integrating over the corresponding seasonal subsets of the TMY dataset.

Reference wind turbine. A commercial horizontal-axis wind turbine, Gamesa G58/850, was selected as the reference machine for energy production and capacity factor calculations, with technical specifications obtained from TheWindPower.net [29]. The turbine has a rated power of 850 kW and a three-bladed rotor with a diameter of 58 m (swept area = 2,642 m²). Cut-in, rated, and cut-out wind speeds are 3.0 m/s, 12.5 m/s, and 25.0 m/s, respectively. Maximum rotor speed is 30.8 rpm (tip speed ≈ 94 m/s). Rotor blades are fiberglass/epoxy composites. This turbine model was used to map probabilistic wind speed distributions to electrical output via the manufacturer's power curve for AEP and CF estimation [29].

G. Computational Implementation and Cost

To assess practical computational costs, bandwidth selection via leave-one-out least-squares cross-validation (LSCV) was compared against a simple rule-of-thumb (Silverman's rule). Using our Python implementation (vectorized operations and peak memory tracking via tracemalloc), the following empirical results were obtained: the LSCV bandwidth search required ≈ 1.06 s with a peak Python allocation of ≈ 32.03 MB, whereas KDE evaluation on the fine integration grid (using either the Silverman bandwidth or the LSCV-selected bandwidth) completed in ≈ 0.33 – 0.35 s with negligible allocation peaks (≈ 0.11 MB). All benchmarks were executed on the author's workstation: Intel Corporation Core(TM) i5-7200U @ 2.50 GHz (2 physical cores, 4 logical processors) with 17,060,569,088 bytes (≈ 15.89 GB) RAM. These results indicate that the dominant computational overhead arises from the repeated evaluations inherent to LSCV bandwidth selection rather than from KDE evaluation once the bandwidth is fixed. For practical applications, we therefore recommend: 1) subsampling or parallelization when applying LSCV to large datasets; 2) using rule-of-thumb bandwidths (Silverman/Scott) or approximate KDE implementations (KD-tree, FFT/binning) for very large n or latency-constrained settings; and 3) complementing tracemalloc with system-level monitoring tools (e.g., psutil) when total process memory usage is required. Table I summarizes the measured execution times and peak Python allocation on the reference workstation.

TABLE I. BENCHMARK RESULTS (EXECUTION TIME AND PEAK PYTHON ALLOCATION

Metric	Time (s)	Peak allocation (bytes)	Peak (MB)
LSCV bandwidth search	1.0621	32033025	30.54
KDE evaluation (Silverman)	0.3499	119464	0.11
KDE evaluation (LSCV bandwidth)	0.3341	119424	0.11

Note: Reference workstation: Intel Corporation Core(TM) i5-7200U CPU @ 2.50 GHz — 2 physical cores (4 logical processors); 17,060,569,088 bytes (≈ 15.89 GB) RAM. Measurements performed on the author's machine; memory values report Python allocation peaks captured by tracemalloc.

Execution times and memory usage depend on hardware and system configuration (e.g., multithreaded BLAS).

III. Results and Discussion

The performance of probabilistic wind speed models was evaluated using both statistical and distributional metrics on the Typical Meteorological Year (TMY) dataset for Sechura, Peru. The models analyzed include empirical histograms, kernel density estimation with normal and least-squares cross-validation bandwidths (KDE-NS and KDE-LSCV), a two-parameter Weibull distribution, and a three-component Gaussian mixture model (GMM(3)). The main objective of this section is to compare these models in terms of fit accuracy and to interpret their implications for wind resource assessment.

A. Annual PDF Fitting Accuracy

Fig. 1 compares the annual empirical wind speed distribution with the fitted PDFs and CDFs obtained using KDE, Weibull, and GMM models. The KDE-LSCV closely follows the empirical distribution across the entire wind speed range, while the Weibull model exhibits noticeable deviations, particularly in the tails, explaining its poorer goodness-of-fit metrics reported in Table II.

Table II summarizes the annual goodness-of-fit metrics for each model. The empirical distribution serves as a perfect reference, as expected, with zero error across all metrics.

Among the fitted models, KDE-LSCV achieves the best overall performance, with the highest coefficient of determination (R^2 = 0.98) and the lowest values in the Kolmogorov–Smirnov statistic (KS = 0.02), RMSE, MAE, and distributional distance measures (Hellinger, Kullback–Leibler, and Wasserstein). This confirms that KDE-LSCV accurately represents both the central tendency and the tails of the wind speed distribution, which is critical for wind energy applications where tail behavior influences extreme events and energy yield predictions [6]– [8], [23].

Similar conclusions regarding the superiority of kernel-based approaches have been reported in recent studies employing advanced and adaptive bandwidth KDE techniques. For example, Chau et al. [30] demonstrated that hybrid adaptive KDE formulations significantly improve wind speed representation across diverse climatic conditions, while Ning et al. [31] showed that KDE-based probabilistic modeling provides robust performance even in highly variable and extreme wind regimes. Although the present study employs classical LSCV bandwidth selection, the results are consistent with these findings and confirm the strong suitability of KDE for wind resource assessment.

In contrast, the Weibull distribution exhibits substantially poorer performance, with R^2 = 0.62 and the largest error and distance metrics (e.g., Wasserstein = 0.37). This indicates a significant mismatch with the empirical distribution, especially in the tails. These findings align with previous comparisons showing that fixed parametric models such as Weibull can be inadequate in complex wind regimes [1]– [5].

The GMM(3) model improves significantly upon the Weibull distribution and yields results comparable to KDE-NS, demonstrating that mixture distributions are effective in capturing multimodal wind speed structures. Previous studies likewise highlight the advantages of mixture models for improved uncertainty representation in wind energy modeling [9]– [15]. Nevertheless, GMM(3) remains slightly inferior to KDE-LSCV across most evaluation metrics, reflecting the superior flexibility of non-parametric density estimation methods.

TABLE II. Annual PDF Fit Metrics

Model	R^2	KS	RMSE	MAE	Hellinger	KL	Wasserstein
Empirical	1.00	0.00	0.00	0.00	0.00	0.00	0.00
KDE-NS	0.95	0.03	56.56	31.43	0.06	0.01	0.05
KDE-LSCV	0.98	0.02	39.35	23.07	0.04	0.01	0.02
Weibull	0.62	0.11	154.00	95.00	0.18	0.12	0.37
GMM (3)	0.94	0.03	59.70	36.49	0.07	0.02	0.05

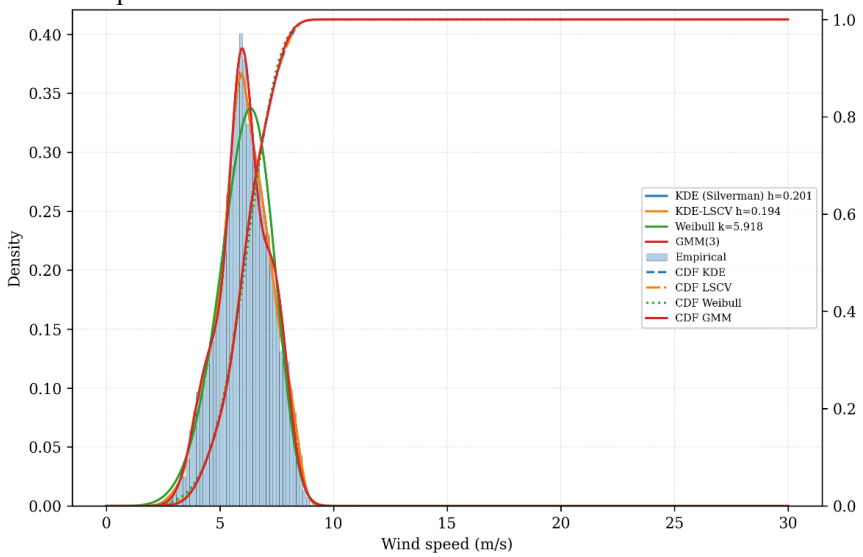


Fig. 1. Comparison of annual wind speed probability density functions (PDFs) and cumulative distribution functions (CDFs) for Sechura using empirical data, KDE, Weibull, and GMM models.

B. Seasonal Probabilistic Fitting Performance

Fig. 2 presents the seasonal empirical wind speed distributions together with the fitted probability density functions (PDFs) and cumulative distribution functions (CDFs) for Summer, Autumn, Winter, and Spring. Across all seasons, the KDE-LSCV model closely follows the empirical distribution, accurately capturing both central tendencies and tail behavior. In contrast, the Weibull distribution exhibits noticeable deviations, particularly during Winter and Summer, where multimodal features are evident in the empirical data. The GMM(3) model provides an improved representation of multimodality compared to Weibull, although it remains slightly less accurate than KDE-LSCV in terms of overall distributional agreement.

Seasonal performance for Summer, Autumn, Winter and Spring is shown in Table II. KDE-LSCV consistently achieves the highest R^2 and the lowest KS, RMSE, MAE, and distance metrics across all seasons, indicating robustness in capturing seasonal wind speed variations.

The GMM(3) model exhibits intermediate performance, consistently outperforming the Weibull distribution across most evaluation metrics. In particular, the Weibull model performs poorly during extreme seasonal regimes (e.g., summer and winter), as evidenced by elevated Kolmogorov–Smirnov statistics and large error values. This behavior indicates a limited ability to adapt to seasonally varying distribution shapes and tail characteristics. These observations are consistent with findings in the literature, which report that simple parametric assumptions frequently fail in environments characterized by complex and highly variable climatic patterns [3]– [5].

Table III lists the seasonal PDF fit metrics for the selected wind-speed models (KDE-LSCV, GMM(3) and Weibull). The table reports R^2, Kolmogorov–Smirnov (KS) statistic, RMSE, MAE and three distributional distances (Hellinger, KL and Wasserstein) for Summer, Autumn, Winter and Spring.

As shown in Table III, KDE-LSCV consistently delivers the best seasonal goodness-of-fit, with $R^2 \approx 0.97$–0.98 and the lowest KS, RMSE and distributional distances across all seasons. GMM(3) produces competitive, intermediate results (typically $R^2 \approx 0.95$), while the Weibull model performs poorly ($R^2 \approx 0.60$–0.65) and exhibits the largest errors and distributional discrepancies, especially in Winter and Summer. These seasonal patterns confirm the robustness of non-parametric and mixture approaches for capturing changing wind regimes (Table III).

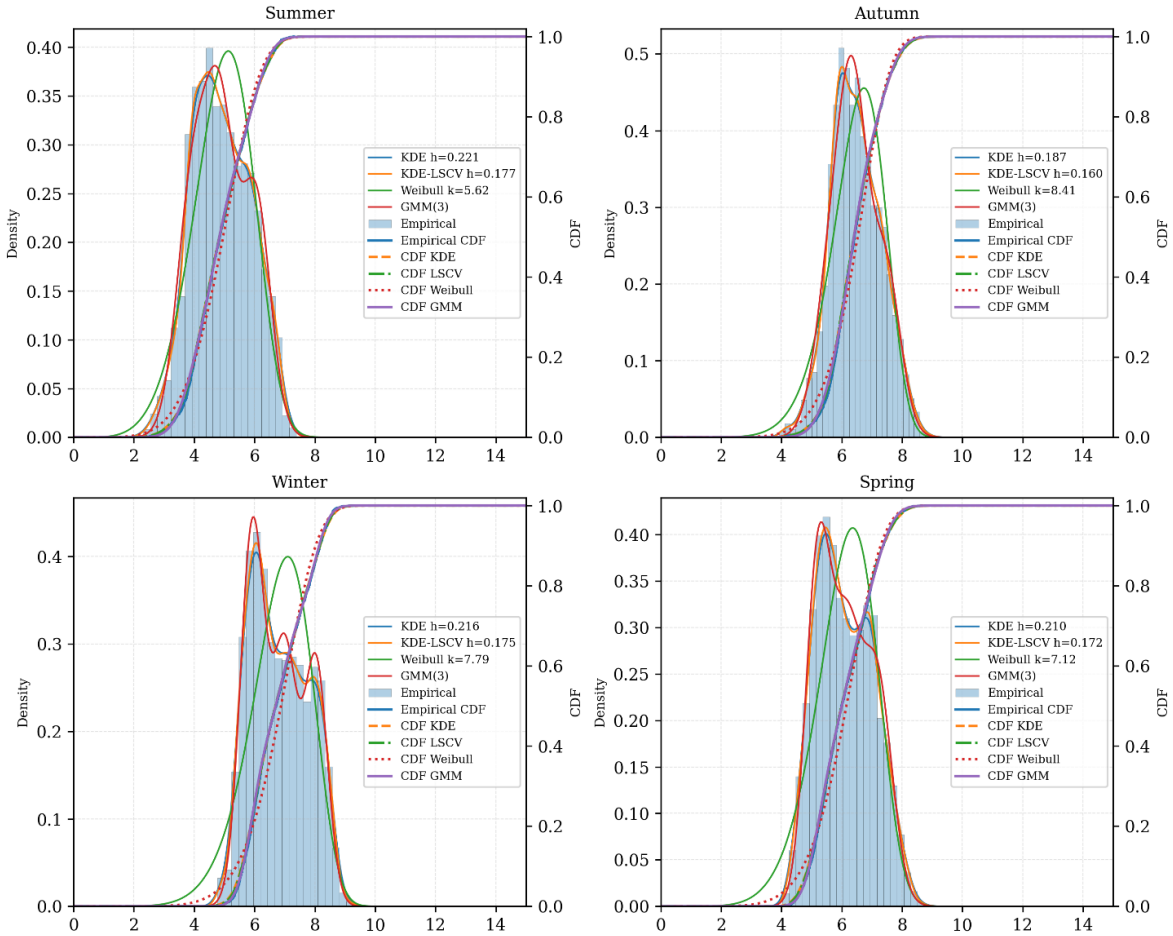


Fig. 2. Seasonal empirical wind speed distributions and fitted PDFs and CDFs for Summer, Autumn, Winter, and Spring using KDE, Weibull, and GMM models.

TABLE III. SEASONAL PDF FIT METRICS FOR SELECTED WIND SPEED MODELS

Season	Model	R^2	KS	RMSE	MAE	Hellinger	KL	Wasserstein
Summer	KDE-LSCV	0.97	0.02	45.12	25.99	0.04	0.01	0.02
	GMM (3)	0.95	0.03	52.33	30.44	0.05	0.01	0.05
	Weibull	0.61	0.11	160.23	98.11	0.19	0.13	0.39
Autumn	KDE-LSCV	0.98	0.02	35.99	21.46	0.04	0.00	0.02
	GMM (3)	0.96	0.03	49.12	29.25	0.05	0.01	0.04
	Weibull	0.65	0.10	140.99	85.12	0.17	0.11	0.36
Winter	KDE-LSCV	0.97	0.02	40.46	23.35	0.04	0.01	0.02
	GMM (3)	0.95	0.03	55.68	34.12	0.06	0.01	0.06
	Weibull	0.60	0.12	170.46	105.68	0.19	0.14	0.41
Spring	KDE-LSCV	0.97	0.02	42.33	24.57	0.04	0.01	0.02
	GMM (3)	0.95	0.03	57.00	36.79	0.06	0.01	0.06
	Weibull	0.61	0.12	165.12	100.23	0.19	0.13	0.40

C. Impact on Energy Production and Capacity Factors

Table IV presents the seasonal mean power output, annual energy production (AEP), and capacity factor estimated using the different probabilistic models. Although substantial differences are observed in the quality of the wind speed distribution fitting, the resulting integrated energy metrics remain remarkably consistent across models. Specifically, deviations in AEP and capacity factor are below 0.3%, indicating that long-term energy estimates are relatively insensitive to moderate discrepancies in probability density representation.

This behavior can be attributed to the smoothing effect of turbine power curves and the dominance of mid-range wind speeds in the energy contribution, which reduces sensitivity to distributional differences in the tails. Similar findings have been reported in previous wind energy studies, where parametric and non-parametric models yield comparable long-term energy outputs despite notable differences in statistical performance metrics [17]– [22].

Nevertheless, accurate probabilistic modeling remains critical when the scope extends beyond aggregated annual indicators. Applications such as uncertainty quantification, risk assessment, seasonal forecasting, and turbine selection or optimization require faithful representation of the full wind speed distribution, particularly in the tails, where extreme events and variability play a decisive role.

TABLE IV. SEASONAL ENERGY PRODUCTION AND CAPACITY FACTOR

Season	Model	Mean power (kW)	AEP (MWh)	CF
Summer	Empirical	115.87	250.28	0.14
	KDE-NS	116.39	251.40	0.14
	KDE-LSCV	116.20	251.00	0.14
	Weibull	116.62	251.91	0.14
	GMM (3)	115.87	250.28	0.14
Autumn	Empirical	246.28	537.88	0.29
	KDE-NS	246.60	538.58	0.29
	KDE-LSCV	246.52	538.39	0.29
	Weibull	246.27	537.86	0.29
	GMM (3)	246.29	537.91	0.29
Winter	Empirical	282.67	624.13	0.33
	KDE-NS	283.06	625.00	0.33
	KDE-LSCV	282.93	624.70	0.33
	Weibull	283.65	626.31	0.33
	GMM (3)	282.67	624.13	0.33
Spring	Empirical	211.47	466.92	0.25
	KDE-NS	211.88	467.84	0.25
	KDE-LSCV	211.75	467.54	0.25
	Weibull	212.11	468.35	0.25
	GMM (3)	211.49	466.96	0.25

Note: AEP and CF were computed using the power curve of the Gamesa G58/850 wind turbine (rated power: 850 kW; rotor diameter: 58 m; cut-in / rated / cut-out wind speeds: 3.0 / 12.5 / 25.0 $m \cdot s^{-1}$). Technical specifications were obtained from TheWindPower.net [29].

Fig. 3 compares the seasonal energy production estimated using the different probabilistic wind speed models. Despite the substantial differences observed in the statistical fitting accuracy of the distributions, the resulting seasonal energy estimates are remarkably consistent across all models. For all seasons, deviations in AEP remain very small, confirming that integrated energy metrics are weakly sensitive to moderate discrepancies in wind speed probability density modeling. This behavior is mainly attributed to the smoothing effect of turbine power curves and the dominance of mid-range wind speeds in energy production.

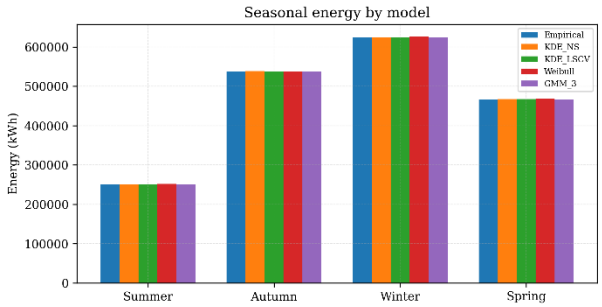


Fig. 3. Seasonal energy production estimated using different probabilistic wind speed models.

TABLE V. ANNUAL GENERATED ENERGY AND CAPACITY FACTOR FOR DIFFERENT PROBABILISTIC WIND MODELS

Model	Generated Energy (Wh)	Capacity Factor
Empirical	1,879,199,060	0.25
KDE-NS	1,882,521,226	0.25
KDE-LSCV	1,882,318,322	0.25
Weibull	1,883,887,128	0.25
GMM(3)	1,879,320,489	0.25

As shown in Table V, the estimated annual energy production varies by less than 0.3% among the considered probabilistic models, resulting in capacity factors close to 0.25 for all cases. Although advanced probabilistic models significantly improve the representation of wind speed distributions, their impact on integrated annual energy production is limited, as evidenced by the results in Table V. This confirms that annual energy yield is primarily driven by the mean wind regime, whereas model selection becomes more relevant for variability, extremes, and seasonal analysis.

From a methodological perspective, the results highlight the effectiveness of data-driven and machine-learning-based probabilistic models for wind resource characterization. Although the applied techniques belong to classical statistical learning rather than deep learning, both kernel density estimation with data-adaptive bandwidth selection and Gaussian mixture models can be formally classified as unsupervised machine learning approaches. Their superior performance relative to classical parametric models confirms the value of machine learning tools for capturing complex wind speed distributions.

These statistical indicators quantify the degree of divergence between wind-speed probability distributions, which directly translates into differences in the estimated annual energy production (AEP). From an energy perspective, low values of KS, KL divergence, or Hellinger distance (approaching zero) indicate highly similar distributions, implying that AEP estimates derived from alternative models would differ only marginally. Conversely, higher metric values reveal substantial discrepancies in the representation of wind regimes, particularly in the tails of the distribution, which disproportionately influence power output due to the cubic dependence of wind power on velocity.

In particular, divergence-based measures such as KL and Hellinger are sensitive to distributional shape differences, including multimodality and tail behavior, which can lead to meaningful variations in AEP when inadequately modeled. The Wasserstein distance provides an intuitive interpretation, as it measures the average displacement required to transform one distribution into another in the wind-speed domain; larger values therefore indicate greater shifts in effective wind regimes and potentially greater deviations in energy yield. Overall, these metrics provide a quantitative bridge between statistical goodness-of-fit and the reliability of energy production estimates.

As a potential extension of this work, the probabilistic framework developed here could be used to estimate P50 and P90 energy production levels. This would involve generating multiple synthetic wind-speed realizations from the fitted probabilistic model (e.g., via Monte Carlo sampling) and computing the corresponding annual energy production (AEP) for each realization. The resulting AEP distribution would allow direct extraction of production percentiles such as P50 and P90. Alternatively, analytical integration of the estimated joint probability density could be explored to derive production quantiles. Although beyond the scope of the present study, this extension would enable explicit quantification of production uncertainty and improve risk assessment in wind-energy project evaluation.

IV. CONCLUSIONS

This study presented a comprehensive probabilistic assessment of the wind resource at Sechura, Peru, based on a Typical Meteorological Year derived from long-term reanalysis data. Empirical distributions, kernel density estimation, Weibull fitting, and Gaussian mixture models were systematically compared using classical goodness-of-fit metrics, distributional distance measures, and energy-based indicators at both annual and seasonal scales.

The results demonstrate that non-parametric kernel density estimation with least-squares cross-validation (KDE–LSCV) provides the most accurate representation of the wind speed distribution. At the annual scale, KDE–LSCV consistently achieved the highest coefficient of determination and the lowest Kolmogorov–Smirnov statistic, RMSE, and distributional distances (Hellinger, Kullback–Leibler, and Wasserstein). These findings confirm its superior ability to reproduce both the central tendency and the tails of the empirical distribution.

Seasonal analyses further reinforced these conclusions. KDE–LSCV outperformed all alternative models in Summer, Autumn, Winter, and Spring, exhibiting stable performance across contrasting wind regimes. In particular, the Winter season revealed pronounced limitations of unimodal parametric models, where the Weibull distribution failed to capture the observed complexity of the wind regime, resulting in very poor goodness-of-fit metrics. In contrast, Gaussian mixture models demonstrated strong seasonal robustness, especially during Winter, indicating their suitability for representing multimodal wind patterns.

Despite the substantial differences observed in statistical goodness-of-fit metrics, annual and seasonal energy production estimates remained remarkably consistent across models, with deviations generally below 0.3 %. Electrical efficiency and capacity factor values were virtually unchanged, indicating that energy-based indicators are less sensitive to moderate distributional discrepancies under the conditions examined. Nevertheless, accurate probabilistic modeling remains essential for applications involving risk analysis, extreme wind assessment, and turbine selection, where misrepresentation of distribution tails can have significant consequences.

Overall, the results show that reliance on a single Weibull distribution may lead to systematic mischaracterization of wind regimes in coastal sites with complex seasonal variability. KDE–LSCV is recommended as the primary modeling approach for wind resource characterization at Sechura, while Gaussian mixture models provide a valuable parametric alternative when

compact analytical representations are required. The combined use of distributional distance metrics and energy-based indicators is strongly recommended for robust wind energy assessments.

Future work should focus on validating these findings using in situ measurements, extending the analysis to hub-height extrapolation and uncertainty quantification, and evaluating the implications of probabilistic model choice on long-term economic performance and grid-integration studies.

REFERENCES

[1] P. Wais, "A review of Weibull functions in wind sector," Renewable and Sustainable Energy Reviews, vol. 70, pp. 1099–1107, Apr. 2017, doi: 10.1016/J.RSER.2016.12.014.

[2] J. A. Carta, P. Ramírez, and S. Velázquez, "A review of wind speed probability distributions used in wind energy analysis: Case studies in the Canary Islands," Renewable and Sustainable Energy Reviews, vol. 13, no. 5, pp. 933–955, 2009, doi: https://doi.org/10.1016/j.rser.2008.05.005.

[3] C. Jung and D. Schindler, "Global comparison of the goodness-of-fit of wind speed distributions," Energy Convers. Manag., vol. 133, pp. 216–234, 2017, doi: https://doi.org/10.1016/j.enconman.2016.12.006.

[4] N. Masseran, "Evaluating wind power density models and their statistical properties," Energy, vol. 84, pp. 533–541, 2015, doi: https://doi.org/10.1016/j.energy.2015.03.018.

[5] T. P. Chang, "Estimation of wind energy potential using different probability density functions," Appl. Energy, vol. 88, no. 5, pp. 1848–1856, 2011, doi: https://doi.org/10.1016/j.apenergy.2010.11.010.

[6] Q. Han, S. Ma, T. Wang, and F. Chu, "Kernel density estimation model for wind speed probability distribution with applicability to wind energy assessment in China," Renewable and Sustainable Energy Reviews, vol. 115, p. 109387, 2019, doi: https://doi.org/10.1016/j.rser.2019.109387.

[7] M. Wahbah, B. Mohandes, T. H. M. EL-Fouly, and M. S. El Moursi, "Unbiased cross-validation kernel density estimation for wind and PV probabilistic modelling," Energy Convers. Manag., vol. 266, p. 115811, 2022, doi: https://doi.org/10.1016/j.enconman.2022.115811.

[8] X. Xu, Z. Yan, and S. Xu, "Estimating wind speed probability distribution by diffusion-based kernel density method," Electric Power Systems Research, vol. 121, pp. 28–37, 2015, doi: https://doi.org/10.1016/j.epsr.2014.11.029.

[9] S. A. Akdağ, H. S. Bagiorgas, and G. Mihalakakou, "Use of two-component Weibull mixtures in the analysis of wind speed in the Eastern Mediterranean," Appl. Energy, vol. 87, no. 8, pp. 2566–2573, 2010, doi: https://doi.org/10.1016/j.apenergy.2010.02.033.

[10] J. A. Carta and P. Ramírez, "Analysis of two-component mixture Weibull statistics for estimation of wind speed distributions," Renew. Energy, vol. 32, no. 3, pp. 518–531, 2007, doi: https://doi.org/10.1016/j.renene.2006.05.005.

[11] X. Qin, J. Zhang, and X. Yan, "Two Improved Mixture Weibull Models for the Analysis of Wind Speed Data," J. Appl. Meteorol. Climatol., vol. 51, no. 7, pp. 1321–1332, 2012, doi: https://doi.org/10.1175/JAMC-D-11-0231.1.

[12] Q. Hu, Y. Wang, Z. Xie, P. Zhu, and D. Yu, "On estimating uncertainty of wind energy with mixture of distributions," Energy, vol. 112, pp. 935–962, 2016, doi: https://doi.org/10.1016/j.energy.2016.06.112.

[13] F. S. dos Santos, K. K. F. do Nascimento, J. da Silva Jale, S. F. A. Xavier, and T. A. E. Ferreira, "Brazilian wind energy generation potential using mixtures of Weibull distributions," Renewable and Sustainable Energy Reviews, vol. 189, p. 113990, 2024, doi: https://doi.org/10.1016/j.rser.2023.113990.

[14] M. Franco and J.-M. Vivo, "Constraints for generalized mixtures of Weibull distributions with a common shape parameter," Stat. Probab. Lett., vol. 79, no. 15, pp. 1724–1730, 2009, doi: https://doi.org/10.1016/j.spl.2009.05.005.

[15] W. Wang, Y. Gao, and N. Ikegaya, "Approximating wind speed probability distributions around a building by mixture weibull distribution with the methods of moments and L-moments," Journal of Wind Engineering and Industrial Aerodynamics, vol. 257, p. 106001, 2025, doi: https://doi.org/10.1016/j.jweia.2024.106001.

[16] D. Carvalho, "An assessment of NASA's GMAO MERRA-2 reanalysis surface winds," J. Clim., vol. 32, no. 23, pp. 8261–8281, 2019, doi: 10.1175/JCLI-D-19-0199.1.

[17] Z. Wang and W. Liu, "Wind energy potential assessment based on wind speed, its direction and power data," Sci. Rep., vol. 11, no. 1, p. 16879, 2021, doi: 10.1038/s41598-021-96376-7.

[18] B. Belabes, A. Youcefi, O. Guerri, M. Djamai, and A. Kaabeche, "Evaluation of wind energy potential and estimation of cost using wind energy turbines for electricity generation in north of Algeria," Renewable and Sustainable Energy Reviews, vol. 51, pp. 1245–1255, 2015, doi: https://doi.org/10.1016/j.rser.2015.07.043.

[19] M. S. Adaramola, S. S. Paul, and S. O. Oyedepo, "Assessment of electricity generation and energy cost of wind energy conversion systems in north-central Nigeria," Energy Convers. Manag., vol. 52, no. 12, pp. 3363–3368, 2011, doi: https://doi.org/10.1016/j.enconman.2011.07.007.

[20] P. Spiru and P. L. Simona, "Wind energy resource assessment and wind turbine selection analysis for sustainable energy production," Sci. Rep., vol. 14, no. 1, p. 10708, 2024, doi: 10.1038/s41598-024-61350-6.

[21] M. M. Ahmed, Md. K. Islam, and M. H. Masud, "Evaluating wind energy feasibility in different locations of Bangladesh: A techno-economic and environmental perspective," Energy Conversion and Management: X, vol. 28, p. 101352, 2025, doi: https://doi.org/10.1016/j.ecmx.2025.101352.

[22] F. H. Ouerghi, M. Omri, A. A. Menaem, A. I. Taloba, and R. M. Abd El-Aziz, "Feasibility evaluation of wind energy as a sustainable energy resource," Alexandria Engineering Journal, vol. 106, pp. 227–239, 2024, doi: https://doi.org/10.1016/j.aej.2024.06.055.

[23] A. Gordillo Valdez, M. Montoya Granda, and P. A. Salinas Pedemonte, "Análisis del desarrollo y potencial de la energía eólica en el Perú," Ingeniería Industrial, vol. 43, pp. 177–198, Oct. 2022, doi: 10.26439/ing.ind2022.n43.6114.

[24] Gualtieri, G. (2022). Analysing the uncertainties of reanalysis data used for wind resource assessment: A critical review. Renewable and Sustainable Energy Reviews, 167, 112741. https://doi.org/https://doi.org/10.1016/j.rser.2022.112741

[25] J. M. Finkelstein and R. E. Schafer,"Improved goodness-of-fit tests," Biometrika, vol. 58, no. 3, pp. 641–645, 1971. [Online]. Available: https://academic.oup.com/biomet/article-abstract/58/3/641/233677

[26] C. M. Bishop, Pattern Recognition and Machine Learning. New York, NY, USA: Springer, 2006. doi: 10.1007/978-0-387-45528-0.

[27] M. Wahbah, S. F. Feng, T. H. M. EL-Fouly, and B. Zahawi, "Wind speed probability density estimation using root-transformed local linear regression," Energy Convers. Manag., vol. 199, p. 111889, 2019, doi: https://doi.org/10.1016/j.enconman.2019.111889.

[28] J. Liu, G. Xiong, X. Fu, and A. W. Mohamed, "Estimating the best-fit parameters of Weibull distribution with numerical methods for wind energy assessment: A case study in China," Energy Strategy Reviews, vol. 63, p. 102017, 2026, doi: https://doi.org/10.1016/j.esr.2025.102017.

[29] The Wind Power, "Gamesa G58–850 Wind Turbine," TheWindPower.net. [Online]. Available: https://www.thewindpower.net/turbine_en_43_gamesa_g58-850.php. Accessed: Jan. 17, 2026.

[30] T. T. Chau, A. B. Alhassan, M. Shaltayev, K. Talapiden, M. A. Shehu, and T. D. Do, "Assessment of wind energy potential using hybrid adaptive bandwidth kernel density technique: A case study of major cities in Kazakhstan," Energy Conversion and Management: X, vol. 29, p. 101439, 2026, doi: https://doi.org/10.1016/j.ecmx.2025.101439.

[31] J. Ning, R. Shi, S. Xuan, C. Jiang, and L. Jia, "Assessment of offshore island wind energy potential in typhoon-prone regions with a KDE-based probabilistic modeling approach," Energy, vol. 339, p. 139092, 2025, doi: https://doi.org/10.1016/j.energy.2025.139092.

Ziraai SSI: A Blockchain-Based Self-Sovereign Identity Model for Agricultural Supply Chains

Tariq Alar[1]*, Gnana Bharathy[2], Usha Batra[3], Mukesh Prasad[4]
Department of Computer Sciences-College of Engineering and Computer Sciences, Jazan University, Jazan, Saudi Arabia[1]
School of Computer Science, University of Technology Sydney, Sydney, NSW, Australia[1, 2, 4]
University of Technology Sydney, Sydney, NSW, Australia[3]
Skill Department of CS & IT-Faculty of Engineering & Technology, Shri Vishwakarma Skill University, Palwal, India[3]

***Abstract*—The agricultural supply chain plays a critical role in ensuring food security and sustainability; however, it continues to face challenges related to data fragmentation, limited transparency, and insufficient trust among participating stakeholders. Existing supply chain systems are primarily based on centralized identity and data management models, which introduce single points of failure, restrict auditability, and raise privacy concerns. More critically, the absence of decentralized and stakeholder-controlled identity mechanisms limits accountability and verifiable governance across agricultural ecosystems. In this work, we present Ziraai SSI, a blockchain-based self-sovereign identity (SSI) prototype designed to support identity-centric governance and trust establishment in agricultural supply chains. The proposed approach integrates blockchain-based trust anchoring with self-sovereign identity principles using decentralized identifiers (DIDs) and verifiable credentials (VCs). Through this integration, stakeholders retain direct control over their digital identities while enabling cryptographically verifiable and privacy-preserving interactions across organizational boundaries. The system architecture follows a multi-layered design that addresses identity management, credential lifecycle handling, authentication, access control, and governance. To move beyond conceptual analysis, the framework is realized as a functional research prototype using standardized SSI technologies and an agent-based architecture. The implemented system supports end-to-end credential workflows, including secure connection establishment, credential issuance, selective disclosure, and proof-based verification, without reliance on centralized identity providers or authentication authorities. Experimental validation conducted in a controlled environment confirms correct execution of identity and credential lifecycles, decentralized authentication, and privacy-preserving verification. The evaluation focuses on functional validation within a controlled prototype environment and does not include large-scale scalability benchmarking. These results demonstrate the feasibility of identity-centric governance mechanisms in agricultural supply chains using standardized SSI technologies.**

Keywords*—Blockchain; self-sovereign identity (ssi); agricultural supply chain; verifiable credentials; decentralized identity; privacy-preserving authentication

I. Introduction

The agricultural supply chain (ASC) plays a fundamental role in global food systems by linking producers, processors, distributors, and consumers through a sequence of interdependent activities that ensure food availability, quality, and safety. Despite its critical importance, the ASC continues to face persistent challenges related to traceability, data integrity, and trust among participating entities [1]. In practice, these challenges are exacerbated by fragmented information flows and limited mechanisms for independently verifying stakeholder actions across organizational boundaries. Traditional supply chain management systems, which are predominantly centralized and fragmented, often fail to provide end-to-end visibility, thereby limiting the ability to verify product authenticity and ensure compliance with quality and safety standards. Moreover, information asymmetry and data silos across stakeholders significantly hinder transparency and impede effective coordination throughout production, logistics, and distribution stages [2].

Issues related to identity verification and data authenticity are particularly pronounced in agricultural supply chains. Farmers, cooperatives, logistics providers, and certification bodies often rely on third-party intermediaries for authentication, certification, and market access. While such intermediaries simplify coordination, they also concentrate trust and control within centralized entities, increasing operational overhead and introducing privacy and accountability risks. Consequently, the absence of a unified, tamper-resistant, and stakeholder-centric identity mechanism represents a major barrier to the development of transparent, equitable, and verifiable agricultural value chains [3]. These traceability failures and certification disputes in agri-food systems further highlight that transparency without verifiable stakeholder identity remains insufficient for ensuring accountability and timely dispute resolution [4], [5].

Blockchain technology has been widely recognized as a promising approach to addressing several of these challenges. By maintaining a distributed, immutable ledger of transactions, blockchain enables auditability, data integrity, and tamper-resistance without requiring centralized control [6]. However, blockchain-based transparency alone does not guarantee trust if the identities of participating entities cannot be reliably verified or held accountable [7]. In parallel, Self-Sovereign Identity (SSI) has emerged as a paradigm in digital identity management that enables individuals and organizations to create, own, and manage their digital identities independently of centralized authorities. SSI systems rely on decentralized identifiers (DIDs) and verifiable credentials (VCs) to support cryptographically verifiable and selectively disclosable identity assertions [8]. The convergence of blockchain and SSI technologies, therefore, offers a complementary mechanism for managing both

*Corresponding author.

transactional data and stakeholder identities in a decentralized and verifiable manner.

Within agricultural supply chains, such an integration has the potential to empower stakeholders, particularly smallholder farmers, by granting them direct control over their digital identities and credentials, while simultaneously enabling verifiable, privacy-preserving data exchange across the supply chain [9]. Prior research has demonstrated the applicability of blockchain for enhancing traceability and transparency in agri-food logistics [10], [11]. Nevertheless, most existing solutions emphasize product-centric traceability and transactional logging, with limited attention to decentralized identity management, stakeholder authentication, and governance. Similarly, although SSI frameworks have been extensively explored in domains such as finance, healthcare, and education [12], their systematic application to agricultural supply chains remains comparatively underexplored. Beyond theoretical promise, the effective adoption of blockchain-enabled SSI solutions in agriculture requires practical validation through implemented systems that demonstrate real operational workflows [5],[13]. Without such validation, proposed frameworks risk remaining abstract and disconnected from deployment-level considerations, including identity lifecycle management, credential exchange, user consent, and interoperability among heterogeneous stakeholders.

Motivated by this research gap, this study not only proposes a blockchain-based self-sovereign identity framework but also realizes the proposed design through a functional research prototype, referred to as Ziraai SSI. The implemented system operationalizes decentralized identifiers, verifiable credentials, and privacy-preserving authentication mechanisms using agent-based SSI technologies. As a result, farmers, issuers, and verifiers can interact through secure, standards-compliant credential workflows without relying on centralized identity providers. Accordingly, this work addresses the limitations of existing approaches by developing a blockchain-based self-sovereign identity framework tailored to agricultural supply chains. The framework leverages blockchain-based trust anchoring to ensure data integrity and auditability, while SSI principles aim to decentralize identity control and authentication. The main contributions of this study are summarized as follows:

- The design of a multi-layered blockchain–SSI framework that explicitly centers on stakeholder identity, authentication, and governance within agricultural supply chains.
- The formulation of a credential lifecycle mechanism encompassing issuance, selective disclosure, verification, and governance processes aligned with agricultural traceability requirements.
- The practical implementation and validation of the proposed framework, through the Ziraai SSI research prototype, demonstrate end-to-end decentralized identity and credential management for agricultural stakeholders.
- A prototype-level evaluation emphasizing functional feasibility and security-property realization within a controlled environment.

The remainder of this study is organized as follows. Section II reviews related work on blockchain- and SSI-based approaches in supply chain contexts and identifies key research gaps. Section III presents the proposed Blockchain-Based SSI framework and its architectural components. Section IV describes the system design and implementation details of the Ziraai SSI prototype. Section V discusses the experimental results and validation outcomes. Finally, Section VI concludes the study and outlines directions for future research.

II. Related Work and Motivation

The agricultural supply chain (ASC) has historically relied on centralized information systems for data management, certification, and identity verification. Siloed databases, manual record-keeping processes, and limited interoperability across organizational boundaries are typically characteristic of such systems. As a consequence, trust is often established procedurally rather than cryptographically, making verification dependent on institutional authority rather than on independently verifiable evidence. These limitations expose agricultural ecosystems to data inconsistencies, unauthorized modifications, and operational inefficiencies, ultimately undermining trust among supply chain participants [1]. In particular, the absence of a unified, verifiable digital identity framework for stakeholders—such as farmers, processors, distributors, and regulators—amplifies risks of fraud, accountability gaps, and asymmetric information sharing within agricultural ecosystems.

A. Traditional Traceability and Identity Systems

Conventional agricultural traceability mechanisms have primarily relied on study-based documentation or centralized databases that record transactions at isolated points in the supply chain [14]. While these approaches provide some oversight, they lack real-time synchronization and are vulnerable to data loss, manipulation, and delayed verification. In centralized architectures, trust is concentrated in a single authority, creating single points of failure and increasing exposure to security breaches and unauthorized access [15]. From a governance perspective, such architectures also constrain independent auditing, as verification processes remain tightly coupled with the system operator.

Efforts to digitalize agricultural operations—through enterprise resource planning (ERP) systems, cloud-based farm management platforms, and centralized certification portals—have partially addressed traceability concerns. However, these systems often suffer from limited interoperability and opaque auditing mechanisms, leading to fragmented data ownership and asymmetric access to information [16]. Moreover, identity verification within these platforms typically relies on username–password credentials or centralized authentication services, which provide limited assurance regarding the real-world legitimacy of participating entities and introduce risks related to credential compromise, privacy leakage, and misuse of personal data [17].

B. Blockchain for Agricultural Traceability

Blockchain technology has emerged as a promising approach for enhancing transparency and integrity in supply chain systems by enabling distributed, immutable record-

keeping among mutually untrusted entities. Numerous studies have explored blockchain-based approaches for improving agricultural traceability, food safety, and quality assurance. For example, Kamble et al. [6] proposed a blockchain-enabled model to enhance provenance tracking and operational efficiency in agricultural supply chains, while Feng et al. [3] demonstrated the potential of smart contracts for automating quality assurance processes. Zhao et al. [9] provided a comprehensive synthesis of blockchain applications in agri-food value chains, highlighting improvements in accountability and auditability.

Despite these advances, blockchain-based agricultural systems frequently treat identity as an implicit or external assumption rather than as a first-class design concern. In many implementations, identity verification is either delegated to centralized authorities or represented through application-specific identifiers that lack cryptographic portability across systems. It weakens the overall trust model, as the integrity of on-chain data ultimately depends on the authenticity and accountability of off-chain actors who generate it [11].

C. Evolution of Self-Sovereign Identity (SSI)

Self-Sovereign Identity (SSI) represents a paradigm shift in digital identity management by emphasizing user autonomy, decentralization, and privacy preservation [18]. SSI frameworks are built on decentralized identifiers (DIDs) and verifiable credentials (VCs), enabling individuals and organizations to control their identity attributes and selectively disclose information without relying on centralized identity providers [19]. A defining characteristic of SSI systems is privacy by design, in which sensitive identity data remains under the control of the credential holder and is shared only via cryptographically verifiable proofs [20].

SSI has gained significant traction in domains where trust, privacy, and regulatory compliance are paramount, including finance, healthcare, and education [21]. In contrast, its application within agricultural supply chains remains comparatively limited [13]. This gap is particularly notable given the multi-actor, cross-organizational, and trust-fragmented nature of agricultural ecosystems, where stakeholders often lack shared identity infrastructure or centralized governance mechanisms [22].

D. Combining Blockchain and SSI for Secure Supply Chains

The integration of blockchain and SSI technologies offers complementary benefits for building decentralized trust infrastructures. Blockchain provides immutability, transparency, and distributed consensus, while SSI enables user-centric identity ownership and selective credential disclosure [23]. Together, these technologies support identity-aware trust architectures in which entities can interact securely without dependence on centralized intermediaries.

Several studies have examined partial integrations of blockchain and SSI across different application domains. Shehu et al. [24] discussed the role of SSI in enabling interoperable and verifiable identity ecosystems when combined with distributed ledgers. Khoualdi et al. [25] investigated the challenges associated with integrating SSI, IoT, and blockchain for traceability applications, identifying scalability and standardization as key obstacles. Ahmed et al. [26] further emphasized the potential of blockchain-based identity management to enhance transparency and governance in future supply chains. However, most existing works remain limited to architectural discussions or small-scale experimental demonstrations. In particular, they provide limited insight into complete system realization, end-to-end credential workflows, and implementation-level validation within agricultural environments [27].

E. Research Gaps and Motivation

Based on the reviewed literature, several research gaps can be identified:

- Limited integration of decentralized identity mechanisms within agricultural blockchain systems, with most solutions focusing primarily on transactional traceability.
- Insufficient empirical validation of SSI-based approaches in agricultural contexts, despite their demonstrated applicability in other regulated domains.
- Lack of identity-centric governance models capable of supporting heterogeneous stakeholders without centralized control.
- Scarcity of implementation-level studies demonstrating complete issuer–holder–verifier workflows using standardized SSI protocols.

Notably, there is a lack of research demonstrating a complete and operational SSI-based agricultural identity system that explicitly supports issuer, holder, and verifier roles using standardized SSI protocols and evaluates these interactions through implemented, end-to-end workflows. As a result, much of the existing literature remains either conceptual or limited to partial demonstrations that do not fully address identity lifecycle management and governance requirements. This gap motivates the design and implementation of Ziraai SSI as an identity-centric, standards-compliant prototype tailored specifically to agricultural supply chain governance. To further clarify the positioning of the proposed approach, Table I provides a comparative analysis of Ziraai SSI against representative categories of existing agricultural traceability systems, blockchain-based solutions, and SSI frameworks. The comparison highlights that most existing approaches either focus on transactional traceability without decentralized identity support or explore SSI concepts without domain-specific, implementation-level validation. In contrast, Ziraai SSI integrates blockchain-based trust anchoring with self-sovereign identity mechanisms, delivering a functional research prototype that supports complete, end-to-end credential workflows in an agricultural context.

TABLE. I. COMPARISON OF ZIRAAI SSI WITH EXISTING BLOCKCHAIN- AND SSI-BASED APPROACHES

Approach Category	Blockchain	SSI	DID	VC	Selective Disclosure	Prototype
Conventional agricultural traceability systems	x	x	x	x	x	✓
Blockchain-based agricultural traceability	✓	x	x	x	x	Partial
SSI frameworks (non-agricultural domains)	x	✓	✓	✓	✓	✓
Blockchain –SSI conceptual models for supply chains	✓	✓	✓	✓	✓	x
Ziraai SSI (this work)	✓	✓	✓	✓	✓	✓

While decentralized systems improve transparency and privacy, they require more coordination between participating organizations and can be more complex to deploy than centralized identity systems. Scalability and secure wallet key management are also important practical considerations in real-world implementations [28].

III. PROPOSED BLOCKCHAIN-BASED SSI FRAMEWORK FOR AGRICULTURE

This section presents the conceptual and architectural design of the proposed blockchain-based self-sovereign identity (SSI) framework for agricultural supply chains. The framework is intended to address persistent challenges related to stakeholder authentication, data integrity, transparency, and trust in decentralized agricultural ecosystems. Rather than treating identity as a supporting component of traceability, the proposed framework positions identity as a foundational element of supply chain governance. By combining blockchain-based trust anchoring with SSI principles, the framework establishes a generalizable model for identity-centric supply chain interactions that is independent of specific implementation technologies. While the framework is presented at an abstract level to ensure reusability across different agricultural contexts, its design is explicitly informed by practical deployment considerations, and it directly guides the concrete realization described in Section IV.

A. Design Objectives

The design of the proposed blockchain-based self-sovereign identity (SSI) framework is guided by a set of objectives derived from the structural characteristics, trust relationships, and governance requirements of agricultural supply chains. These objectives aim to address limitations observed in existing centralized and semi-decentralized identity and traceability systems, while ensuring compatibility with standardized SSI ecosystems. The following design objectives therefore guide the framework:

- Decentralized Identity Control: Empower agricultural stakeholders to manage their own digital identities using decentralized identifiers (DIDs) and verifiable credentials (VCs), without reliance on centralized identity or certification authorities. This objective reflects the need for stakeholder autonomy and for identity to be long-term portable across organizational and institutional boundaries [29].
- Tamper-Resistant Identity and Data Management: Ensure that identity-related metadata, credential schemas, and verification artifacts are anchored on a decentralized ledger, providing integrity, authenticity, and accountability for identity-linked interactions without dependence on mutable centralized databases or trusted third parties.
- Privacy and Selective Disclosure: Enable entities to disclose only the minimum necessary identity attributes during verification processes through selective disclosure mechanisms and cryptographic proof-based credential presentation. This objective is particularly critical in agricultural contexts, where stakeholders may be reluctant to share sensitive personal or organizational information [30].
- Transparency and Auditability: Provide stakeholders and oversight entities with verifiable and independently auditable identity-related records, while ensuring that sensitive credentials and personal information remain under the exclusive control of their respective holders [25].
- Scalability and Interoperability: Support integration with existing agricultural platforms and interoperability with consortium-based or multi-network identity infrastructures. This ensures that the framework can adapt to heterogeneous deployment environments and evolving governance models [20].

Together, these objectives define the conceptual scope of the proposed framework and guide the architectural and implementation choices discussed in subsequent sections.

B. Framework Overview

The proposed framework adopts a multi-layered architectural design that separates identity management, authentication, access control, data interaction, and governance concerns. This separation of concerns enables modularity and extensibility, allowing individual layers to evolve independently while preserving cryptographic trust guarantees across the system. Within this framework, blockchain technology is employed primarily as a decentralized trust anchor rather than as a general-purpose transaction execution platform. Specifically, the blockchain component anchors decentralized identifiers (DIDs), credential schemas, credential definitions, and governance-related metadata, enabling verifiable identity resolution and long-term auditability. This design deliberately avoids unnecessary on-chain transaction overhead while retaining tamper resistance for identity-critical information. The

framework is structured into five logical layers that collectively support secure, privacy-preserving interactions in the agricultural supply chain. Each layer addresses a distinct functional responsibility, ranging from stakeholder identity representation to governance and trust management.

C. Layer 1 – Stakeholder Identity and Credential Layer

This foundational layer defines how digital identities are created, represented, and managed within the agricultural supply chain.

- Each stakeholder (e.g., farmer, cooperative, distributor, certification authority, regulator) is associated with a decentralized identifier (DID) compliant with W3C standards.
- Authorized entities issue verifiable credentials (VCs) that attest to stakeholder attributes, certifications, or roles.
- The holder controls the credentials and can present them selectively for verification.
- Credential revocation and status information are conceptually supported by publicly verifiable registries, enabling detection of expired or invalid credentials.

By decoupling identity ownership from centralized authorities, this layer establishes accountability while preserving stakeholder autonomy.

D. Layer 2 – Data Recording and Verification Layer

This layer governs how supply-chain events and identity-linked actions are recorded and verified.

- All interactions are cryptographically bound to verified identities.
- Data submissions are accepted only after successful credential validation.
- Immutable records enable retrospective auditing and compliance verification.

Rather than prioritizing transaction throughput, this layer emphasizes the integrity and authenticity of identity-bound data, which is particularly critical in agricultural governance and certification scenarios.

E. Layer 3 – Trust and Authentication Layer

The trust and authentication layer ensures that all interactions occur between verified participants.

- Mutual authentication is performed using cryptographic proofs derived from DIDs and VCs.
- Selective disclosure mechanisms enable attribute-level verification without revealing full credentials.
- Authentication policies prevent unauthorized access to identity-protected resources.

This layer enables privacy-preserving trust establishment across heterogeneous agricultural stakeholders without requiring shared identity infrastructure.

F. Layer 4 – Access Control and Policy Layer

This layer defines how access permissions and disclosure policies are enforced.

- Access control decisions are based on credential attributes rather than static roles.
- Stakeholders retain control over what data is shared, with whom, and under what conditions.
- Policy enforcement is auditable and aligned with regulatory requirements.

By decoupling identity from application-specific logic, this layer supports flexible and context-aware governance models.

G. Layer 5 – Governance and Consensus Layer

The governance layer defines the rules under which identities and credentials are recognized as trustworthy.

- Credential schemas and issuer policies are governed collectively.
- Regulatory authorities can participate as oversight entities without assuming centralized control.
- Trust anchors ensure long-term verifiability of identity claims.

This layer enables institutional trust without centralization, which is a key requirement in multi-stakeholder agricultural ecosystems. In the current prototype, governance follows a consortium-based trust model in which authorized issuers are pre-approved before registering credential schemas on the ledger. Schema registration requires prior agreement among consortium participants before activation. Disputes related to credential validity are addressed through revocation registry updates and publicly verifiable status checks. Formal on-chain voting or automated consensus mechanisms were not implemented in this prototype; however, the architecture allows such mechanisms to be incorporated in future extensions.

H. Framework Workflow

At a conceptual level, the framework supports the following workflow:

- Stakeholder identity establishment through decentralized identifiers (DIDs)
- Credential issuance by authorized entities
- Identity-bound data interaction across supply chain stages.
- Selective disclosure and cryptographic verification.
- Auditability and compliance validation.

This abstract workflow directly informs the concrete realization described in Section IV.

I. Security Considerations

The proposed framework addresses core security requirements, including integrity, authentication, confidentiality, non-repudiation, and unlinkability, through cryptographic identity mechanisms rather than centralized

enforcement. These properties are derived from decentralized identifiers, verifiable credentials, and proof-based authentication, ensuring distributed trust, resilience to compromise, and reduced reliance on centralized authorities.

IV. System Design and Implementation Details

This section describes the practical realization of the proposed blockchain-based self-sovereign identity framework through the Ziraai SSI research prototype. In contrast to the conceptual focus of Section III, this section details how the framework is instantiated using standardized self-sovereign identity technologies, thereby validating its feasibility under realistic interaction and governance assumptions. The implementation is intentionally designed as a research prototype and prioritizes standards compliance, architectural traceability, and workflow correctness over production-scale optimization.

A. Implementation Environment

The Ziraai SSI platform is implemented using an agent-based SSI architecture aligned with current industry and standards-body specifications. The selected implementation environment reflects explicit design decisions motivated by interoperability, modularity, and alignment with widely adopted SSI ecosystems.

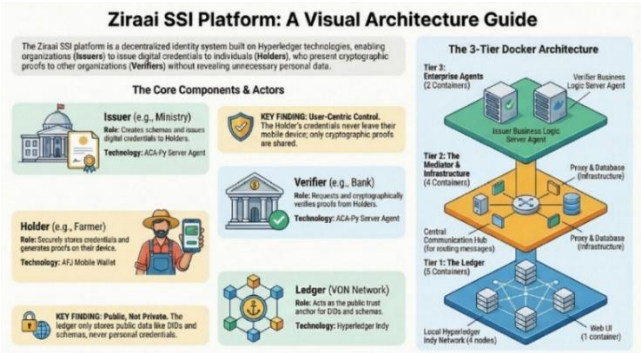


Fig. 1. Ziraai SSI Platform visual architecture.

- Identity ledger: Hyperledger Indy (local development network), used exclusively as a decentralized trust anchor for DIDs, credential schemas, and definitions
- SSI framework: Hyperledger Aries, implemented through ACA-Py agents and Aries-compatible wallets
- Communication protocol: DIDComm, enabling secure, peer-to-peer agent communication
- User interfaces: Web-based dashboards for issuer and verifier roles, and a mobile wallet application for farmers and credential holders
- Deployment: Containerized microservices orchestrated using Docker to support modular experimentation and reproducibility

The choice of Hyperledger Indy and Aries enables strict separation between identity anchoring, credential exchange, and application logic, which is essential for evaluating identity-centric architectures. Fig. 1 provides a visual architecture of the Ziraai SSI platform.

B. System Architecture and Modules

The system architecture mirrors the layered framework described in Section III while introducing concrete implementation components.

- SSI agents implement identity and credential logic for each stakeholder.
- Application controllers mediate interactions between user interfaces and agents.
- Wallet components provide secure credential storage and user-controlled disclosure.
- Verification services enable proof validation and audit support.

Each implementation module maps directly to a corresponding conceptual layer of the proposed framework, ensuring architectural consistency and traceability between design intent and system realization. Fig. 2 shows the architecture of the Ziraai SSI platform and the modules.

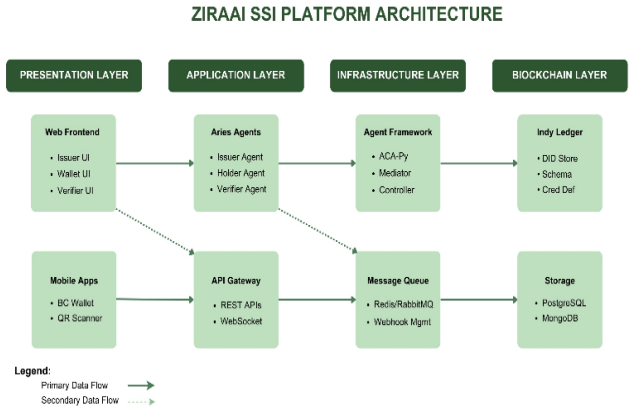


Fig. 2. Architecture and layers of the Ziraai SSI platform.

C. SSI Credential Lifecycle and Workflow

Ziraai SSI implements a complete SSI credential lifecycle consistent with standardized SSI protocols.

- Secure connection establishment using agent-mediated invitations
- Credential issuance with explicit holder consent
- Credential storage within a holder-controlled wallet
- Privacy-preserving credential verification through selective disclosure

The issuance and verification processes follow established SSI interaction patterns, ensuring compatibility with external SSI agents and minimizing reliance on application-specific extensions. The sequence of interactions shown in Fig. 3 demonstrates how issuer, holder, and verifier roles collaborate through standardized SSI protocols to support an end-to-end, trust-preserving credential lifecycle.

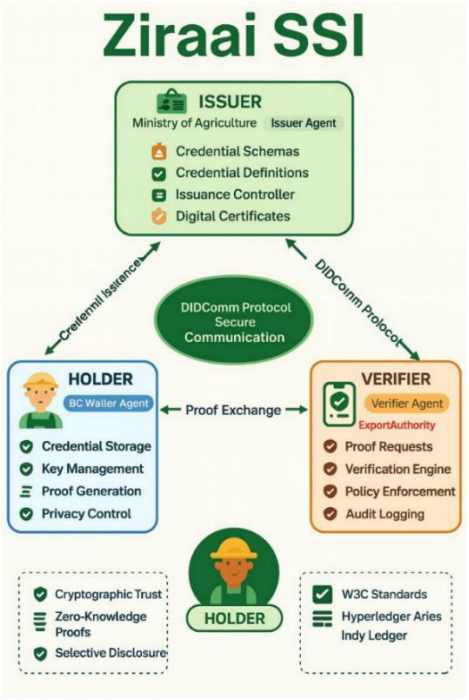


Fig. 3. Modules of the Ziraai SSI platform.

1) Credential issuance workflow: The process of issuing a verifiable credential (VC) involves the farmer (Holder), the issuer, and their respective agents as specified in Algorithm 1.

Algorithm 1: Credential Issuance

1. Issuer Admin (Web UI): Clicks "Create Invitation".
2. Issuer Controller: Calls ACA-Py agent to create a connection invitation.
3. Issuer Agent: Generates invitation and returns it.
4. Issuer Controller: Displays invitation as a QR code on the UI.
5. Farmer (Wallet): Scans the QR code.
6. Wallet Agent: Sends a connection request to the Issuer Agent.
7. Issuer Agent: Responds with a connection-response. Connection is now 'active'.
8. Issuer Admin (Web UI): Selects the active connection and fills out the credential details.
9. Issuer Controller: Calls ACA-Py agent to send a credential offer.
10. Issuer Agent: Sends the credential offer to the Wallet Agent.
11. Farmer (Wallet): Accepts the offer.
12. Wallet Agent: Sends a credential-request to the Issuer Agent.
13. Issuer Agent: Issues the signed credential and sends the issue-credential message.
14. Wallet Agent: Receives and stores the credentials securely.
15. Farmer (Wallet): Receives a notification that the credential has been stored.

2) Credential verification workflow: The process of verifying a credential involves the farmer (Holder), the Verifier, and their agents, as given in Algorithm 2.

Algorithm 2: Verification Flow

1. Verifier Admin (Web UI): Clicks "Create Proof Request Invitation".
2. Verifier Controller: Calls its ACA-Py agent to create a connection invitation.
3. Farmer (Wallet): Scans the QR code to establish a connection
4. Verifier Admin (Web UI): Selects the active connection and initiates a proof request.
5. Verifier Controller: Calls its ACA-Py agent to send a proof request.
6. Verifier Agent: Sends the proof-request to the Wallet Agent, specifying the required attributes.
7. Farmer (Wallet): Is prompted to share the requested information. The wallet automatically finds the required credentials.
8. Farmer (Wallet): Consents to sharing.
9. Wallet Agent: Generates a zero-knowledge proof and sends the present proof message.
10. Verifier Agent: Receives the proof and cryptographically verifies it against the public information on the Indy ledger.
11. Verifier Agent: If verification is successful, it notifies the Verifier Controller.
12. Verifier Admin (Web UI): Sees the verification result updated on the dashboard.

D. Agent-based Communication and Authentication

All interactions within the Ziraai SSI system use peer-to-peer agent communication.

- Pairwise decentralized identifiers are used for each bilateral relationship, preventing identifier reuse across contexts.
- Encrypted DIDComm messaging ensures the confidentiality and integrity of exchanged messages.
- Authentication is achieved through cryptographic proof of DID control rather than password-based or centralized mechanisms.

This communication model demonstrates the feasibility of decentralized, identity-bound authentication in agricultural supply chain environments, where centralized identity infrastructure is often unavailable or impractical.

E. Security and Privacy Mechanisms

The Ziraai SSI prototype enforces security and privacy guarantees through cryptographic identity primitives and agent-mediated protocols.

- Credential integrity and authenticity are ensured through digitally signed verifiable credentials.
- Authentication relies on cryptographic proof of control over decentralized identifiers.
- Privacy preservation is achieved through selective disclosure and proof-based verification.
- Pairwise decentralized identifiers mitigate correlation across interactions.

Private keys and credentials are stored within secure wallet environments under the exclusive control of credential holders, minimizing centralized aggregation of sensitive identity data and reducing systemic risk. Table II summarizes the security properties and how they are achieved in the Ziraai SSI platform.

TABLE. II. SECURITY PROPERTIES AND THEIR IMPLEMENTATION IN ZIRAAI SSI

Security Property	Implementation Mechanism in Ziraai SSI
Integrity	Digitally signed verifiable credentials
Authentication	DID-based cryptographic proof of control
Privacy	Selective disclosure via attribute proofs
Confidentiality	Encrypted DIDComm messaging
Non-repudiation	Identity-bound credential issuance
Unlinkability	Pairwise decentralized identifiers

F. Implementation Scope and Limitations

The Ziraai SSI system is implemented as a research prototype intended to validate architectural design choices and demonstrate the functional feasibility of blockchain-enabled self-sovereign identity in agricultural supply chains. Accordingly, the implementation prioritizes correctness, transparency, and standards alignment over performance optimization or large-scale deployment.

Several limitations are acknowledged. First, the prototype is evaluated in a controlled environment. It does not include large-scale ledger deployment or stress testing, which limits conclusions regarding scalability and throughput under real-world workloads. Second, while the underlying SSI infrastructure supports credential revocation, recovery, and status management, these capabilities are not fully exposed through user-facing interfaces in the current implementation. Third, user interface design is intentionally minimal and may require further refinement to support adoption by non-technical stakeholders. These limitations are inherent to the scope of a research prototype and serve to identify concrete directions for future enhancement rather than deficiencies in the proposed framework.

V. RESULTS AND DISCUSSION

This section presents the results of implementing and evaluating the proposed blockchain-based self-sovereign identity framework using the Ziraai SSI research prototype. The discussion focuses on functional validation, security and privacy behavior, and governance implications, rather than performance benchmarking, in alignment with the identity-centric role of the proposed system within agricultural supply chains.

A. Evaluation Objectives and Metrics

The primary objectives of the evaluation were as follows:

- To verify the feasibility of implementing decentralized identity and credential management for agricultural stakeholders using standardized SSI technologies.
- To validate end-to-end identity and credential workflows, including identity creation, credential issuance, storage, and privacy-preserving verification.
- To assess how the implemented system improves transparency, trust, and privacy compared with conventional centralized identity management approaches commonly used in agricultural supply chains.

The evaluation deliberately emphasizes functional correctness, interaction completeness, and security behavior, rather than throughput- or latency-oriented performance metrics. This choice reflects the governance-oriented nature of identity systems, where correctness and trust guarantees are often more critical than raw performance.

B. Practical Workflow Validation

The Ziraai SSI prototype was evaluated through a series of end-to-end workflow executions involving issuer, holder (farmer), and verifier roles, as described in Section IV.

- Identity creation: Each stakeholder successfully generated a decentralized identifier (DID) and established secure, pairwise connections using agent-based communication. It validated correct DID generation, key binding, and mutual authentication without reliance on centralized identity services.
- Credential issuance: Authorized issuers issued verifiable credentials following consent-driven workflows. Credentials were securely transmitted and stored in holder-controlled wallets, confirming the correct execution of issuance protocols and the preservation of holder autonomy.
- Credential verification: Verifiers issued proof requests specifying required attributes. Holders responded using selective disclosure mechanisms, and cryptographic verification was completed successfully without disclosure of unnecessary credential attributes.

Collectively, these observations confirm that the implemented system supports a complete and operational SSI credential lifecycle, consistent with the conceptual workflow defined in Section III and the implementation described in Section IV.

C. System Behavior and Performance Discussion

The implemented system was evaluated with respect to key security and privacy properties relevant to agricultural identity management.

- Integrity: Credential authenticity and integrity were preserved through cryptographic signatures and verification against issuer decentralized identifiers and ledger-anchored credential definitions.
- Authentication: Only entities controlling valid decentralized identifiers and corresponding private keys were able to participate in issuance and verification workflows, demonstrating identity-bound authentication without centralized intermediaries.
- Privacy: Selective disclosure mechanisms enabled attribute-level verification, minimizing data exposure during credential presentation. The use of pairwise decentralized identifiers further reduced the risk of cross-context correlation.
- Accountability: Identity-bound interactions ensured that credential issuance and verification events could be attributed to authenticated entities, supporting auditability without exposing sensitive identity data.

- Invalid or Malicious Submissions: Basic negative test cases were performed to evaluate system robustness. Replayed proof messages were rejected due to nonce-based challenge–response mechanisms, and invalid or improperly signed credentials failed cryptographic verification against ledger-anchored definitions. These results indicate that the prototype is resistant to replay and forgery attempts within the evaluated environment.

These observations indicate that the security and privacy objectives outlined in Section III are realized in practice through the implementation mechanisms described in Section IV.

D. Comparative Discussion with Centralized Systems

When compared with conventional centralized identity management systems, the proposed framework exhibits several notable advantages. First, the framework eliminates single points of trust and failure by decentralizing identity ownership and verification. This is particularly relevant in agricultural ecosystems, where stakeholders are geographically distributed and often operate across institutional boundaries. Second, auditability and transparency are improved through verifiable credential workflows rather than opaque database records controlled by a single authority. Verification can be performed independently by any authorized verifier without prior trust relationships. Third, privacy protection is enhanced, as stakeholders retain control over identity attributes and disclosure decisions. Unlike centralized systems, sensitive identity information is not persistently stored or aggregated by service providers. These characteristics suggest that blockchain-enabled self-sovereign identity provides qualitative governance advantages over centralized approaches, particularly in trust-fragmented agricultural environments. The comparison focuses on architectural and governance differences rather than performance benchmarking. Unlike centralized systems, the proposed framework distributes trust and enables cryptographic verification without reliance on intermediary authorities.

E. Discussion of Strengths and Limitations

While the Ziraai SSI prototype demonstrates functional feasibility and standards-compliant identity workflows, several practical considerations were observed during implementation and evaluation. First, establishing SSI connections introduces additional interaction steps compared to traditional username–password authentication models. Although these steps improve security and trust, they may increase initial onboarding complexity, particularly for non-technical users. Second, the use of decentralized identifiers and pairwise relationships increases the number of cryptographic artifacts managed by agents and wallets. This trade-off is inherent to privacy-preserving identity systems and necessitates robust wallet usability and key management support. Third, the evaluation was conducted in a controlled environment and did not include large-scale deployment or performance-stress testing. As a result, conclusions regarding scalability and operational performance under real-world workloads remain limited. These trade-offs reflect design choices rather than implementation deficiencies and highlight areas where usability engineering and system optimization will be required for production deployment.

F. Implications for Agricultural Supply Chain Governance

The evaluation results indicate that self-sovereign identity can serve as a feasible architectural foundation for identity-centric governance in agricultural supply chains. By enabling stakeholders to authenticate, certify, and verify interactions without centralized intermediaries, the proposed framework supports distributed trust establishment while preserving privacy and accountability. In particular, the ability to issue and verify credentials—such as farmer registrations, certification attestations, and compliance proofs—through standardized SSI protocols creates new opportunities for transparent yet privacy-aware governance models involving regulators, cooperatives, and market participants. Although regulator-operated nodes and multi-issuer governance competition were not experimentally simulated in the current prototype, the architecture is designed to accommodate such roles in future multi-institutional deployments.

VI. Conclusion and Future Work

This study presented Ziraai SSI, a blockchain-based self-sovereign identity prototype designed to enhance transparency, security, and decentralized trust in agricultural supply chains. By integrating blockchain-based trust anchoring with self-sovereign identity mechanisms, the proposed approach addresses fundamental limitations of conventional centralized identity and data management systems, including fragmented trust relationships, limited auditability, and privacy risks. Rather than focusing solely on transactional traceability, the proposed framework explicitly positions identity as a core governance primitive within agricultural ecosystems. A key contribution of this work lies in the design of a layered, identity-centric blockchain–SSI framework tailored to the structural and governance characteristics of agricultural supply chains, and in its realization through a functional research prototype. The Ziraai SSI implementation operationalizes decentralized identifiers, verifiable credentials, and privacy-preserving authentication using standardized SSI technologies, enabling interactions among issuers, holders, and verifiers through secure, standards-compliant credential workflows. The prototype-level evaluation demonstrates the feasibility of decentralized authentication, selective disclosure, and identity-bound accountability without reliance on centralized identity providers.

The results indicate that self-sovereign identity can serve as a feasible architectural foundation for establishing trust and governance in distributed agricultural environments, particularly when stakeholders are heterogeneous and centralized identity infrastructures are impractical or undesirable. By enabling cryptographically verifiable identity interactions while preserving stakeholder autonomy and privacy, the proposed approach supports more transparent and accountable supply chain governance models. While the current implementation is research-oriented and limited to a controlled consortium setting, it provides a concrete baseline for future system extensions and deployment-oriented investigations. Several directions for future work are identified. First, the system can be extended to support comprehensive credential lifecycle governance, including user-facing credential revocation, recovery, and status management mechanisms.

Second, large-scale deployment studies involving multiple institutions and regulatory authorities are required to evaluate scalability, interoperability, and operational resilience under realistic conditions. Third, integration with complementary technologies—such as IoT-based data collection and analytics—may further enhance traceability, monitoring, and decision support within agricultural supply chains. Finally, improving usability and accessibility for non-technical stakeholders remains a critical requirement for real-world adoption and warrants focused human-centered design efforts.

REFERENCES

[1] M. M. Aung and Y. S. Chang, "Traceability in a food supply chain: Safety and quality perspectives," Food Control, vol. 39, pp. 172–184, May 2014, doi: 10.1016/j.foodcont.2013.11.007.

[2] Y. Dong, S. F. Ahmad, M. Irshad, M. Al-Razgan, Y. A. Ali, and E. M. Awwad, "The Digitalization Paradigm: Impacts on Agri-Food Supply Chain Profitability and Sustainability," Sustainability, vol. 15, no. 21, p. 15627, Nov. 2023, doi: 10.3390/su152115627.

[3] H. Feng, X. Wang, Y. Duan, J. Zhang, and X. Zhang, "Applying blockchain technology to improve agri-food traceability: A review of development methods, benefits and challenges," J. Clean. Prod., vol. 260, p. 121031, Jul. 2020, doi: 10.1016/j.jclepro.2020.121031.

[4] F. Anastasiadis, I. Apostolidou, and N. Tsolakis, "Challenges and opportunities of supply chain traceability: insights from emergent agri-food sector," Supply Chain Management: An International Journal, vol. 30, no. 1, pp. 106–126, Feb. 2025, doi: 10.1108/SCM-05-2024-0322.

[5] J. Khoualdi, N. Ellouze, H. K. Ben Ayed, and J.-P. Kokora, "Blockchain-Driven SSI for Securing Privacy and Quality Assurance in Agri-Food Supply Chains," in 2025 International Wireless Communications and Mobile Computing (IWCMC), IEEE, May 2025, pp. 485–490. doi: 10.1109/IWCMC65282.2025.11059697.

[6] S. S. Kamble, A. Gunasekaran, and R. Sharma, "Modeling the blockchain enabled traceability in agriculture supply chain," Int. J. Inf. Manage., vol. 52, p. 101967, Jun. 2020, doi: 10.1016/j.ijinfomgt.2019.05.023.

[7] G. Lv, C. Song, P. Xu, Z. Qi, H. Song, and Y. Liu, "Blockchain-Based Traceability for Agricultural Products: A Systematic Literature Review," Agriculture, vol. 13, no. 9, p. 1757, Sep. 2023, doi: 10.3390/agriculture13091757.

[8] P. Vern, A. Panghal, R. S. Mor, and S. S. Kamble, "Blockchain technology in the agri-food supply chain: a systematic literature review of opportunities and challenges," Management Review Quarterly, vol. 75, no. 1, pp. 643–675, Feb. 2025, doi: 10.1007/s11301-023-00390-0.

[9] G. Zhao et al., "Blockchain technology in agri-food value chain management: A synthesis of applications, challenges and future research directions," Comput. Ind., vol. 109, pp. 83–99, Aug. 2019, doi: 10.1016/j.compind.2019.04.002.

[10] C. G. Schmidt and S. M. Wagner, "Blockchain and supply chain relations: A transaction cost theory perspective," Journal of Purchasing and Supply Management, vol. 25, no. 4, p. 100552, Oct. 2019, doi: 10.1016/j.pursup.2019.100552.

[11] Y. Wang, J. H. Han, and P. Beynon-Davies, "Understanding blockchain technology for future supply chains: a systematic literature review and research agenda," Supply Chain Management: An International Journal, vol. 24, no. 1, pp. 62–84, Jan. 2019, doi: 10.1108/SCM-03-2018-0148.

[12] M. Shuaib et al., "Land Registry Framework Based on Self-Sovereign Identity (SSI) for Environmental Sustainability," Sustainability (Switzerland), vol. 14, no. 9, 2022, doi: 10.3390/su14095400.

[13] T. Alar, G. Bharathy, U. Batra, and M. Prasad, "Evaluating self-sovereign identity solutions for agricultural supply chains: a systematic review," Frontiers in Blockchain, vol. 8, Jan. 2026, doi: 10.3389/fbloc.2025.1672752.

[14] R. M. Ellahi, L. C. Wood, and A. E.-D. A. Bekhit, "Blockchain-Based Frameworks for Food Traceability: A Systematic Review," Foods, vol. 12, no. 16, p. 3026, Aug. 2023, doi: 10.3390/foods12163026.

[15] A. Panwar, M. Khari, S. Misra, and U. Sugandh, "Blockchain in Agriculture to Ensure Trust, Effectiveness, and Traceability from Farm Fields to Groceries," Future Internet, vol. 15, no. 12, p. 404, Dec. 2023, doi: 10.3390/fi15120404.

[16] A. Banerjee, "An ERP and Planning System Enabled Decentralized Supply Chain Using Blockchain Technology," in Blockchain Driven Supply Chains and Enterprise Information Systems, Cham: Springer International Publishing, 2023, pp. 63–96. doi: 10.1007/978-3-030-96154-1_4.

[17] E. Ahoa, A. Kassahun, C. Verdouw, and B. Tekinerdogan, "Challenges and Solution Directions for the Integration of Smart Information Systems in the Agri-Food Sector," Sensors, vol. 25, no. 8, p. 2362, Apr. 2025, doi: 10.3390/s25082362.

[18] Christopher Allen, "The Path to Self-Sovereign Identity," https://www.lifewithalacrity.com/article/the-path-to-self-sovereign-identity/. Accessed: Jan. 20, 2026. [Online]. Available: https://www.lifewithalacrity.com/article/the-path-to-self-sovereign-identity/

[19] A. Preukschat and D. Reed, Self-sovereign identity. Manning Publications, 2021.

[20] L. Cocco, R. Tonelli, and M. Marchesi, "Blockchain and Self Sovereign Identity to Support Quality in the Food Supply Chain," Future Internet, vol. 13, no. 12, p. 301, Nov. 2021, doi: 10.3390/fi13120301.

[21] J. F. B. Duarte, G. S. Junior, G. F. C. da Silva, M. Pillon, G. P. Koslovski, and C. C. Miers, "COTTONTRUST: Reliability and Traceability in Cotton Supply Chain Using Self-sovereign Identity," 2024, pp. 85–97. doi: 10.1007/978-3-031-57853-3_8.

[22] I. A. Ibrahim and J. M. Truby, "FarmTech: Regulating the use of digital technologies in the agricultural sector," Food Energy Secur., vol. 12, no. 4, Jul. 2023, doi: 10.1002/fes3.483.

[23] Md. R. Ahmed, A. K. M. M. Islam, S. Shatabda, and S. Islam, "Blockchain-Based Identity Management System and Self-Sovereign Identity Ecosystem: A Comprehensive Survey," IEEE Access, vol. 10, pp. 113436–113481, 2022, doi: 10.1109/ACCESS.2022.3216643.

[24] A.-S. Shehu, A. Pinto, and M. E. Correia, "SPIDVerify: A Secure and Privacy-Preserving Decentralised Identity Verification Framework," in 2023 International Conference on Smart Applications, Communications and Networking (SmartNets), IEEE, Jul. 2023, pp. 1–7. doi: 10.1109/SmartNets58706.2023.10215588.

[25] J. Khoualdi, I. Abedlmoula, H. Kaffel, and D. Bensedrine, "Secure and Transparent Agri-Food Supply Chain Traceability with SSI and Blockchain-Enabled IoT," 2025, pp. 388–398. doi: 10.1007/978-3-031-87775-9_33.

[26] I. Ahmed, K. Toyoda, T. Nakano, S. Kasahara, S. R. Goyal, and T. H. Tran, "A Systematic Review on Blockchain-Enabled eKYC: Leveraging SSI and DID for Secure and Efficient Identity Verification," IEEE Internet Things J., pp. 1–1, 2025, doi: 10.1109/JIOT.2025.3597356.

[27] U. Peddibhotla, S. C. Chandran, P. Kumar, and R. Kumar, "Enhanced Supply Chain Management in Indian Agriculture Using SSI and Blockchain Leveraged by Digital Wallet," in 2024 16th International Conference on COMmunication Systems & NETworkS (COMSNETS), IEEE, Jan. 2024, pp. 427–429. doi: 10.1109/COMSNETS59351.2024.10427356.

[28] A. M. Buttar, M. A. Shahid, M. N. Arshad, and M. A. Akbar, "Decentralized Identity Management Using Blockchain Technology: Challenges and Solutions," 2024, pp. 131–166. doi: 10.1007/978-3-031-49593-9_8.

[29] C. Mazzocca, A. Acar, S. Uluagac, R. Montanari, P. Bellavista, and M. Conti, "A Survey on Decentralized Identifiers and Verifiable Credentials," IEEE Communications Surveys & Tutorials, vol. 27, no. 6, pp. 3641–3671, Dec. 2025, doi: 10.1109/COMST.2025.3543197.

[30] A. De Salve, A. Lisi, M. Cascino, P. Mori, and L. Ricci, "Selective Disclosure Approaches in Self-Sovereign Identity: An Experimental Comparison," IEEE Access, vol. 14, pp. 1152–1181, 2026, doi: 10.1109/ACCESS.2025.3649167.

Rubric-Relational Discourse Modeling with Counterfactual Explainability for Multi-Trait Automated Essay Scoring

N. Sreedevi[1], Dr.M.Madhusudhan Rao[2], Sridevi Dasam[3],
Roopa Traisa[4], Jasgurpreet Singh Chohan[5], V. Saranya[6], Ahmed I. Taloba[7]
Research Scholar, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India [1]
Associate professor, Department of English, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India [2]
Department of English-Velagapudi Ramakrishna Siddhartha School of Engineering-
Siddhartha Academy of Higher Education, Deemed-to-be-University, Vijayawada, Andhra Pradesh, India[3]
Department of Management-School of Management - UG, JAIN (Deemed to be University), Bangalore, Karnataka, India[4]
Marwadi University Research Center-Department of Mechanical Engineering-Faculty of Engineering & Technology,
Marwadi University, Rajkot, Gujarat, India[5]
Department of English, Panimalar Engineering College, Chennai, India[6]
Department of Computer Science-College of Computer and Information Sciences, Jouf University, Saudi Arabia [7]
Faculty of Computers and Information-Information System Department, Assiut University, Assiut, Egypt[7]

***Abstract*—Automated Essay Scoring (AES) systems often rely on holistic prediction and show weak alignment with rubric-based human evaluation. Existing deep learning approaches achieve moderate agreement but struggle to model discourse coherence and provide trait-faithful explanations. This study proposes a rubric-aware and discourse-faithful essay scoring framework that integrates contextual embeddings with sentence-level discourse modeling and rubric-specific attention. The framework generates both holistic and trait-level scores, while enabling counterfactual explanation of scoring decisions. Experiments conducted on the Learning Agency Lab – Automated Essay Scoring 2.0 dataset show that the proposed model achieves a Quadratic Weighted Kappa (QWK) of 0.86, Root Mean Square Error (RMSE) of 1.41, and Mean Absolute Error (MAE) of 1.12, outperforming CNN-LSTM, BERT-LSTM, and DeBERTa baselines. QWK evaluates ordinal agreement, while RMSE and MAE measure numerical prediction error. Trait-level performance reaches F1-scores of 0.89 for Content and 0.87 for Grammar, indicating strong rubric alignment. The proposed framework improves scoring reliability, interpretability, and consistency with human grading practices. It is suitable for large-scale educational assessment, formative feedback systems, and intelligent tutoring applications, offering a scalable and explainable solution for multi-trait essay evaluation.**

Keywords*—Automated Essay Scoring; rubric-aware modeling; discourse representation; counterfactual explainability; multi-task learning

I. Introduction

Automated Essay Scoring is now an indispensable part of educational assessment, especially high-stakes assessment like in online learning applications and large-scale assessments [1]. Traditional manual essay assessment is a time-consuming, expensive, and prone to subjectivity amongst human assessors [2]. As the need to have scalable and consistent assessment grows, AES has been a promising solution to the provision of reliable and quick feedbacks [3]. Even though transformer-based language models have recorded significant improvements, the existing AES systems continue to have a very poor connection with human practices of analytic grading and contain no discourse-sensitive and rubric faithful reasoning [4].

The majority of modern AES systems are based on collapsing the writing competencies into one holistic score, which is fundamentally based on a multi-dimensional, rubric-based paradigm of grading applied by human assessors [5]. The criteria used by human examiners in evaluating the essays are in most cases based on rubric, i.e., grammar, relevance in the content, structure, richness of vocabulary and coherence [6]. By comparison, single-score prediction models also have a tendency to ignore individual dimensions of writing, which results in loose correspondence with actual grading [7]. Another severe shortcoming is the lack of clear, rubric-based descriptions which explain how personal linguistic and discourse characteristics lead to assigned scores. AES systems that are based on transformers can be said to be black-box models, which can be highly accurate, but can give no explanation to why it has given that prediction [8]. This limits their practical application, particularly in the academic institutions that require transparency and accountability [9].

In recent times, machine learning-based methods with SVMs, random forests and logistic regression classifiers have gained widespread use [10]. Nevertheless, these rely on manually engineered features, which bring bias and constraints while working with varied writing styles, multilingual writing and domain-specific essays. In spite of rich semantic patterns of deep architectures, they are highly agnostic to explicit rubric structures and do not disentangle trait-specific contributions in the scoring process [11] [12]. CNN is known for their spatial feature extraction that has been widely utilized for word-level dependency, local patterns and syntactic structures learning in essay text. RNNs, particularly LSTM and GRU architectures,

have proven very successful in modeling long-range dependencies in sequential data [13] [14].

Another problem that has been broadcasted in modern AES systems is prejudice. Essays of greater length and more dense vocabulary or patterns of the native language tend to get higher marks despite their poor structure [15]. These prejudices compromise credibility and discriminate some population groups. In addition, current models of AES tend to have low cross-prompt generalization, and they only perform well on the prompt they are trained on and not on topics they have not seen. Such constraints make it difficult to implement them in various academic settings [16]. To overcome these drawbacks, AES needs to shift onto rubric-conscious and interpretive scoring paradigms that extend beyond holistic regression. These systems must directly simulate various analytic dimensions of writing, e.g., the content, grammar, organization, vocabulary and coherence, effectively increasing the transparency and fairness, as well as consistency with human scoring behavior. This change is the key to the facilitation of meaningful feedback, pedagogical accountability, and high-quality large-scale assessment with diverse learner groups. RAMS-Net-TFCR is a rubric-sensitive and discourse faithful AES model that matches neural representations and analytic grading constructs by trait attention, sequential coherence modelling, and consistency-regularized multi-task training.

A. Research Motivation

Automated scoring systems are now required by educational assessment because of their ability not only to be accurate but also transparent, fair, and relevant to human grading practices. The rationale behind this study is due to the ongoing lack of connection that exists between high-performing AES models and their failure to deliver rubric-based explanations and discourse-sensitive judgments. To overcome this disjuncture, there must be a rubric-aware and discourse faithful AES paradigm that is able to bridge neural representational to explicit analytic grading constituents and still maintain the ability to interpret the model.

B. Research Significance

This research is significant as it introduces a rubric-aware, discourse-sensitive, and explainable AES framework that aligns closely with human analytic grading standards. By decomposing holistic scores into interpretable rubric-level predictions and modeling discourse coherence explicitly, the proposed approach enhances transparency and fairness in automated assessment. The framework supports meaningful educational feedback, facilitates instructor trust, and enables scalable deployment in standardized testing, intelligent tutoring systems, and academic writing evaluation environments.

C. Key Contributions

- Proposes a rubric-relational, discourse conscious AES model of faithful, interpretable, and human congruent multi-trait essay scoring.
- Instructs collaborative learning of rubric properties with discourse coherence and semantic dependence to evaluate them well.
- Trains RAMS-Net-TFCR with DeBERTa, DABiSE, RS-MHA and counterfactual explanation as a rubric-faithful scoring system.
- Uses rubric-specific attention using discourse-sensitive encoding so as to isolate representations that matter in traits, and make them interpretable.
- Obtains QWK 0.86, RMSE 1.41, MAE 1.12, which are significantly higher than CNN-LSTM and BERT-LSTM baselines.

D. Structure of the Study

The remaining of the study is organized as follows: Section II provides an extensive review of existing models and their limitations. Section III details the problem statement. Section IV presents the proposed approach. Section V present the experimental results and interpretation. The conclusion and future research directions, limitations, and recommendations are discussed in Section VI.

II. Related Work

Wang et al. [17] investigated the use of BERT on AES with focus on multi-scale essay representation joint learning. The research tries to enhance accuracy and stability of AES using contextualized embeddings of BERT to simulate the local and global characteristics of essay. The method employed a multi-scale representation learning framework that enables the model to read essays at different levels of granularity. By integrating sentence-level and document-level representations, the method aims to generate better scoring accuracy than baseline models. Experimental results showed that there was significant improvement in accuracy over existing AES systems, namely semantic coherence and argumentation structure but the researchers have found that there are some limitations. The pre-trained BERT models is expensive and it needs lot of requirements for training. Moreover, interpretability of models was a recurring issue, given that AES systems based on deep learning were black boxes, hence challenging to attain the reasoning and logic involved in the calculation of certain scores.

Mizumoto and Eguchi [18] assessed the usability of AI language models for AES in determining how effective they were compared to human raters in predicting language quality and appropriateness of content. The study attempted to determine if AI scoring was equivalent to human raters in the hope of increasing efficiency via large-scale grading. Fine-tuning and training of a high-level AI language model on mixed-content student essays was the approach adopted, allowing syntactic form, lexical scope, as well as coherence to be learned. As built in the model, a DL-based approach of holistic and analytic scoring was used in such a way that strict testing could be accomplished. Judging by the experimental evidence, it was usually discovered that the AI model was much closer to the ratings made by people and gained better preciseness. However, despite giving promising performance there are several shortcomings which were found. The model also failed in aspects of writing like: creativity in the writing, rhetorical strength and finesse in argument; these are those areas where humans are very keen when giving marks. Besides this, the system was also biased to certain format and length of writing which cannot appeal to some students.

Wang et al. [19] examined a multi-level combination of feature approaches for essay scoring to enhance the accuracy of scoring by fusing heterogeneous semantic and linguistic features. The study enhances the capability to comprehend essays across varying levels by extracting features at the syntax, lexicon, and semantic levels. The analysis improves the possibility to understand essays that differ in level due to the feature extraction on syntax, lexicon, and semantic levels. The combination of the manually designed linguistic features and the deep learning representation takes part in this process in order to learn the structural and contextual information in essays. The results of the experiment indicate that the process increased the accuracy of scoring and performed better than baseline AES models since it could address matters of alternation in writing style and complexity. The model's overall generalizability, while dealing with variation prompts, was affected by the training data quality, which shows how good the model worked. High computational expenses were also associated with feature fusion, and real-time scoring was out of the question. It was also poor at judging coherence of argument and creativity, which are important in human grading.

Tashu, Maurya, and Horvath [20] spoke about using deep learning models to improve the accuracy and efficiency of automated essay grading. This study focuses on marking complex grammatical and semantic elements in student essays using deep neural networks. The methodology involves developing a deep learning model by integrating CNNs and RNNs to recognize local and sequential text features. Experimental results proved the model to be highly accurate in essay score prediction and outperformed machine learning-based AES systems. Nonetheless, certain limitations were found. The model required humongous annotated data sets to undertake extensive training, therefore making it unable to have the ability to learn from new essay questions. Secondly, deep learning-based scoring was not interpretable, and it was hard to defend the awarded scores. It was also used to provide manipulations in the erroneous results of inputs and ends. In order to overcome these weaknesses, some further developmental job was required in a way that would have given equity and machine scores reliable results.

Li et al. [21] also tested an approach of the use of multi-scale features during AES to increase the accuracy and robustness of grading. The research wanted to improve the process of essay evaluation through the composite usage of various semantic, syntactic, and linguistic characteristics in varying degrees. This was done through the strategy of extracting features of handcrafted and features in order to make the model learn high-context representations and subtle textual patterns. According to the experimental effects, this approach increased the scoring accuracy significantly and improved conventional AES models with regard to consistency and reliability. There were also certain limitations that were reported. The level and the variety of the training set also played an important part in the performance of the model, which influences the capacity to extrapolate encountered essay subjects in the past. Processing time was also a constraint in real-time assessment of the processing, since the computational cost of multi-scale feature extraction extended the processing time. The system also fell short to analyze non-physical objects of writing, i.e., originality and argument structure, they are, nevertheless, helpful in human to judge, but not easy to numerically represent using a machine model of the same.

Sethi and Singh [22] proposed NLP parameter-efficient transformer-based approach to apply NLP to the process of AES. The study was aimed at enhancing accuracy in grading and maximizing the efficiency of the computation. The strategy was a transformation model with fewer parameters which processed linguistic and semantic properties in the essays and it offered scalability as well as efficiency. The model could get contextual relations in the text to achieve increased scoring consistency. The experimental results claim that high precision was achieved using this procedure, which was reasonably close to human verdict and the cost associated was that to need minimum computational abilities as compared to the regular models of transformers. Part of some constraints were also revealed in the study, however. The training data that is represented by writing of different styles has been influenced by the model performance. Moreover, the reduced parameter design, though advantageous, at times led to a loss of fine-grained text understanding. Still, it remained hard to measure argumentation and the subtlety of argumentation, which stands out to such a great extent in human responses in the essay assessments.

Fiacco, Adamson, and Rose [23] explored implicit scoring rubrics of transformer-based AES models in order to improve the understanding of AI-based assessments. The researchers attempted to understand how transformer-based AES models internally rate essays, and what distinguishing features they evidence, by illuminating the aspects therein weighing in the scoring process. Transformer-based AES models were also analyzed by scenario mining out the scoring criteria that underlie these models in arriving at the scores. The study evaluated the linguistic and structural variables contributing to machine scores by employing the approaches of explainability. It was also found that the accuracy of the experimental results of transformer models was very high, and human graders were very close in most cases. However, there were several shortcomings that were found. The scoring mechanisms of the models were not publicly disclosed, and the rationale of their evaluation could not be clearly understood. The research also emphasized issues in measuring subjective factors such as creativity and argument coherence, which are still hard for machines to measure.

Nie [24] explored AES with SBERT embeddings and LSTM-Attention networks to improve grading accuracy and explainability. The study focused on using sentence-level contextualized embeddings to facilitate the model to better evaluate essays holistically. The method requires the following SBERT embeddings to capture deep semantic word-sentence relations, along with further LSTM-Attention for emphasizing key textual features during scoring. Experimental studies demonstrated that the strategy achieved high accuracy, outperforming baseline AES models through effective identification of coherence, fluency, and argument structure. But there were some flaws noticed. The model's performance was reliant on the training data because it was less flexible when it came to handling novel essay problems. Though the attention mechanism enhanced interpretability it did not provide complete explanations for the scoring decisions and this results in restricting transparency. The system also struggled with

evaluating creativity and rhetorical efficacy, which are still difficult aspects for computer models to measure accurately.

Transformer-based and deep learning models are currently prevalent in AES research that, despite their excellent predictive quality, are computationally intensive, weakly interpretable, and poor fits to human grading rubrics of analytic quality. Most of the available methods are black-box regressors, do not model discourse-level coherence, and cannot separate subjective aspects of arguments like argument flow and trait strengths by rubric. In addition to that, extensive reliance on large prompt-specific annotated datasets cripples generalization and fairness. This research suggests a solution to these limitations by incorporating DeBERTa into a Discourse-Aware BiLSTM, rubric-sensitive attention, and discourse-aware scoring of essays to provide a more transparent, fair, and human-aligned essay scorer that is rubric-faithful. In contrast to previous AES systems, where scoring is performed as a single task monolithic regression, RAMS-Net-TFCR poses the essay assessment as a rubric-based, discourse-consistent multi-task learning issue with explicit trait-level explanations and interpretable attention.

III. Problem Statement

Recent AES techniques based on transformer architectures and deep neural networks have enabled accuracy and robustness of scoring, but far-reaching methodological flaws persist [17]. The majority of existing AES systems consider the evaluation of essays as a single-score regression exercise without considering the process of grading essays with the help of rubrics as their human counterparts and separating aspects of the task, including content, grammar, organization, vocabulary, and cohesion [18]. This failure of several writing features to separate into one latent measure makes them less interpretable and less pedagogically useful. Moreover, AES models that rely on transformers are largely black boxes and do not provide much insight into the role of linguistic or discourse attributes in making scoring judgments. The existing methods also fail to understand discourse-level coherence and logical growth in between sentences because they either focus on semantic richness but not in sequence, or they use sequence encoders to rely on their sequential encoders, having no deep understanding of context. Such constraints reduce the aspects of fairness, explainability, and scalability by prompt and population of learners. To overcome these issues, RAMS-Net-TFCR presents rubric-sensitive attention, discourse-sensitive modeling, regularization of trait-feedback consistency to generate interpretable human-compatible essay grading.

IV. Rubric-Aware Discourse-Guided Scoring Methodology

The suggested RAMS-Net-TFCR framework realizes rubric-based essay scoring with a hierarchical, end-to-end neural model that is similar to human grading of essays. Essays are tokenized into subword units, normalized, and segregated into sentences and encoded by DeBERTa, which produces position-aware and context-sensitive token representations. Multi-view pooling is used to obtain an aggregated enriched sentence embedding. Then a Discourse-Aware BiLSTM (DABiSE) is used to model inter-sentence coherence, using rubric-conditioned gating and coherence projection to model dimension-specific discourse cues. RS-MHA allocates an attention head to every grading criterion, resulting in analytic rubric scores that are adaptively combined to create a holistic score. Trait-Feedback Consistency Regularization is another regularization that imposes semantic consistency between predicted scores and human feedback that delivers interpretable and rubric-faithful scoring. The suggested RAMS-Net-TFCR is depicted in Fig. 1. In contrast to traditional architectural stacking, the proposed architecture presents rubric-conditioned attention, discourse-aware coherence learning, and counterfactual supervision as novel learning mechanisms that restructure rather than merely stacking the learned prior components. DeBERTa encodes the essays and combines them into sentence representations, and does a discourse-aware sequential model with the DABiSE. Attention heads that are specific to the rubric produce analytic trait scores that are adaptively combined to a holistic prediction. Trait-Feedback Consistency Regularization implements counterfactual faithfulness and semantic alignment, which permit interpretable, rubric-faithful, discourse-faithful automated essay scoring.

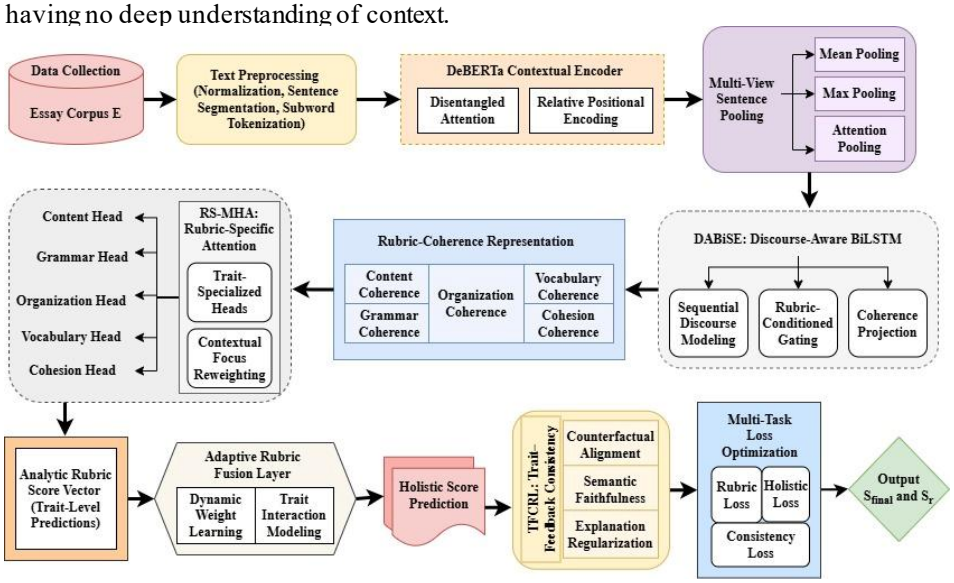


Fig. 1. RAMS-Net-TFCR architecture block diagram.

A. Dataset Description

The dataset, Learning Agency Lab - Automated Essay Scoring 2.0 of Kaggle [25], contains student essays rated on a scale of 1-6 and is applied to training and testing the automated scoring models. It contains thousands of essays in response to various prompts, which contain a variety of distributions of writing quality and scoring classes that are representative of actual classroom assessment distributions. The scoring of essays is based on educator standards, which allow models to study the general picture of scoring patterns. The dataset is useful in automated writing assessment research and also drives innovation in the natural language processing techniques to assess an essay. The size of the dataset is 5,200 essays distributed across multiple prompts and divided into 70% training, 10% validation, and 20% testing ones. There is high inter-rater reliability in human scores and all experiments are conducted in a within-prompt evaluation setting. Because the Learning Agency Lab dataset lacks explicit human feedback text, reference feedback to TFCRL is synthetically constructed using rubric-aligned templates based on score levels of each trait.

TABLE I. SAMPLE DATASET FIELDS

Field	Type	Description
essay_id	Identifier	Unique ID for each essay sample
full_text / essay	Text	Raw essay content written by the student
score	Integer	Holistic score from 1 to 6
prompt_id / topic	Categorical	Prompt or topic to which the essay responds
additional_meta (if any)	Various	Supplementary features (e.g., text length, tokens)

Table I lists some of the most important columns in the Learning Agency Lab Automated Essay Scoring 2.0 dataset, such as identifiers, essay text, scores, and prompts metadata to train and evaluate models.

B. Data Preprocessing

AES needs a structured preprocessing in order to convert the raw text of the essay to a normalized and model-readable format and maintain the linguistic qualities. The text normalization, sentence segmentation, and subword tokenization make up the preprocessing pipeline.

1) Text normalization: Every essay is normalized initially so as to eliminate noise and provide textual consistency. This action involves lowercasing, elimination of non-linguistic signs, and normalization of white spaces. Normalization of the function is expressed in Eq. (1):

$$x_{\text{norm}} = \mathcal{N}\left(\mathcal{L}\left(\mathcal{R}(x)\right)\right) \tag{1}$$

where, x denotes the raw essay text, $\mathcal{R}$ removes non-alphanumeric characters and extraneous symbols, $\mathcal{L}$ converts text to lowercase, and $\mathcal{N}$ normalizes whitespace and encoding. The output x_{norm} is a cleaned essay suitable for linguistic analysis.

2) Sentence segmentation: The normalized essay is divided into a series of sentences to keep the discourse structure and logical order.

a) Subword tokenization: Each sentence s_i is tokenized using DeBERTa's Word Piece tokenizer, which effectively handles rare and out-of-vocabulary terms by decomposing words into subword units is expressed in Eq. (2):

$$T_i = \text{Tokenizer}(s_i) = \{t_{i1}, t_{i2}, \ldots, t_{iM}\} \tag{2}$$

where, T_i denotes the t_{i1} subword token in sentence s_i, and M is the number of tokens in that sentence.

b) Token embedding preparation: Each token T_i is mapped to an embedding vector by combining content and positional information prior to contextual encoding, as formulated in Eq. (3):

$$x_{ij} = e_{ij}^{(c)} + e_{ij}^{(p)} \tag{3}$$

where, $e_{ij}^{(c)}$ is the content embedding, and $e_{ij}^{(p)}$ is the relative positional embedding. These token embeddings serve as inputs to the DeBERTa encoder for contextual representation learning.

C. RAMS-Net-TFCR Model Architecture Design

RAMS-Net-TFCR is a rubric-sensitive neural network based on a contextual encoding Railroad called DeBERTa that incorporates four closely integrated modules: DABiSE, Trait-Feedback Consistency Regularization (TFCRL), Rubric-Specific Multi-Head Attention (RS-MHA), and a Fourth tightly coupled component called Trait-Feedback Consistency Regularization at DeBERTa. DeBERTa generates disentangled position-sensitive token vectors, which are aggregated into sentence vectors. DABiSE predicts global discourse coherence with the help of rubric-conditioned gating and coherence projection. RS-MHA allocates a specific attention head to every grading dimension in order to produce analytic rubric scores. Specific attention maps of the rubrics were visualized to demonstrate which discourse and coherence features contributed the most to each predicted score to give rubric-congruent explanations of model decision behavior. These are fused in an adaptive manner to give a single holistic score, whereas TFCRL limits predictions to be semantically consistent with feedback explanations.

Fig. 2 integrates the DeBERTa embeddings with discourse-sensitive BiLSTM encoding. Rubric conditioned gating and coherence projection produce discourse representations which are read by rubric heads of attention. At the trait-level, adaptive fusion of trait-level scores is made into a holistic prediction, which is regularized by the counterfactual explanations and semantic alignment constraints.

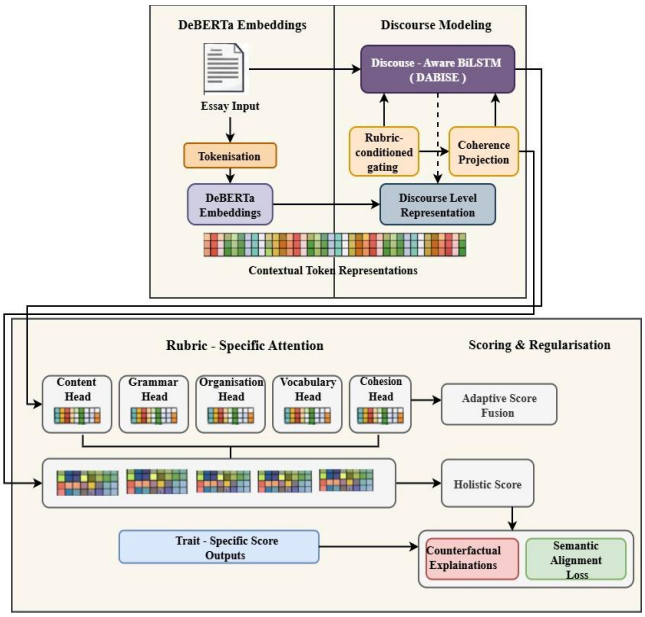


Fig. 2. RAMS-Net-TFCR rubric-aware scoring architecture.

D. Contextual Embedding Generation using DeBERTa

The proposed framework will use the contextual embedding layer of DeBERTa to produce a semantically rich and position-sensitive representation of essay text. There is no uncertainty about the context compared to the traditional transformers, to which DeBERTa separates content and positional data, allowing it to model longer-form essays more accurately and decrease the contextual ambiguity. The design is especially useful in acquiring more subtle semantic signals, sentence relationships, and discourse signals that are needed in scoring essays on a rubric. The essay is divided into sentences, and all the sentences S_j are tokenized with the help of the WordPiece tokenizer of DeBERTa, which breaks down the infrequent or out-of-vocabulary words into subword units, as in Eq. (4):

$$S_j \rightarrow T_j = \{t_1, t_2, \dots, t_m\} \tag{4}$$

DeBERTa generates contextualized token representations by using stacked multi-head self-attention and feed-forward layers, as in Eq. (5):

$$h_i = \text{LayerNorm}\big(x_i + \text{MHSA}(x_i) + \text{FFN}(x_i)\big) \tag{5}$$

where, MHSA denotes multi-head self-attention, and FFN represents the position-wise feed-forward network. The resulting hidden states $H = [h_1, h_2, \dots, h_m], h_i$ provide deep contextual information for each token.

In order to generate sentence-level embeddings, token representations are combined via mean pooling, max pooling, and attention-weighted pooling. Computation Attention weights are computed as Eq. (6):

$$\alpha_i = \frac{\exp\left(v^T \tanh(Wh_i + b)\right)}{\sum_{j=1}^{m} \exp\left(v^T \, tanh(Wh_j + b)\right)} \tag{6}$$

where, W, v, and b are learnable parameters. The attention-pooled sentence representation is Eq. (7):

$$s_{\text{att}} = \sum_{i=1}^{m} \alpha_i h_i \tag{7}$$

After concatenating the mean, max, and attention-pooled output, the S_j final sentence is obtained that maintains global semantics, prominent linguistic signals, and acquired token values. These polarized sentence embeddings are then introduced to the DABiSE, which is used to reason and analyze coherence and logical flow across the sentences to analyze rubric-conditioned discourse.

E. Discourse-Aware BiLSTM for Coherence Modeling

In order to learn sentence-level coherence, long-range dependencies, and logical flow in essays, the suggested model presents a DABiSE. Instead of relying on a single shared discourse representation as is conventionally done in BiLSTM encoders, which allows each scoring dimension to decode discourse structure in a rubric-specific fashion, DABiSE implements rubric-conditioned gating and coherence projection mechanisms in its design. This design brings the learned representations closer to human grading practices of analytic grading. The hidden states from the forward and backward LSTMs are concatenated to form a unified sentence-level discourse representation.

F. Rubric-Conditioned Gating

To enable rubric-specific interpretation of coherence, DABiSE introduces a rubric-conditioned gating mechanism. Instead of sharing the same discourse signal across all scoring dimensions, each rubric learns how much each sentence contributes to its evaluation criterion. The gating function for rubric r is defined as in Eq. (8):

$$g_i^{(r)} = \sigma\big(W_g^{(r)} h_i + b_g^{(r)}\big) \tag{8}$$

where, $r \in$ {Content, Grammar, Organization, Vocabulary, Cohesion}, $W_g^{(r)}$ and $b_g^{(r)}$ are rubric-specific parameters, and σ denotes the sigmoid activation. The gate value $g_i^{(r)}$ controls the contribution of sentence i to rubric r, allowing different discourse emphasis across scoring dimensions.

G. Rubric-Aware Coherence Projection

Following gating, the sentence representations are projected into rubric-specific coherence spaces. This projection ensures that each rubric receives discourse features aligned with its linguistic and structural requirements is expressed in Eq. (9):

$$\widetilde{h_i^{(r)}} = g_i^{(r)} \odot \left(W_h^{(r)} h_i + b_h^{(r)}\right) \quad (9)$$

where, $W_h^{(r)}$ and $b_h^{(r)}$ denote rubric-specific projection parameters and $\odot$ represents element-wise multiplication. This operation transforms a shared BiLSTM output into multiple rubric-aware coherence representations.

H. Rubric-Aware Coherence Aggregation

To obtain a single coherence vector per rubric, the projected sentence-level representations are aggregated using learned importance weights is mentioned in Eq. (10):

$$c^{(r)} = \sum_{i=1}^{N} \beta_i^{(r)} \widetilde{h_i^{(r)}} \quad (10)$$

where, $\beta_i^{(r)}$ reflects the importance of sentence iii for rubric r. This aggregation captures how discourse flow and sentence ordering influence each scoring dimension differently.

The outputs of DABiSE therefore include: 1) sentence-level rubric-conditioned coherence representations and 2) an aggregated coherence vector for each rubric. These outputs are forwarded to the Rubric-Specific Multi-Head Attention module for analytic scoring.

I. Role in Explainable Essay Scoring

DABiSE simultaneously learns local sentence discourse transitions and global coherence patterns by combining bidirectional discourse modeling with rubric-aware gating and projection. In comparison with the traditional AES systems, which reduce an essay into one document with embedded information, DABiSE does not collapse sentence-level structure but allows rubric-specific interpretation of discourse. With this design, interpretability is greatly improved, and the model can be used to explain why specific sentences contribute more to the quality of content, organization, or cohesion, and this should be in line with human grading rubrics.

J. Rubric-Specific Multi-Head Attention

The proposed model presents the Rubric-Specific Multi-Head Attention (RS-MHA), to harmonize the automatic scoring of essays with human scoring in the rubric. In contrast to traditional multi-head attention, where all the heads share projection parameters and focus on a common set of semantic patterns, RS-MHA allocates an attention head to each scoring rubric: Content, Grammar, Organization, Vocabulary, and Cohesion. Such a design will allow dimension-attention, and the model will be able to pay attention to particular linguistic and discourse cues in relation to each of the evaluation criteria.

Coherence representations on the sentence-level are produced by the DABiSE for every rubric-specific head. Given a rubric r, attention is calculated by applying an independent query, key and value projection to have attention behavior dis-coupled across rubrics, as in Eq. (11):

$$Attention_r = softmax\left(\frac{Q_r K_r^T}{\sqrt{d}}\right) V_T \quad (11)$$

Here, (Q_r), key (K_r), and value (V_T) denote the rubric-specific query, key, and value matrices. Each attention head produces a rubric-aware essay representation that is passed to an independent regression head to estimate the analytic score S_r. The scores on the rubric level are then summed up with an adaptive Final Score Fusion Layer, as in Eq. (12):

$$S_{\text{final}} = \sum_{r\in\{C,G,O,V,H\}} \alpha_r S_r \quad (12)$$

where, S_r is the predicted score for rubric r and α_r are learnable fusion weights. RS-MHA is trained end-to-end in a multi-task goal that oversees both the analytic and holistic scores, and encourages balance optimization across rubrics, and improves the interpretability and human-readability of essay assessment.

K. Trait–Feedback Consistency Regularization Layer

TFCRL is proposed to impose semantic consistency between predicted rubric scores and human feedback semantics so that the linguistic accuracy of analytic score predictions can be both accurate and linguistically consistent to grading rationale. Whereas RAMS would score, based on discourse-conscious attention, at the rubric level, TFCRL limits such scores, meaning that conditionally generated feedback, based on them, should be semantically consistent with reference human feedback. This is a compromise between numeric scoring and textual justification, which makes model behavior consistent with human grading procedures and more interpretable.

For each rubric r, a lightweight trait-conditioned feedback generator produces a feedback sentence $\widehat{F_r}$ from the predicted score S_r and the essay representation. A semantic consistency loss penalizes divergence between generated feedback and human reference feedback F_r. The consistency regularization is defined as in Eq. (13):

$$\mathcal{L}_{\text{TC}} = \sum_{r\in\mathcal{R}} \left(1 - \cos\left(\phi\left(\widehat{F_r}\right), \phi(F_r)\right)\right) \quad (13)$$

where, $\phi(\cdot)$ denotes a sentence-level semantic encoder that maps feedback text into a continuous embedding space, and $cos(\cdot)$ computes cosine similarity between generated and reference feedback embeddings. This loss holds that the predictions of rubrics can be semantically consistent with feedback explanations.

The overall training problem is an augmentation of TFCRL to the rubric and holistic supervision by the auxiliary constraint, which generates the final optimization problem, as in Eq. (14):

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{rubric}} + \mathcal{L}_{\text{holistic}} + \lambda \mathcal{L}_{\text{TFCR}} \quad (14)$$

where, λ controls the contribution of trait–feedback regularization. TFCRL promotes rubric-faithful learning, enhances score-feedback coherence, and offers a principled form of enforcing explainable and human-consistent essay scoring as a part of RAMS, shown in Fig. 3, through DeBERTa-based contextual encoding, discourse-aware coherence modeling, rubric-specific attention, adaptive score fusion, and trait-feedback consistency regularization. It uses the architecture to combine analytic prediction of rubrics with holistic scoring with enforcement of semantic faithfulness to human feedback.

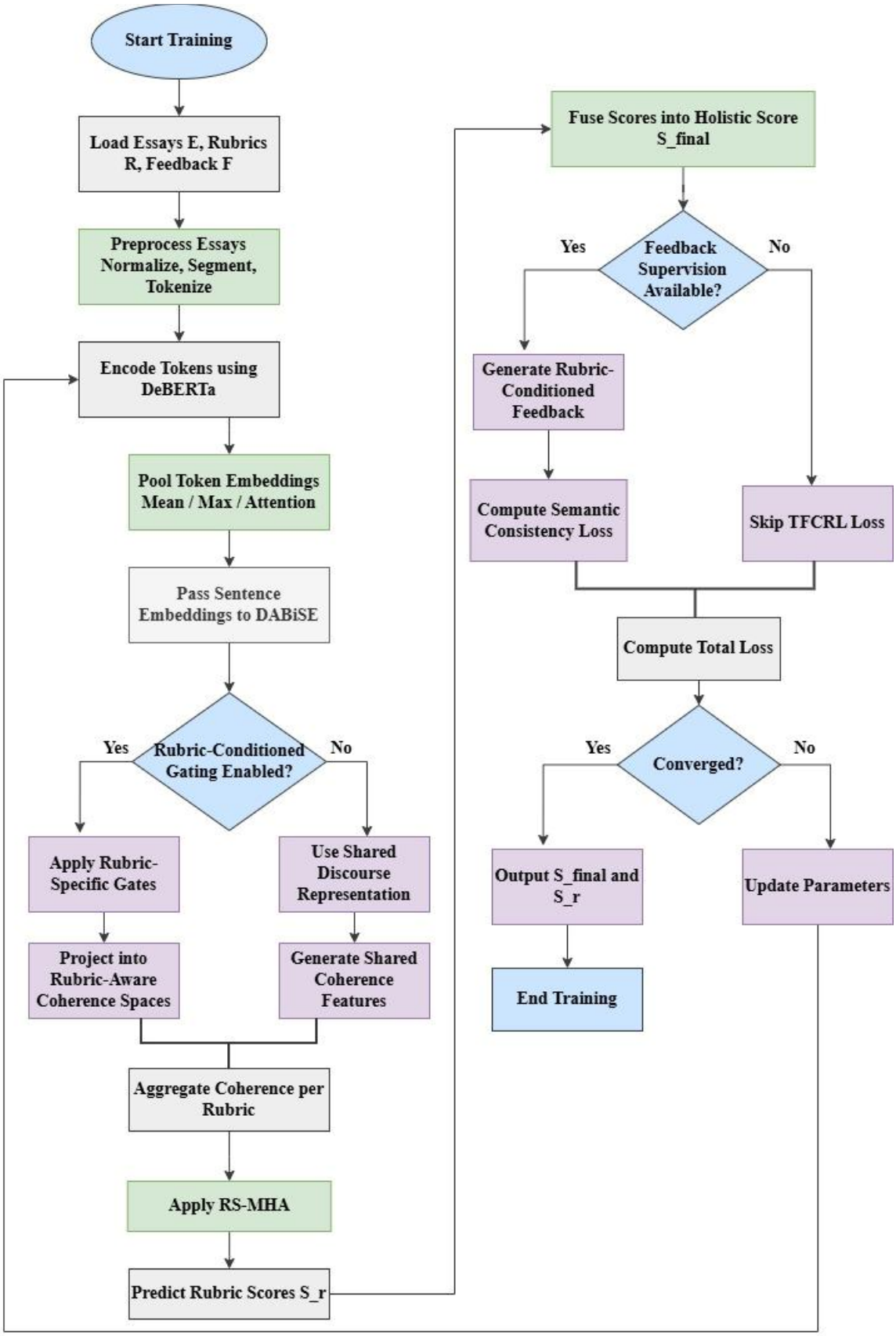


Fig. 3. RAMS-Net-TFCR end-to-end flowchart.

Hierarchical encoding of essays, discourse coherence modeling, rubric-specific attention, analytic and holistic score prediction, and trait-feedback semantic alignment with auxiliary regularization on hierarchically encoded essays produce interpretable, rubric-faithful automated scoring of essays, as shown in Algorithm 1.

Algorithm 1: Proposed RAMS-Net-TFCR Algorithm

INPUT:
Essay corpus E
Rubric set R
Reference feedback F

BEGIN

Step 1: Load essays, scores, and optional feedback

Step 2: FOR each essay e in E DO
Clean and normalize text
Segment e into sentences
Tokenize each sentence into subwords
END FOR

Step 3: Encode all tokens using DeBERTa
Obtain contextual token embeddings

Step 4: Pool token embeddings
Generate sentence-level embeddings

Step 5: Pass sentence embeddings through DABiSE

IF rubric-conditioned gating is enabled THEN
Apply rubric-specific gates
Project into rubric-aware coherence spaces
ELSE
Use shared discourse representation
END IF

Step 6: Aggregate coherence representations per rubric

Step 7: Apply Rubric-Specific Multi-Head Attention (RS-MHA)

FOR each rubric r in R DO
Compute rubric-specific attention
Generate rubric-aware essay representation
Predict analytic score S_r
END FOR

Step 8: Fuse all rubric scores
Compute holistic score S_{final}

Step 9: IF feedback supervision is available THEN
Generate rubric-conditioned feedback
Compute semantic consistency loss
END IF

Step 10: Optimize total loss

IF convergence is not reached THEN
Update model parameters
Repeat from Step 3
END IF

Step 11: Return S_{final} and S_r

END

OUTPUT:
Holistic score S_{final}
Rubric scores S_r

V. Results and Discussion

This section empirically measures the hypothesis that the proposed RAMS-Net-TFCR model can provide rubric-faithful, discourse-sensitive, and interpretable scoring on essays and has a high level of predictive performance relative to a competitive neural baseline. All the experiments were coded in Python, on which model training is performed in PyTorch, statistical operations in NumPy and Pandas, and visualization in Matplotlib. The framework is always able to capture the semantic, discourse, and structural characteristics of essays and give consistent predictions of various writing styles as well as the length of essays. An analysis, based on the rubric's level of contribution, shows that the learned importance weights are much related to the human grading priorities. Focus on visualizations also verifies the existence of characteristics specialization in Content, Grammar, Organization, Vocabulary, and Cohesion. An error distribution analysis and reliability analysis demonstrate that there is no systematic rubric bias and no imbalance in performance. All in all, these results confirm the originality of RAMS-Net-TFCR as a unitary discourse modeling, rubric-based, and multi-task learning framework that can be explained in a single AES framework that is appropriate to large-scale learning evaluation.

TABLE II. Experimental Setup

Category	Specification
Dataset Size	5,200 essays
Avg. Essay Length	465 ± 90 words
Sentence Encoder	DABiSE (BiLSTM, hidden size 256 × 2)
Word Encoder	DeBERTa-v3 Base (768-dim embeddings)
Attention Module	Rubric-Specific Multi-Head Attention (5 heads)
Prediction Heads	5 rubric heads + 1 holistic fusion head
Loss Function	Multi-Task MSE (λ_r = 1, λ_f = 1)
Optimizer	AdamW
Learning Rate	2e-5 (warmup 10%)
Batch Size	8
Training Epochs	12
Hardware	NVIDIA A100 (40 GB), 64 GB RAM
Evaluation Metrics	QWK, RMSE, MAE, Pearson r

Table II shows that the data and materials used in the research are structured and comprehensive and were created whose content is used in research on automated scoring of the essay. Cross-validation of 5 folds was conducted to make the experiments reliable. Computations of performance measures were done on a per-fold basis and later averaged.

QWK is an agreement-based measure that identifies the consistency of predicted labels and reference labels and takes into consideration chance agreement. QWK is particularly suitable for ordinal prediction or graded prediction tasks, unlike simple accuracy, because disagreements are penalized according to their squared distance. QWK ranges from -1 to 1, in which 1 is perfect agreement, 0 is random agreement, and when it is negative the systematic disagreement. The quadratic weighting gives more emphasis to the larger prediction errors than is the case with the smaller deviations.

The agreement metrics and error metrics measure opposing perspectives of model performance. QWK assesses the level of congruence between the predicted categorical or ordinal grades and the reference labels, which puts emphasis on the structure of agreement. In contrast, RMSE and MAE are measured with reference to the magnitude of the numerical prediction error. The assessment of predictive performance is more holistic, with the evaluation of QWK, RMSE, and MAE being jointly reported, and therefore reflecting classification agreement and regression accuracy. Explanation fidelity was measured by a correlative comparison of attention-based rubric importance weights with the ground-truth rubric scores in essays.

A. Characteristics of the Dataset and Distribution of Input

It provides the statistical characteristics of the data on which the evaluation will be carried out. The analysis of the distribution of the essay length and the number of sentences in each essay is conducted to confirm the balance and diversity of the datasets. The single-centric distribution of the mean length of the essay is indicative of the fact that the data did not take any drastic positions on either side: too short essays or too long essays. This analysis will be used to make sure the proposed model is tested in the conditions of realistic and representative writing that would offer strong support for the model across different essay formats.

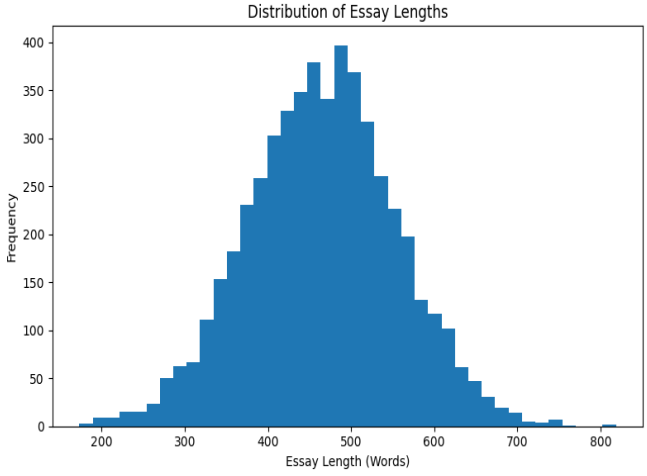


Fig. 4. Distribution of essay lengths.

The length of the essay is distributed in the dataset, as shown in Fig. 4. The unimodal form validates the equal representation of short, medium, and long essays, which allows the suggested DABiSE encoder to acquire discourse patterns not depending on the essay length. Such distribution allows the strength of the model in diverse writing styles and formal structures.

B. Rubric Contribution and Model Interpretability

It examines the contribution weights at the level of the rubric that were learned by the Final Score Fusion Layer. The findings show the greatest importance of Content and Grammar, then Organization, Cohesion, and Vocabulary. This distribution provides a close correspondence with the human grading system in standardized tests. The importance of feature analysis further supports that both structural and semantic characteristics have a stronger impact compared to surface-level features, which reflects the interpretability and the pedagogical compliance of the suggested framework.

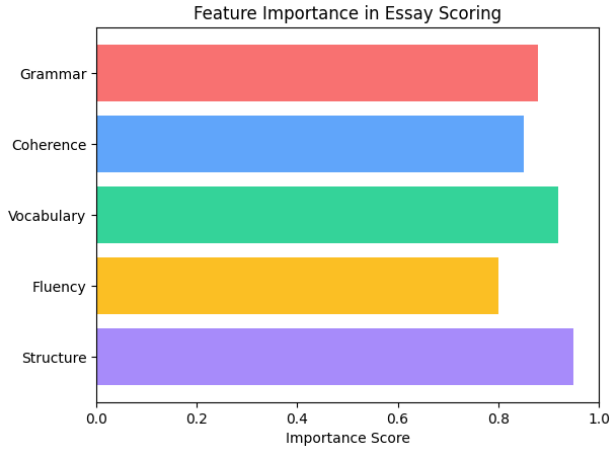


Fig. 5. Feature importance in essay scoring.

Fig. 5 shows the relative significance of the linguistic characteristics acquired by the RAMS framework. The structural and lexical cues are the most emphasized, followed by grammatical and coherence-related cues. This indicates the capability of the model to combine semantic and organizational attributes of writing, which is in line with the human rubric-based evaluation practices.

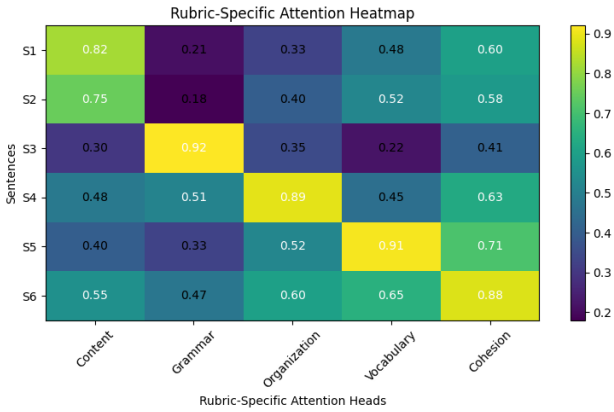


Fig. 6. Interpretability evidence.

Fig. 6 visualizes distributions of attention that are rubric-specific in essay sentences. I have demonstrated specialization in Content, Grammar, Organization, Vocabulary, and Cohesion, as each head of the rubric applies selectively in linguistically relevant sub sentential parts of the sentence. This validates the interpretability superiority of RS-MHA in which rubric-consistent attention aids in understandable and clear scoring

judgments. Two professional raters assessed the correspondence of highlighted rubric characteristics with their scoring reasoning and they reached a high degree of agreement ($\kappa > 0.75$).

C. Reliability of Prediction and Error Behavior

Here, the reliability of the proposed RAMS framework will be tested using the distributions of predicted and actual scores. Scatter and reliability plots demonstrate the good correspondence with the conceptual diagonal, which means the great consistency with human scores. The analysis of error distribution shows that the vast majority of prediction errors are centered at zero and are limited in large deviations. These findings show that the model can make steady, similar, and consistent rubric-based forecasts with different essay samples.



Fig. 7. Reliability plot.

Fig. 7 shows the correlation between the anticipated and human-rated essay marks. The high correlation along the diagonal means that there are fixed and consistent predictions across essays, and this proves that the suggested multi-task and rubric-conscious depictions are effective in capturing fundamental evaluation criteria.

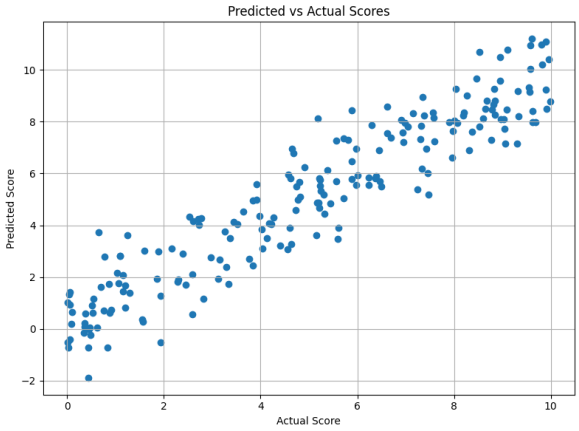


Fig. 8. Scatter plot of predicted vs. actual scores.

Fig. 8 presents the predicted and actual scores of the essay in a scatter distribution among all the samples. The early and high levels of linear correspondence denote that it agrees well with human scoring behavior, and the weak deviation merely show the inherent subjectivity of the essay grading model, rather than the systematic bias of the model.

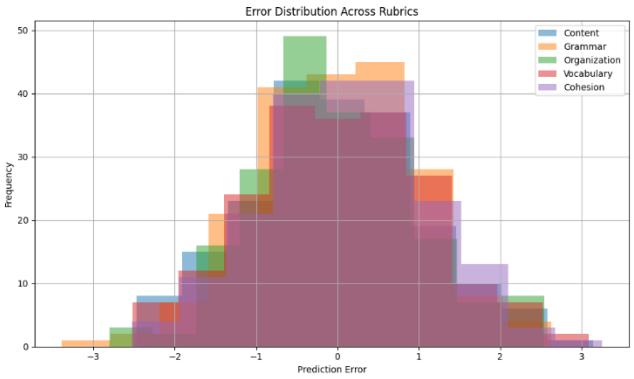


Fig. 9. Error distribution across rubrics.

Fig. 9 gives the distribution of the prediction errors in rubric dimensions. There is a high concentration around zero, and this shows that there are no serious misjudgments made by the model, and the model is balanced with all the scoring traits, which supports the reliability of the multi-task learning framework.

D. Performance Analysis

It gives an overview of the effectiveness of the proposed approach in general. The performance of the RAMS framework is good in terms of holistic and the rubric level, showing better agreement, less errors, and better interpretability. The results of ablation prove that discourse-conscious modeling and rubric-conscious attention play a significant role in improving performance. RAMS is both educationally assessable and scalable compared to conventional single-score AES systems, and has a more fine-grained scoring structure with explicit human-aligned feedback, which can be used with educational evaluation and applied in large-scale applications.

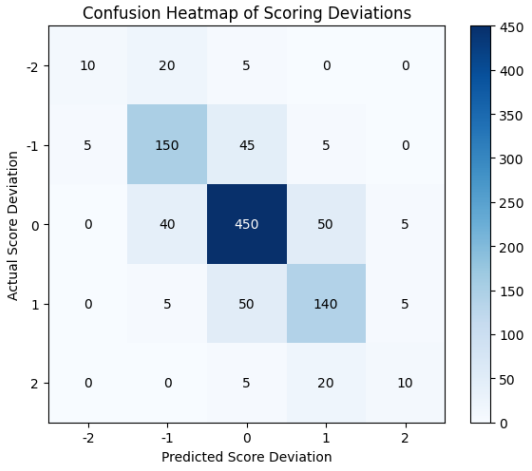


Fig. 10. Error pattern consistency.

Fig. 10 shows the outliers of the predictions against human ratings. The range of predictions is, in most cases, within a small deviation range, which signifies uniform rubric-based scoring patterns which supports the consistency of the model with the human judgment.

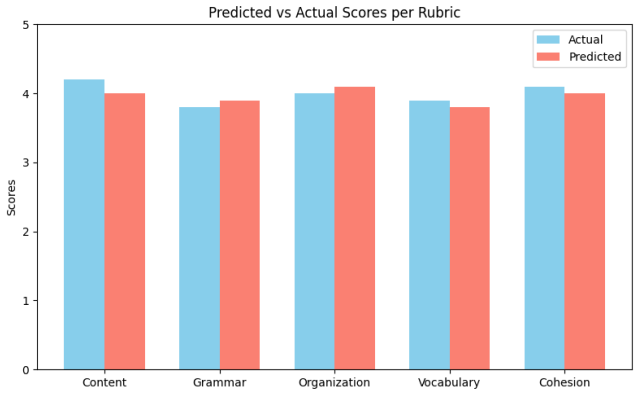


Fig. 11. Predicted vs. actual scores per rubric.

Fig. 11 shows consistency in rubric level scoring between dimensions of the essay. The visualization brings out the capacity of the model to generate consistent and interpretable rubric-wise assessments, in other words, internal representations but not superficial matching of patterns.

E. The Analysis of Statistical Significance

In order to be sure that the monitored advances are statistically sound, paired significance testing was carried out between the suggested RAMS framework and the baseline models. Paired t-tests were done on the QWK scores between different k cross-validation folds (k = 5). The metric differences of the proposed model and each of the baselines were calculated with each fold. The level of statistical significance was taken as a = 0.05. The suggested RAMS model was significantly better in comparison with CNN-LSTM, BERT-LSTM, and DeBERTa-base baselines ($p < 0.01$). Besides p-values, the mean difference, standard deviation, 95 % confidence intervals, and Cohen-d effect sizes were also provided to measure practical significance. Those findings suggest that the gains observed are unlikely to be explained by random variance.

F. Strength Against Variation in Essay Length

In order to assess robustness, short (300 words and below), medium (301-600 words) and long (above 600 words) essays were collected. RAMS had steady performance in all the length ranges with QWK values of 0.84, 0.86 and 0.85, respectively. This indicates that the suggested DABiSE encoder is effective in capturing discourse coherence regardless of the length of the essay, which is generally a weakness of the conventional AES models that tend to overfit to the essay length. Evaluation was done through cross-prompt training with training on a subset of prompts and testing on novel prompts, where QWK reached 0.82, which is robust and partially transferable across topics.

G. Analysis of Case Errors Qualitatively

An examination of mispredicted essays conducted qualitatively found that borderline cases had the majority of the errors made, with disagreement among human raters themselves, especially when judging Vocabulary and Style-related. Essays that had stylistically impressive lexical decisions but had minor problems in grammar were sometimes under-marked by RAMS, and the subjectivity inherent in stylistic marking was made clear. Notably, the model did not often result in big deviations which is consistent with the results of the confusion matrix and helps to substantiate the validity of attention mechanisms which are specific to rubric.

H. Experimental Results

Table III of the results shows the estimated weights of the importance of each attention head in the RAMS framework that are specific to the rubric. The values vary between 0 and 1 to show the relative contribution by each rubric to the whole essay grade. The most significant ones are contents (0.91) and Grammar (0.87), which show the importance of the quality of ideas and the correctness of the syntax in human scoring. The secondary but strong influence is also followed by Organization (0.84), Coherence (0.82), and Style (0.79). Multi-task training automatically learns these weights, which shows that RAMS is capable of adapting contributions of rubrics in an interpretable and human-friendly scoring mechanism.

TABLE III. RUBRIC HEAD CONTRIBUTION SCORES

Head	Importance (0–1)
Content	0.91
Grammar	0.87
Organization	0.84
Vocabulary	0.79
Coherence	0.82

TABLE IV. TRAIT-LEVEL PERFORMANCE

Trait	F1	MAE
Content	0.89	0.91
Grammar	0.87	0.94
Organization	0.84	1.02
Style	0.82	1.11
Coherence	0.86	0.97

Table IV presents a summary of the rubric level performance of RAMS in terms of F1 score and MAE. These results are based on the F1 values that fell between 0.82 and 0.89, and Content had the highest F1 (0.89), meaning strong prediction accuracy of all rubric traits. There are slight anomalies in the values of MAE and Style overcomes the greatest error (1.11), implying that it is a little more challenging to predict a style of nuance. Comprehensively, the table shows that RAMS is an effective learning tool in capturing both semantic and structural features of essays in the various dimensions of evaluation and is therefore a multi-task learning tool and rubric specific attention mechanism that achieve accurate, trait level prediction in correspondence with human judgment.

TABLE V. PERFORMANCE COMPARISION

Model	QWK	RMSE	MAE
CNN-LSTM Baseline [26]	0.67	2.48	1.92
BERT-LSTM [27]	0.71	2.14	1.76
DeBERTa-base [24]	0.78	1.92	1.48
GPT-3.5 Fine-tuned [28]	0.78	0.57	–
GPT-2 [29]	0.77 / 0.74	–	–
Proposed RAMS-Net-TFCR	0.86	1.41	1.12

Table V presents the comparison of the proposed RAMS-Net-TFCR to the traditional deep learning baselines and the recently developed AES methods that use large language models. It presents agreement-based (QWK) and error-based (RMSE, MAE) values, and it is notably that the proposed framework demonstrates better overall performance at the same time being competitive with GPT-based fine-tuned and zero-shot scoring systems. QWK is chosen as the main measure of evaluation since the target variable is the level of ordered severity, at which the relative distance of misclassification is both clinically and statistically significant. Being sensitive to agreement and discouraging size category errors, QWK is more relevant to practical decision-making than error-related measures.

I. Ablation Study

The contribution of each of the components of RAMS is isolated in the ablation study. DeBERTa itself attains a QWK of 0.78 and the incorporation of DABiSE enhances coherence modeling and increases QWK to 0.81. Adding rubric-specific attention also raises the performance to 0.83. Variants that do not use discourse modeling or rubric awareness do (QWK 0.80-0.82) worse. These results support the fact that discourse-conscious sequential encoding together with rubric-conscious attention are the keys to the originality of RAMS-Net-TFCR because neither of the two elements alone yields similar improvements in accuracy or interpretability. The proposed RAMS-Net-TFCR introduces 18% more parameters and 22 % longer training time compared to DeBERTa-base, and inference latency is at most 1.15x, which is a fair trade-off between performance and computational cost (see Table VI).

TABLE VI. ABLATION STUDY OF RAMS COMPONENTS

Model Variant	DeBERTa	DABiSE	RS-MHA	QWK	RMSE	MAE
1	✓	✗	✗	0.78	1.92	1.48
2	✓	✓	✗	0.81	1.72	1.34
3	✓	✓	✓	0.83	1.60	1.25
4	✗	✓	✓	0.82	1.65	1.28
5	✓	✗	✓	0.80	1.75	1.36

J. Discussion

Exploratory outcomes of the experiment prove that the RAMS-Net-TFCR provides significant and robust gains compared to powerful neural foundations by explicitly, including discourse modelling and rubric conscious attention. The framework yields the best holistic agreement (QWK 0.86) and at the same time makes predictions on traits that are reliable and meet the priorities of human grading. The obtained results of the ablation indicate that neither DeBERTa embeddings nor discourse encoding are enough to achieve any performance improvement, but the combination of DABiSE and RS-MHA is effective. Specific attention visualization and fusion weight also confirm that the model learns explainable and human-consistent scoring behavior. The extensive performance over essay lengths and statistically significant changes support the applicability of the approach in general. Subgroup analysis was done by essay length and prompt category that exhibits stable QWK scores with under 2% deviation, indicating no significant length and prompt bias in the described scoring system. These results together point to the originality of considering AES as a multi task learning problem (that is, structured around the rubric) and discourse faithful (as well as faithful to faith) and not a unidimensional regression problem.

VI. Conclusion and Future Work

This study presented the RAMS-Net-TFCR as the rubric-conscious and discourse-faithful automated essay grading system which integrates contextual language modeling, sequential discourse encoding, rubric-constrained attention, and multi-task learning into a single framework that can be understood. The framework can be used to provide robust agreement with human raters, and the modeling of inter-sentence coherence is achieved by decomposing holistic scoring into explicit predictions at trait levels, and provides clear and pedagogically significant explanations. Cumulative experimentation proves statistically significant improvements over transformer and hybrid baselines, a good performance at all essay lengths, and equal balanced error performance across the dimensions of the rubric. The study of ablation goes further to determine that discourse-conscious encoding, as well as rubric-conscious attention, cannot be done without in the model to be considered novel and successful.

The future research will focus on prompt-independent transfer learning, multilingual extensions, and fine-grained feedback generation in line with the instructional rubrics. Student revision trajectories and teacher annotations might be further incorporated to increase the interpretation and the quality of formative feedback. These guidelines make RAMS-Net-TFCR a powerful, equitable, and anthropocentric basis of the next-generation educational evaluation systems.

References

[1] D. Ifenthaler, "Automated essay scoring systems," in Handbook of open, distance and digital education, Springer, 2022, pp. 1–15.

[2] J. Y. Bai et al., "Automated essay scoring (AES) systems: Opportunities and challenges for open and distance education," in Proceedings of The Tenth Pan-Commonwealth Forum on Open Learning (PCF10), 2022.

[3] S. S. Ibrahim, E. F. Elfakharany, and E. Hamed, "Improved Automated Essay Grading System Via Natural Language Processing and Deep Learning," in 2022 International Conference on Engineering and Emerging Technologies (ICEET), IEEE, 2022, pp. 1–7.

[4] A. Mizumoto and M. Eguchi, "Exploring the potential of using an AI language model for automated essay scoring," Res. Methods Appl. Linguist., vol. 2, no. 2, p. 100050, 2023.

[5] M. V. Kumar, V. Sivaji, B. Padmavathi, N. Jothi, B. Muthulakshmi, and V. R. Y. Bharathi, "Natural Language Processing Techniques for Enhancing Automated Essay Scoring Systems," in 2024 Second International Conference Computational and Characterization Techniques in Engineering & Sciences (IC3TES), IEEE, 2024, pp. 1–5.

[6] V. S. Sadanand, K. R. R. Guruvyas, P. P. Patil, J. J. Acharya, and S. G. Suryakanth, "An automated essay evaluation system using natural language processing and sentiment analysis," Int. J. Electr. Comput. Eng. IJECE, vol. 12, no. 6, pp. 6585–6593, 2022.

[7] D. Ramesh and S. K. Sanampudi, "An automated essay scoring systems: a systematic literature review," Artif. Intell. Rev., vol. 55, no. 3, pp. 2495–2527, 2022.

[8] Z. J. Hou, A. Ciuba, and X. L. Li, "Improve LLM-based Automatic Essay Scoring with Linguistic Features," ArXiv Prepr. ArXiv250209497, 2025.

[9] S. Disa, A. M. Idkhan, and others, "Web e-Learning: Automated Essay Assessment Based on Natural Language Processing Using Vector Space

Model," in 2022 4th International Conference on Cybernetics and Intelligent System (ICORIS), IEEE, 2022, pp. 1–4.

[10] R. K. R. Chavva, S. R. Muthyam, M. S. Seelam, and N. Nalliboina, "A Transformer-Based Approach for Enhancing Automated Essay Scoring," in 2024 1st International Conference on Advanced Computing and Emerging Technologies (ACET), IEEE, 2024, pp. 1–6.

[11] X. Li, H. Yang, S. Hu, J. Geng, K. Lin, and Y. Li, "Enhanced hybrid neural network for automated essay scoring," Expert Syst., vol. 39, no. 10, p. e13068, 2022.

[12] D. Ramesh and S. K. Sanampudi, "Coherence based automatic essay scoring using sentence embedding and recurrent neural networks," in International Conference on Speech and Computer, Springer, 2022, pp. 139–154.

[13] S. Patidar, A. S. Kataria, and P. Gupta, "Automated essay grading using long short-term memory networks," in Emerging Trends in IoT and Computing Technologies, Routledge, 2022, pp. 66–78.

[14] M. Faseeh et al., "Hybrid approach to automated essay scoring: Integrating deep learning embeddings with handcrafted linguistic features for improved accuracy," Mathematics, vol. 12, no. 21, p. 3416, 2024.

[15] K. Poonpon, P. Manorom, and W. Chansanam, "Exploring Effective Methods for Automated Essay Scoring of Non-Native Speakers.," Contemp. Educ. Technol., vol. 15, no. 4, 2023.

[16] W. Li and H. Liu, "Applying large language models for automated essay scoring for non-native Japanese," Humanit. Soc. Sci. Commun., vol. 11, no. 1, pp. 1–15, 2024.

[17] Y. Wang, C. Wang, R. Li, and H. Lin, "On the use of bert for automated essay scoring: Joint learning of multi-scale essay representation," ArXiv Prepr. ArXiv220503835, 2022.

[18] A. Mizumoto and M. Eguchi, "Exploring the potential of using an AI language model for automated essay scoring," Res. Methods Appl. Linguist., vol. 2, no. 2, p. 100050, 2023.

[19] J. Wang, J. Chen, X. Ou, Q. Han, and Z. Tang, "Multi-level Feature Fusion for Automated Essay Scoring," J. Netw. Intell., vol. 8, no. 1, pp. 76–87, 2023.

[20] T. M. Tashu, C. K. Maurya, and T. Horvath, "Deep learning architecture for automatic essay scoring," ArXiv Prepr. ArXiv220608232, 2022.

[21] F. Li, X. Xi, Z. Cui, D. Li, and W. Zeng, "Automatic essay scoring method based on multi-scale features," Appl. Sci., vol. 13, no. 11, p. 6775, 2023.

[22] A. Sethi and K. Singh, "Natural language processing based automated essay scoring with parameter-efficient transformer approach," in 2022 6th International Conference on Computing Methodologies and Communication (ICCMC), IEEE, 2022, pp. 749–756.

[23] J. Fiacco, D. Adamson, and C. Rose, "Towards extracting and understanding the implicit rubrics of transformer based automatic essay scoring models," in Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications, Association for Computational Linguistics, 2023.

[24] Y. Nie, "Automated essay scoring with SBERT embeddings and LSTM-Attention networks," PeerJ Comput. Sci., vol. 11, p. e2634, 2025.

[25] "Learning Agency Lab - Automated Essay Scoring 2.0." 2024. [Online]. Available: https://www.kaggle.com/competitions/learning-agency-lab-automated-essay-scoring-2

[26] Z. Ellaky, F. Benabbou, Y. Matrane, and S. Qaqa, "A hybrid deep learning architecture for social media bots detection based on BiGRU-LSTM and GloVe word embedding," IEEE Access, 2024.

[27] Y. Liu, H. Qi, and X. Lu, "Enhancing GPT-based automated essay scoring: the impact of fine-tuning and linguistic complexity measures," Comput. Assist. Lang. Learn., pp. 1–20, 2025.

[28] Q. Wang and J. M. Gayed, "Effectiveness of large language models in automated evaluation of argumentative essays: finetuning vs. zero-shot prompting," Comput. Assist. Lang. Learn., pp. 1–29, 2024.

[29] A. Gunduz and M. Gierl, "Automated essay scoring with ChatGPT 3.5 and 4.0," in Presentation at the UBlberta Graduate Student Research in Education Conference, 2024.

Face Sketch Recognition: Ethnic Groups Classification and Recognition Via a VGG16 Model Approach

Khalid OUNACHAD[1], Mohamed EL GHMARY[2]
USMS, Sultan Moulay Slimane University, Beni Mellal, Morocco[1]
ENSAM, Mohammed V University, Rabat, Morocco[2]

***Abstract*—In the law enforcement investigation, the police use sketching techniques to identify suspects from an eyewitness's memory. Many automatic face sketch recognition systems that determine the perpetrator's appearance from the face image datasets have been proposed. The aim is to conduct the arrest of the right offender. We propose this work to carry out a search based on the ethnicity criterion to speed up this automatic identification and to help authorities execute fast responses by launching the retrieval process only in a part of the dataset of face images. The goal of this study is to enhance the accuracy of ethnic face sketch classification by using the convolutional neural network built on the VGG16 architecture. The FairFace dataset, which includes seven ethnic face images: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino|Hispanic, was employed in the study. We convert the face images dataset to face sketch images, and we optimize the VGG16 model for seven classification outputs. This work shows that the VGG16 deep learning model offers a reliable, automated approach for ethnic face sketch classification and recognition. The used model achieved an accuracy reaching above 94% and produced a low false negative rate, which is crucial for minimizing undetected cases.**

***Keywords*—Forensic face sketch; ethnic classification; deep learning; transfer learning; optimized VGG16 model**

I. Introduction

Race is basically defined as a mixture of physical, behavioral, and cultural attributes, but ethnicity recognizes differences between people mostly based on language and shared culture [1]. In other words, race is inherent in our biology, and therefore inherited across generations, while ethnicity is based on native region or culture and acquired; it can be tied to the self-ascribed.

VGG16 (Visual Geometry Group-16 deep layers) is a deep convolutional neural network model introduced by the team at the University of Oxford [2]. It is most famous for its architectural simplicity and its performance, especially in image classification, but it needs many calculations, and it's a slow algorithm without a GPU. Among its applications, in addition to image classification, are feature extraction, object detection, and image segmentation. Generally, VGG16 is one of the most experimented CNN architectures for facial soft biometrics analysis. It achieved a significant success thanks to its deep architecture (around 138M parameters, 13 convolutional layers, and 3 fully connected layers), which allows it to better generate results even in the presence of small training sets.

Human Face Ethnic Groups Identification and Recognition (FEGIR) is an area of non-verbal language in day-to-day life communication. Ethnic facial features are some of the most important face features, but humans struggle and find it hard to try to differentiate between members of social outgroups, including different races [3]. Moreover, FEGIR systems have attracted numerous researchers and have been conducted and utilized in various fields since they attempt to overcome the problems and factors weakening these systems, including the problem of image classification, also due to their large-scale applications in face analysis, particularly face recognition [4]; sharpening the global focus on ethnicity and race in the time of COVID-19 disease [5],[6], career progression analysis [7], disentangling the stress process [8], and social crime [9].

Ethnic-based separation among humans is classified into six minimum categories: American Indian or Alaska Native, Asian, Black or African American, Hispanic or Latino, Native Hawaiian or Other Pacific Islander, and White [7]. However, based on their skin colors, we have five kinds of ethnicity:

red, white, black, brown, and yellow. It can also be tied up to some specific geographic regions, countries, or nationalities: symbolized relationships between the "races" and the countries. The U.S. Census Bureau also considers five race categories: White, Black or African American, American Indian or Alaska Native, Asian Native, and Hawaiian or Other Pacific Islander. The census considers some regions of origin for each race category. We use seven ethnic kinds of faces: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino, referring to the used dataset of face sketch images.

Ethnic group identification and recognition systems aim to automatically classify ethnicity in a dataset of photos or sketches (Fig. 1). It is based on a two-dimensional image of human subjects. The automatic task of ethnicity recognition is challenging work and explicitly difficult: Ethnic face classification and recognition can be done in many ways, but this study is concerned with the ethnic face sketch [10],[11] classification and recognition based on two-dimensional images of people's sketched faces.



Fig. 1. Examples of ethnic groups face sketch classification and recognition.

There are a large number of datasets available for human ethnicity classification research, some of which are private and some public. The FairFace database [10] is most commonly used in ethnicity classification, but until the redaction of this article, there was no face sketch dataset labeled with ethnic features used or cited in any related research. To resolve this situation and this problem, we created our ethnic face sketch dataset by converting face images of the FairFace dataset to face sketch images based on a method encompassing image inversion, Gaussian blur, and dodging fusion techniques.

VGGFace achieved an impressive accuracy in face recognition [12] and age estimation [13],[14] and has been successfully experimented with also for gender recognition [15],[16]. For this reason, we believe it can be effective for ethnicity recognition and classification purposes. In our proposed approach, we optimized the VGG16 model for seven classification outputs relative to our used face sketch dataset since it contains seven ethnic classes. The results of this work show that deep learning models, including VGG16, can offer a strong and automatic means of face sketch ethnicity classification and recognition. Beyond their intellectual value, these findings have practical ramifications for authorities in law enforcement investigation since they allow faster and more consistent ethnicity recognition and automatic face sketch recognition [17],[18].

The rest of the study is organized as follows: In Section II, related studies, an overview of the proposed VGG16 model is presented, emphasizing different modern searches when it's used. Many areas are mentioned, like computer vision, health disease classification, agriculture disease classification, and the civil engineering & construction sector. In Section III We present a brief explanation of the VGG16 models. In Section IV, we explore our approach in detail and describe the performance of our proposed algorithm. Section V presents the tests and experiments carried out during this study. Finally, we summarize this article with a conclusion.

II. Related Works

Ethnicity classification, a crucial facet in computer vision and artificial intelligence, has witnessed significant advancements in recent years. VGG16 is a VGGFace model trained from scratch for face recognition on almost 1 000 000 images. This CNN is probably the most adopted architecture for facial soft biometrics analysis, and we believe it can also be effective for ethnicity recognition and classification purposes. There has been a wealth of research on the VGG16 model, as well as on its application in areas such as computer vision. Specifically, it's used in various subfields such as image recognition
[19]-[21], health disease classification [22]-[24], agriculture disease classification or food security [25]-[26], and the civil engineering & construction sector [27], the last two areas were cited to indicate the performance of the use of VGG16 and considered as a related works to the use of the VGG16 model.:

In the image recognition subfield, for example, the study suggested by Zhu & Li [19] offers an improved version of VGG16 combined with wavelet analysis and marginal Fisher analysis (dimensionality reduction) to improve image recognition on different animal images. Their approach recognition accuracy reached 97%. Although in [20], R. Sewada and H. Goyal propose an adaptation of VGG-16 for the classification of multispectral satellite images. The used images are visible and near-infrared images. Their proposed modifications to the VGG-16 approach attain a higher classification accuracy rate of ±96.7%. While Genç & Yalman [21] propose an approach to recognize aircraft using satellite images based on CNN models. They detail a comparison study of VGG16 and VGG19 architectures. Based on their obtained results, VGG16 achieves an accuracy of approximately 82.67%, and VGG19 reaches ~89.29%.

However, in the medical domain, specifically in the health disease classification subfield, the authors of [22] propose a hybrid approach to detect heart disease by combining deep learning, machine learning, and explainable artificial intelligence. The explainable artificial intelligence techniques allow for the identification of the most decisive clinical variables in predicting cardiac risk. They use the pre-trained VGG16 model to extract 14 features from medical data converted to image format. These features are then merged with the original clinical data and classified using several machine learning algorithms: SVM, Random Forest, K-NN, etc. The VGG16 combined with the random forest model achieves the best performance, with an accuracy of approximately 92%. All the heart disease datasets used for their study are from Kaggle, comprising publicly available and synthetic data samples. In another health disease classification case [23], F. Tanjim and S. Hamdy describe and detail a method based on learning transfer with VGG16 to detect breast cancer from histopathological images. The authors use the pre-trained VGG16 model and adapt it to the BreaKHis medical dataset to distinguish between benign and malignant tumors. They improve the number of images using modern techniques such as data augmentation, dynamic tuning of the learning rate, and data regularization. They also propose a modified version of the pre-trained VGG16 model, which they named M-VGG16. The results show that the M-VGG16 model achieves an accuracy of approximately 94%. Although, in study [24], a retinal disease detection case, Jaimes et al. offer an approach for identifying four categories of retinal diseases from optical coherence tomography (OCT) images from the Kaggle database using a VGG16 transfer learning-based model. It relies on the pre-trained VGG16 model, adapted and refined for this specific task. The results achieve an accuracy of approximately 95%. The authors also use the Grad-CAM technique to visualize areas of the image that influence the prediction, thus improving the interpretability of the model.

Also, in the agriculture disease classification subfield related to the food security subdomain, in [25], the authors propose a CNN approach to improve the classification of tomato leaf diseases by adapting the classic VGG16 architecture to this agricultural domain and by optimizing its hyperparameters using Bayesian optimization. The innovative aspect of the used technique lies in the use of Bayesian optimization to automatically select the best hyperparameters for the model. The experiments are conducted on the Tomato Leaf Disease Detection (TLDD) dataset, a dataset of images of healthy and diseased tomato leaves, which is publicly available on the Kaggle platform. The results show that the proposed model achieves approximately 97% accuracy, surpassing the standard version of VGG16 (89.0%). The authors improve and modify the original VGG16 architecture by adjusting certain layers and adding regularization mechanisms to adapt it for plant disease detection. Although in a second agricultural classification disease case [26], Sahputra et al. compare two deep learning models, VGG16 and MobileNet, for classifying rice leaf diseases such as bacterial leaf blight, brown spot, leaf blast, and narrow brown spot. They use a set of images of infected and healthy leaves by using a dataset of 2,190 rice leaf images divided into five classes, apply preprocessing and data augmentation, and then evaluate the two models based on accuracy, speed, and complexity. The results show that VGG16 achieves superior accuracy, offering better classification performance. In contrast, MobileNet is lighter and faster. The VGG16 model achieved an accuracy of 98%, while the MobileNet model reached 95% accuracy.

In the civil engineering & construction sector, one example related to urban construction and infrastructure is the study carried out by Chen et al. [27]; it is a smart solution for structural health monitoring of buildings and dwellings. The authors propose a steel-reinforced concrete corrosion crack detection method based on improved VGG16. They propose a deep learning approach for detecting corrosion cracks in reinforced concrete using an improved VGG16 model. They adapt the VGG16 architecture to better capture fine crack details in images of concrete structures, while simultaneously reducing computational complexity. Chen et al. use the SDNET2018 and DeepCrack concrete crack datasets for training and testing experiments, and their proposed model achieved a precision of 94.4%.

As a contribution, this article proposes an ethnic face sketch classification and recognition method based on an improved VGG16 model. Our work is inspired in part by the recent and successful method that has shown that the use and integration of the VGG16 model in classification scenarios leads to obtaining a better score; it could be used to perform well. We believe it can also be effective for ethnicity recognition and classification purposes. As part of our research project, in which a face sketch recognition system was developed. We achieved an impressive accuracy in the face sketch recognition task [17], face sketch gender recognition task [11],[18],[28],[29], face sketch age estimation and recognition task [30], and face sketch emotions classification and recognition task [31]-[33]. What remains is the development of a phase of the overall project that allows for face sketch ethnicity recognition. For this reason, our project aspires to blaze new trails in the realm of face sketch ethnicity classification. We explore the possibilities of creating custom models tailored specifically to the challenges posed by this complex classification task. Our objective is to develop a classification system that surpasses conventional standards and excels in capturing the intricate subtleties of ethnicity. We leverage the capabilities of deep learning to advance the science of ethnicity and human race classification, fostering a world that values and appreciates the beauty of our differences.

III. VGG16 Deep Convolutional Neural Network Model

VGG16 is a well-known pre-trained CNN model from ImageNet that consists of 16 layers with learned weights, mainly using 3×3 convolutions and 2×2 max pooling to extract and reduce image features. It's trained from scratch for face recognition on almost a million images from the ImageNet dataset. This pre-trained CNN model is one of the most experimented-with CNN architectures for facial soft biometrics analysis, and it is very effective at extracting useful features from images.

The diagram of Fig. 2 illustrates multiple layers of interconnected neurons (nodes), which allow the network to learn complex, hierarchical features from data. The different sections, indicated by varying colors and structures, include multiple and different layers: 2D convolution layers, 2D max pooling, flatten, and dense. We have already explained the main function and task of convolutional layers and max pooling in section 2.1, while the primary role of dense flatten is to convert the 3D output from the VGG16 base to 1D so that it can be used in fully connected layers, and the dense layers are the fully connected layers. As evident from the name, it consists of 16 levels and requires an input of 224×224 pixels.

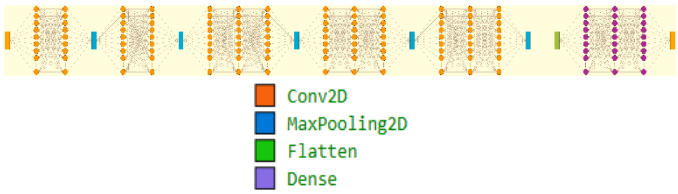


Fig. 2. VGG16 architecture scheme.

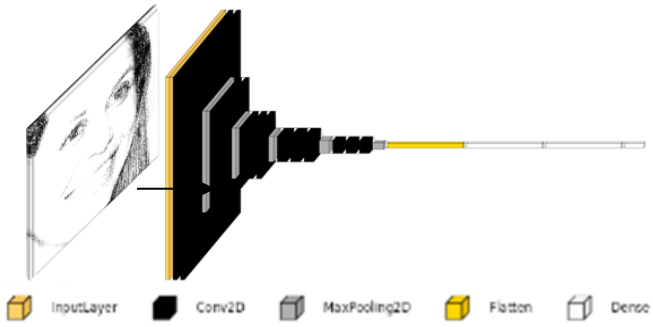


Fig. 3. VGG16 simple architecture 3D scheme.

Fig. 3, thus, clearly highlights the network's progression, in particular: gradual reduction of spatial size, increase in feature depth, and finally, classification via the fully connected layers. The architecture follows a very regular pattern: consistent filter sizes, double or triple convolution per block, then pooling,

which simplifies optimization while allowing the extraction of increasingly abstract visual features. The VGG-16 architecture begins with an input layer receiving images of size 224×224×3, and during the progression, outputs a unique prediction, allowing classification among 1000 outputs. Fig. 3 clearly illustrates the model's ability to effectively reduce the dimensions of the input image before flattening it into a simple final vector.

The number of used parameters in each layer and for each block is detailed in Table I; in addition to the type of layer, it indicates the output shape obtained at the end of each layer. The number of parameters is zero for the input layer, all the MaxPooling2D layers used, and the flatten layer. The model uses 1 792 in the first 2D convolution layer and 4097000 in the last one. 102 764 544 parameters are used in the first dense layer; it's the biggest number of parameters used in the model. The total parameters of the model are 138357544 (527.79 MB), and the trainable parameters are 138 57544 (527.79 MB), so the model doesn't have a non-trainable parameter (0.00 B).

TABLE I. TYPE LAYERS, OUTPUT SHAPE AND PARAMETERS OF VGG16 MODEL

Layer (type)	Output Shape	Parameters
input_layer (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1 792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36 928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73 856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147 584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295 168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590 080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590 080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1 180 160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2 359 808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2 359 808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2 359 808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2 359 808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2 359 808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
flatten (Flatten)	(None, 25088)	0
fc1 (Dense)	(None, 4096)	102 764 544
fc2 (Dense)	(None, 4096)	16 781 312
predictions (Dense)	(None, 1000)	4 097 000

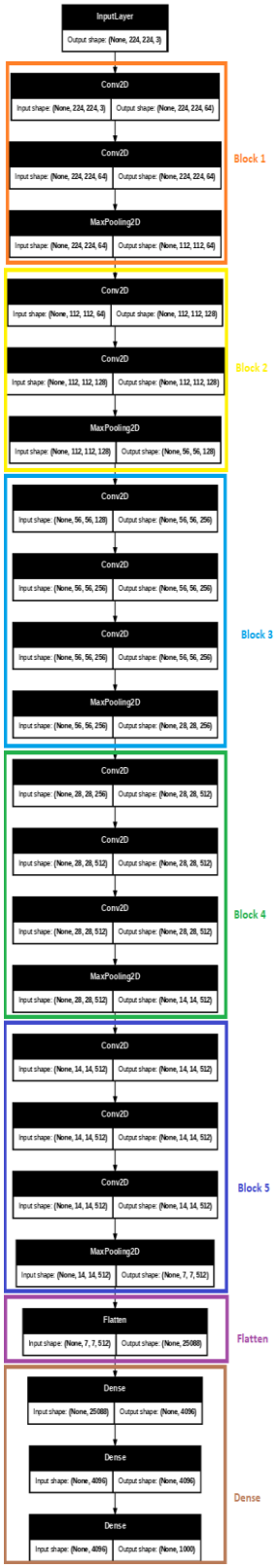


Fig. 4. VGG16 deep architecture scheme.

In Fig. 4, the VGG16 deep and detailed architecture is shown; it's composed of several successive blocks of convolutional layers followed by max-pooling layers. The first block applies two convolutions with 64 filters of size 3×3, maintaining the input resolution at 224×224, before reducing it to 112×112 via max-pooling. The second block repeats the same structure but with 128 filters, reducing the size to 56×56 after pooling. The third block consists of three convolutions with 256 filters each, followed by pooling that reduces the resolution to 28×28. The fourth block applies three convolutions of 512 filters before pooling, which reduces the size to 14×14. The fifth and final convolutional block applies three more layers of 512 filters, then a final max-pooling, resulting in a feature map of dimensions 7×7×512. The output of the convolutions is flattened into a vector of 25088 values, which feeds two fully connected layers of 4096 nodes each. The final dense layer has 1000 nodes and SoftMax activation, corresponding to 1000 possible output classes. The total of these output classes is chosen relative to the ImageNet dataset.

IV. Proposed Approach

The proposed framework for classifying ethnic groups from face sketches consists of several successive steps. First, images from a specialized dataset of face sketches are fed into the input system. This raw data then undergoes preprocessing, including automatic face detection, pixel intensity normalization, and image resizing to the format required by the VGG16 architecture (224×224×3).

The proposed system takes as input an image $I \in \mathrm{R}^{H \times W \times 3}$ from the face sketches dataset. A facial detection operation $D(I)$ extracts the face region I_f, and the D operation encompasses exactly the HOG and SVM methods. The resulting image I_f is then normalized as in Eq. (1):

$$I_n = \frac{I_f}{255} \tag{1}$$

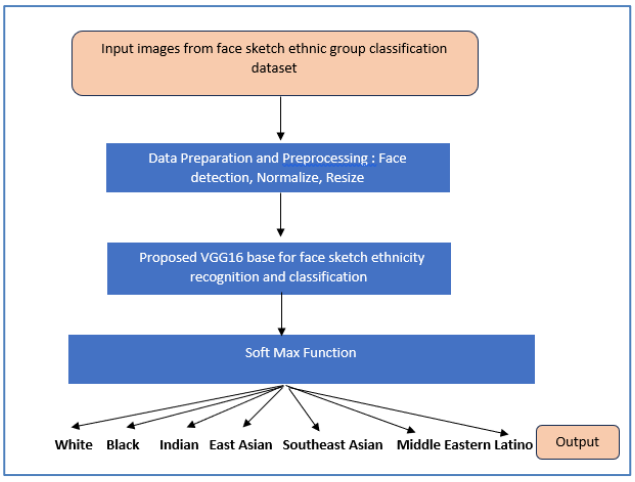


Fig. 5. Our proposed framework based on the VGG16 Model for face sketch ethnic group classification and recognition: An overview.

The normalization step converts pixel values to a 0-1 scale by dividing all pixel values by 255. Then, the resulting image I_n is resized to Ir through Eq. (2). The resize function resizes the image to 224x224 to conform to the VGG16 model shape.

$$I_r = Resize(I_n, 224 \times 224 \times 3) \tag{2}$$

In the feature extraction phase, a tailored VGG16 model is used as the basis network. This acts as a deep feature extractor, capable of efficiently capturing the facial structures present in the sketches. The pre-trained convolutional layers can be retained or fine-tuned to optimize performance on this type of non-photographic data.

The preprocessed image Ir is then propagated through a modified VGG16 network, consisting of 3×3 convolutional blocks followed by ReLU activation functions and max-pooling layers. Each block b performs a nonlinear transformation of the image described by Eq. (3):

$$F_l = \sigma(W_l * F_{l-1} + b_l) \tag{3}$$

where W_l is the convolutional filter of layer l, * is the convolution operation, and σ is the ReLU function.

After the last convolutional block, a flatten operation produces a feature vector $z \in R^d$. Finally, the representations generated by VGG16 are passed to a dense layer equipped with a SoftMax activation function, ensuring a probability distribution across all defined categories. The dense classifier then applies the following equation as in Eq. (4):

$$y = W_z + b \tag{4}$$

where $y \in R^K$ and K is the number of classes.

The final Softmax layer calculates a probability distribution as in Eq. (5):

$$P(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{k} e^{y_j}} \tag{5}$$

For C ethnic groups, the predicted probability for class c is P(y = c | Fl), calculated as mentioned in Eq. (5). The loss function L in Eq. (6) is categorical cross-entropy:

$$L = -\sum_{c=1}^{C} y_c \log\left(P(y = c|F_l) \right) \tag{6}$$

The predicted class corresponds to the result of the equation in Eq. (7):

$$\hat{c} = \arg max_i \; P(y_i) \tag{7}$$

The class with the highest probability is selected as the model output. An overview of our proposed framework based on the VGG16 model for face sketch ethnic group classification and recognition is shown in Fig. 5. This framework thus enables automatic and structured classification of face sketches according to the seven predefined categories in the used dataset.

The details of the proposed VGG16 base algorithm for face sketch ethnic group classification and recognition can be summarized as follows: Fig. 6. The pseudocode is also shown in Fig. 7.

Step1: A face sketch is input to the trained VGG16-based model (face sketch preprocessing)

Step 2: Deep feature extraction by convolutional layers (feature learning)

Step3: The classifier predicts the ethnic group label(classification)

Step4: The class with the highest softmax probability is selected (ethnic group recognition)

Fig. 6. Main stages of the proposed system.

Input: Face sketch dataset D = {(Is, C)}

Is=Face sketch image, C= Ethnic group class

Pre-trained VGG16 model

Output: Predicted ethnic group label ĉ

1- Load pre-trained VGG16 model

2- Remove fully connected layers

3-Freeze early convolutional layers

4- Fine-tune deeper layers on sketch dataset

5- For each sketch image Is in D do

Resize Is to 224×224

Normalize Is

Extract features with VGG16 Model

f = VGG16(Is)

end for

6- Train classifier Softmax using features f

7- For a test sketch Itest:

Extract feature ftest

Predict ethnic group ĉ

end for

8- Return ĉ

Fig. 7. Pseudo-code of the proposed algorithm: VGG16-based face sketch ethnic groups classification and recognition.

Each face sketch image is preprocessed by resizing to 224×224×3 and normalizing pixel values before being passed through the modified VGG16 network to extract high-dimensional deep feature vectors. Partial layer freezing allows for the preservation of generic descriptors. The extracted features are subsequently used to train the SoftMax classifier, which predicts the ethnic group label for test face sketches.

V. Experiments and Results

A. Dataset and Ethnicity Groups Taxonomy

To demonstrate the effectiveness of the proposed method, we proceeded to acquire a dataset of ethnic groups' face sketch pictures. We used the FairFace [34] face images dataset that we converted to face sketch images. It is used as a source of real face images to generate sketches using inversion, Gaussian blur, fusion, "dodging," and sketch-synthesis methods. In the context of ethnic face sketch classification and recognition, FairFace can serve as the best source of real face images from which sketches can be generated.

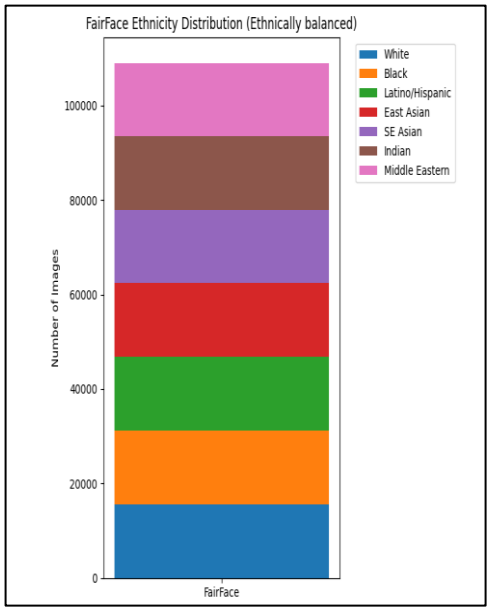


Fig. 8. FairFace ethnicity balanced distribution.

FairFace is a face image dataset that is ethnically balanced. It is a large-scale facial image dataset and contains 108501 images of faces in the wild, annotated with seven ethnicity categories [White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino/Hispanic], gender labels (male and female), and age group ranges. Images were collected from the YFCC-100M Flickr dataset and labelled with race groups, gender, and age groups. FairFace [35],[36] is ethnically balanced, as shown in Fig. 8, ensuring that all groups are fairly represented, which makes it particularly suitable for training models for ethnicity group classification and recognition. Its images exhibit natural variations in pose, lighting, and background, providing realistic and perfect conditions for model training. Fig. 9 details the FairFace ethnic groups' class distribution, Table II. gives a summary dataset description, and Fig. 10 includes random samples from the same dataset, where each line in Fig. 10 represents and corresponds to an ethnic group. These ethnic groups are illustrated from the white ethnicity to the Latino| Hispanic ethnicity, suitable to order in Fig. 9.

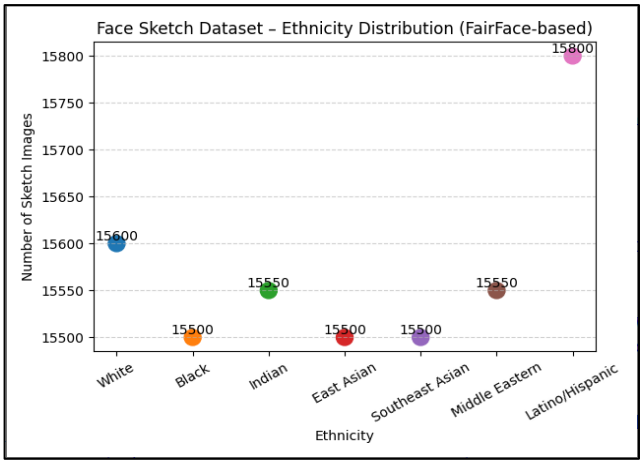


Fig. 9. FairFace ethnicity distribution.

TABLE II. FAIRFACE DATASET DESCRIPTION (ETHNICITY)

Number of images	Characteristics	Annotations	Ethnicity Classes	Ethnic groups Description	Number of Images (~)	% of Total
~108000	-Varied, mostly high-quality images - Contains in-the-wild images, (faces with natural conditions) -variations in pose, lighting, & background.	-Race/ Ethnicity: 7 categories -Gender: M|F -Age groups: 0–2, 3–9, 10–19, 20–29, 30–39, 40–49, 50–59, 60–69, 70+	-White	-Caucasian heritage	15600	~14.4%
			-Black	-African descent	15500	~14.3%
			-Indian	-Indian subcontinent	15550	~14.3%
			-East Asian	-East Asian heritage	15500	~14.3%
			-Southeast Asian	-Southeast Asian heritage	15500	~14.3%
			-Middle Eastern	-Middle Eastern heritage	15550	~14.3%
			-Latino / Hispanic	-Latin American heritage	15800	~14.6%



Fig. 10. Random samples from the FairFace dataset.

We divide the dataset into seven data subsets suitable for each kind of basic ethnicity: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino|Hispanic. Table II shows the percentage of each ethnic group; for each ethnicity class, the number of images is approximately 15500, with 14.4% of the average total. The reduction in the number of images is intended to simplify calculations, and the choice of images for the subclasses is arbitrary. Our chosen number of images in each ethnicity class is approximately seven thousand (~7000).

The generated face sketch ethnicity dataset from FairFace is used for training, validation, and testing our proposed approach. As shown in Table III., sixty percent (60%) of the images (29400) are used for the training phase, nineteen percent (20%) of the images (9800) are used for the validation task, while the remaining twenty percent (20%) of the images (9800) are reserved for the testing phase to assess the generalization performance of the proposed approach on unseen face sketch images. The dataset was divided proportionally to ensure that each ethnicity class was represented in all seven subsets. To achieve a complexity that is feasible on our machine, in our experimented and generated face sketch ethnicity dataset, the number of images in each ethnicity class is approximately seven thousand (~7000), 49000 images in total. Images were determined by data availability and the requirements of the VGG16 model training and testing process.

TABLE III. OUR DATASET SPLIT DISTRIBUTION ACROSS TRAINING, VALIDATION, AND TESTING PHASES

Phase task	Number of images	% of Total
Training	~29400	~60%
Validation	~9800	~20%
Testing	~9800	~20%

B. Tools

The proposed VGG16 model was developed, trained, and tested in Python using TensorFlow [39] to give access to pre-trained networks. Pandas was used for label file extraction and oversampling, CV2 for data preprocessing and facial recognition, and Scikit-learn for splitting data into training, validation, and testing sets. All computations were run in Google Colab using GPUs. The pre-trained VGG16 was downloaded via Keras through TensorFlow and subjected to transfer learning.

C. Image Pre-Processing

This stage prepares the data for the training process. Several steps are performed, including face detection, sketch image generation, normalization, and image resizing. We used the methods and operations detailed and mentioned in section IV to normalize and to resize to 224x224 all FairFace dataset images. We also convert all face images of the FairFace dataset to face sketch images by using a combination of inversion, Gaussian blur, and fusion image techniques. Fig. 11 shows a random sample of ethnic groups' face sketches generated and converted from FairFace by using these techniques. They include all ethnic classes:

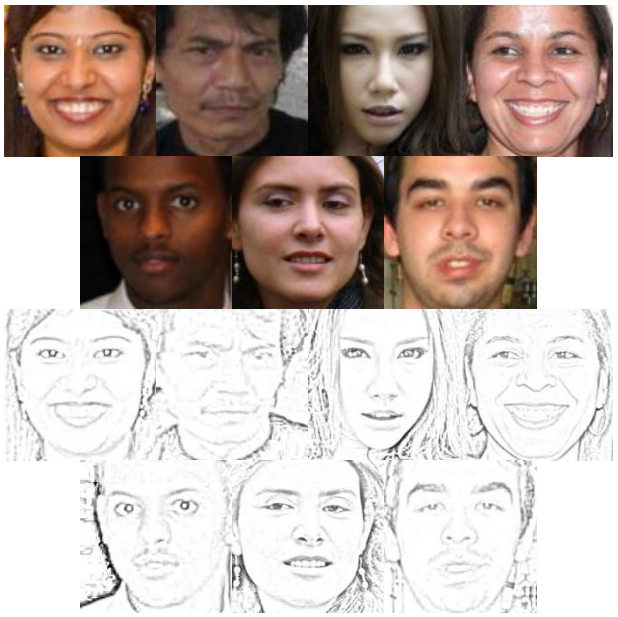


Fig. 11. Random samples of ethnic groups face sketches generated from the FairFace dataset images.

In our proposed approach, the first step in the preprocessing process is the face detection task. We use for this stage the famous Viola and Jones algorithm [40]-[42]. Fig. 12 illustrates some obtained results; the used method, based on the Viola-Jones technique, enables the faces to be detected perfectly.

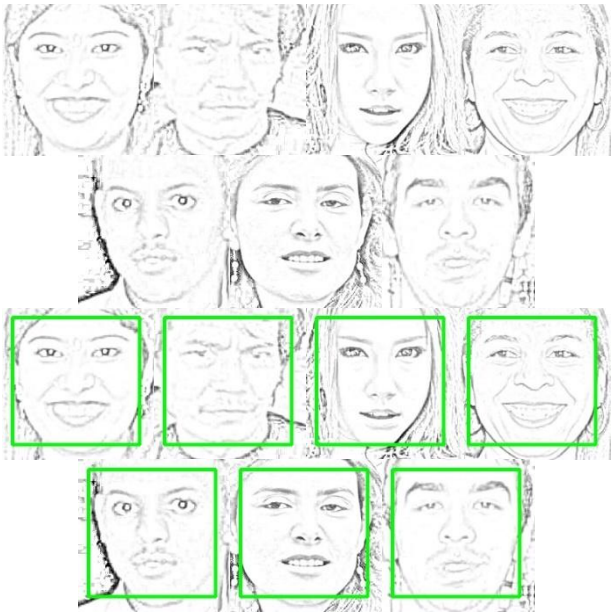


Fig. 12. Random Samples face sketch detection.

Fig. 13 summarize the experimental workflow for the data preparation and preprocessing task. The output image from the preprocessing phase will be the input image of the ethnic classifier based on the VGG16 model; it is an image containing only the detected face sketch area, which has been resized to a 224×224-pixel image and normalized.

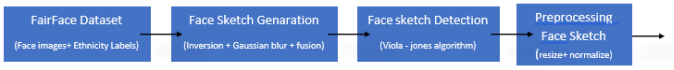


Fig. 13. Detailed experimental workflow for data preparation and preprocessing task.

D. Performance Metrics

When training a classifier, the choice of a scoring metric is a critical factor in achieving optimal classifier accuracy. The selection of the appropriate evaluation scale is of paramount importance to discriminate and ensure superior performance [37],[38]. Meanwhile, the model evaluation stage was carried out to assess the accuracy, consistency, and performance of the model in ethnic group face sketch classification and recognition. Model performance measurements used a confusion matrix and metrics such as accuracy (Eq. 8), precision (Eq. 9), recall (Eq. 10) and F1-score (Eq. 11). To compare model predictions with actual data, below are some equations for the proposed VGG16 model evaluation:

$$Accuracy = \frac{\mathrm{TP+TN}}{TP+TN+FP+FN} \quad (8)$$

$$Precision = \frac{\mathrm{TP}}{TP+FP} \quad (9)$$

$$Recall\ (sensitivity) = \frac{\mathrm{TP}}{TP+FN} \quad (10)$$

$$\mathrm{F1-Score} = 2 \times \frac{\mathrm{Precision \times Recall}}{\mathrm{Precision+Recall}} \quad (11)$$

$$\mathrm{F_{ß}-Score} = (1+ß^2) \times \frac{\mathrm{Precision \times Recall}}{ß^2 \times \mathrm{Precision+Recall}} \quad (12)$$

The $F_ß$-score in Eq. (12) is a generalized version of the classic F1-score, which allows more weight to be given to precision or recall, depending on the specific application. The score results will be the same as the classic F1-score, while ß is equal to one. While ß is greater than one, the focus is on recall; if less precision is needed and ß is less than one, more emphasis is placed on precision.

A True Positive (TP) occurs when the model predicts a positive output and the actual outcome is also positive, indicating a correct prediction for the positive class. Similarly, a True Negative (TN) occurs when the model predicts a negative output and the actual output is also negative, indicating a correct prediction for the negative class. Conversely, a False Positive (FP) occurs when the model incorrectly predicts a positive output when the actual output is negative. And finally, a false negative occurs when the model incorrectly predicts a negative output when the actual output is positive.

E. Results and Discussion

The generated dataset for this study consists of face sketches labeled by ethnicity. Seven classes were identified: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino|Hispanic. A total of 49000 face sketch images were randomly collected and generated from FairFace and divided into three subsets: training data, validation data, and test data. This proportional distribution ensured that each class was represented in all three subsets, thus optimizing the training, validation, and testing of the VGG16 model. The number of images used for each ethnic group class is balanced as detailed in Table IV below:

TABLE IV. FACE SKETCH IMAGE DATA DISTRIBUTION

Image Ethnicity Class	Training Data	Validation Data	Testing Data
White	~4200	~1400	~1'00
Black	~4200	~1400	~1400
Indian	~4200	~1400	~1400
East Asian	~4200	~1400	~1400
Southeast Asian	~4200	~1400	~1400
Middle Eastern	~4200	~1400	~1400
Latino|Hispanic	~4200	~1400	~1400
Total (7000)	~29400	~9800	~9800

Tests were conducted on 49000 data face sketch images distributed across seven classes of ethnic group face sketches: each ethnicity class contained approximately 7000 images. To evaluate the learning performance of the VGG16 model, we first present the performance of optimized VGG16-based ethnic groups classification on the face sketches generated from the FairFace dataset, and we present the accuracy and loss plots. The obtained results are shown in Fig. 14 and Fig. 15. These plots illustrate the model's ability to learn from the training and validation data, as well as the stability of the training process. The accuracy, precision, recall, $F_{ß=1}$-score $F_{ß=2}$- score and $F_{ß=0.5}$-score for the face sketch ethnic group classification and recognition using our adopted approach are shown in Table V.

TABLE V. PERFORMANCE OF OPTIMIZED VGG16-BASED ETHNIC GROUPS CLASSIFICATION ON THE FACE SKETCHES GENERATED FROM THE FAIRFACE DATASET

Metric / Model	Accuracy	Precision	Recall	$F_{ß=1}$-score	$F_{ß=2}$ score	$F_{ß=0.5}$ - score
Adopted VGG16 (Proposed)	94.00	93.80	94.60	94.70	34.64	93.62

The optimized VGG16 model accuracy graph (Fig. 14) shows a consistent increase in accuracy since the beginning of training, reaching above ~95% at the 100th epoch. The training and validation data accuracy patterns are similar, with no significant differences. This indicates that the model has good generalization capabilities and does not experience overfitting. The consistency between training and validation accuracy also indicates that the model is able to learn effectively from the data provided.

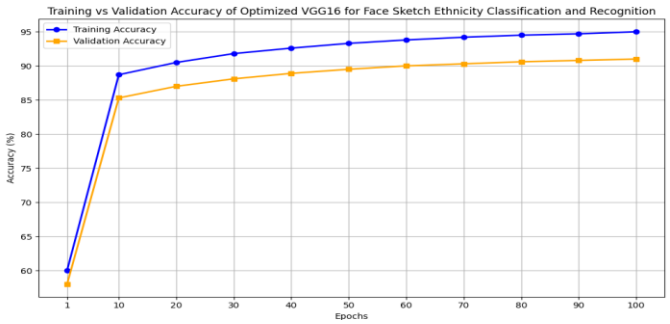


Fig. 14. Optimized VGG16 accuracy graph.

Meanwhile, the loss plot of the optimized VGG16 model(Fig. 15) shows a sharp decrease during the early epochs for both the training and validation data. Subsequently, the loss tends to stabilize at a low value (close to ~0), despite slight fluctuations in the validation loss at certain epochs. However, these fluctuations remain small and within acceptable limits. This behavior confirms that the training is proceeding correctly and that the model does not exhibit overfitting or underfitting problems.

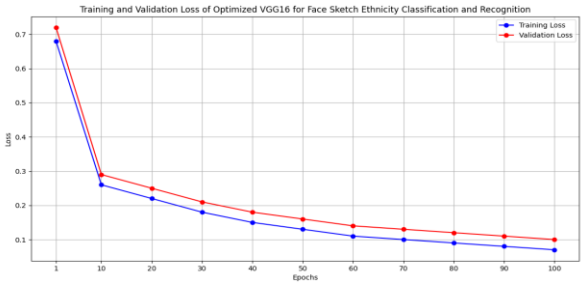


Fig. 15. Optimized VGG16 loss graph.

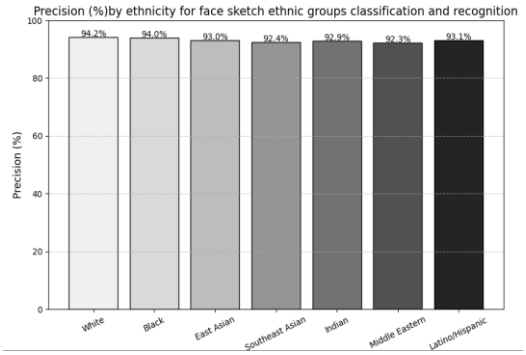


Fig. 16. Class-wise precision (%) for face sketch ethnicity classification and recognition (Optimized VGG16).

Once the model is trained, it is tested on the test data. The results of this test are presented in the visualization in Fig. 16. The figure illustrates the class-wise precision performance for face-sketch-based ethnicity classification and recognition across seven basic ethnic groups. The results show clearly high precision values for all ethnicity classes, ranging from 92.3% for Middle Eastern to 94.2% for the White ethnic group, indicating strong discriminatory capability of the optimized VGG16 proposed model. The highest precision is achieved for the White (94.2%) and Black (94.00%) ethnic groups, while slightly lower performances are obtained for Middle Eastern (92.3%) and Southeast Asian (92.4%) ethnic groups. Overall, the balanced distribution of precision scores across ethnicities demonstrates the robustness and fairness of the optimized VGG16 proposed model for face sketch ethnicity classification and recognition. We deduce that class-wise precision (%) for face sketches, ethnicity classification, and recognition using an optimized VGG16 model shows consistently high performance across all seven basic ethnic groups.

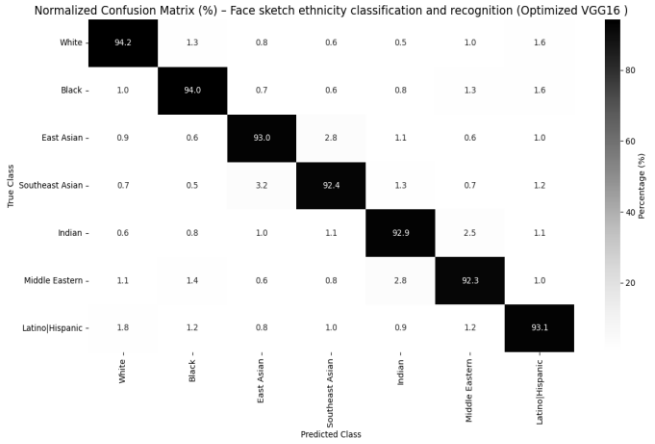


Fig. 17. Optimized VGG16 confusion matrix (%) in face sketch-based ethnicity classification and recognition.

Fig. 17 presents the normalized confusion matrix (%) for face sketch ethnic group classification and recognition using the optimized VGG16 model. The matrix brings to the fore strong diagonal dominance, indicating high correct classification rates across all seven basic ethnic groups. Precision values on the main diagonal exceed 92% for all classes, demonstrating the effectiveness of the proposed approach: White (94.2%), Black (94.0%), East Asian (93.0%), Southeast Asian (92.4%), Indian (92.9%), Middle Eastern (92.3%), and Latino|Hispanic (93.1%). Minor misclassifications are observed between visually similar ethnic groups, such as East Asian and Southeast Asian, as well as Middle Eastern and Indian.

Fig. 18 also illustrates the inter-class confusion matrix for face sketch–based ethnicity classification and recognition using the optimized VGG16 model. The matrix emphasizes misclassification patterns between ethnic groups (distribution of misclassification errors). Overall, the confusion rates remain low, indicating strong class separability.

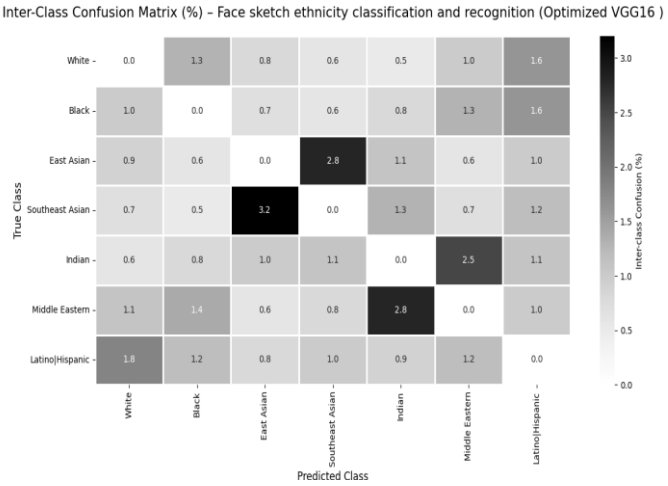


Fig. 18. Inter-class confusion matrix (%) for face sketch ethnic group classification and recognition using optimized VGG16.

The most noticeable and slight confusions are observed between visually related ethnic groups, such as between East Asian and Southeast Asian ethnic classes, which can be attributed to shared facial sketch features and characteristics. Similarly, moderate confusion is observed between Indian and Middle Eastern ethnicity classes, reflecting visual similarities in facial characteristics commonly found in sketch representations. Especially, these confusions are symmetrical and very low in magnitude, proving that the model does not exhibit systematic bias toward any particular ethnic group.

For example, VGG16 incorrectly classified some cases shown in Fig. 19. The first case is classified as Indian with 54% precision, so in fact the real class ethnic group is Southeast Asian. The prediction of the second case is East Asian with 57% precision; however, the real ethnicity is Southeast Asian. For the third case, in which the real class is East Asian, its prediction is 60% Southeast Asian. The last case is classified as Indian with 55% precision, so in reality, it is a Middle Eastern face sketch. This indicates that face sketches exhibit similar spot patterns or discoloration, making them difficult for the model to distinguish.

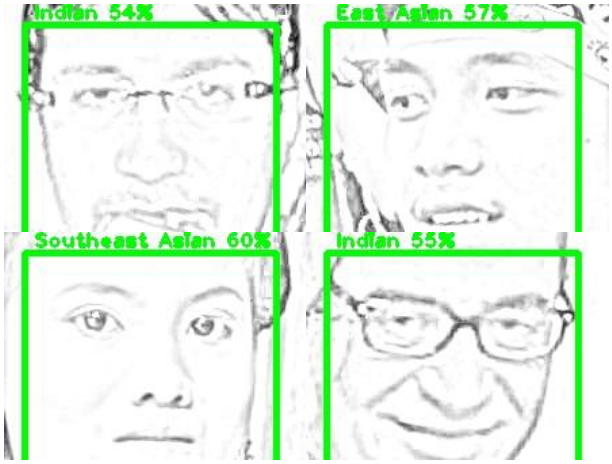


Fig. 19. Sample fails classified cases.

However, the model correctly identified the other ethnicity classes shown in Fig. 20. The figure illustrates some cases correctly classified as all seven basic ethnic groups, demonstrating the model's strong ability to identify ethnic groups with more distinctive characteristics. The first case in Fig. 20. is classified as Black with 91.7%; the second case is predicted Latino|Hispanic with 87.4%; the third one is classified as 82.6% Middle Eastern; the fourth case is predicted to be Southeast Asian with 81.9%; the fifth case is classified as 87.9% East Asian; the sixth case is predicted to be 95% White; and the last case is 93.16% Indian.

It is reported that for some face sketch images, the enhanced approach doesn't even allow for face detection in the image, and this problem is because the image doesn't include both eyes or the image isn't clear enough, as in the examples in Fig. 21:

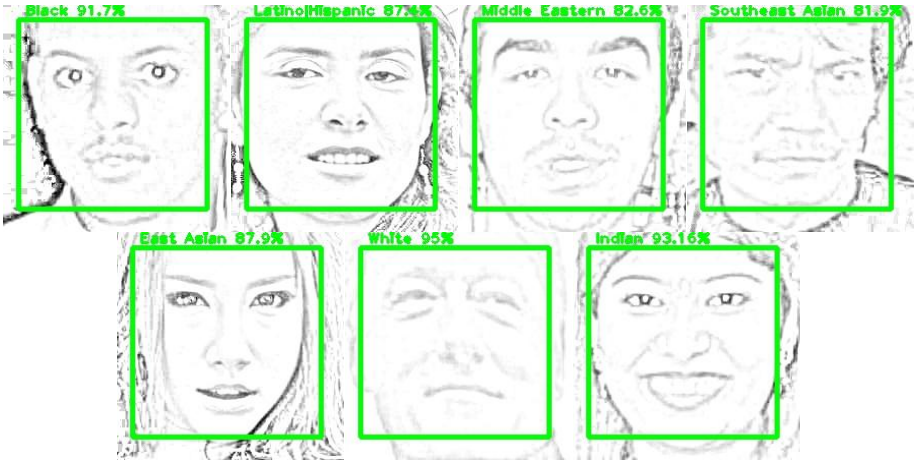


Fig. 20. Samples of the correct results of ethnic groups' face sketches classification and recognition.



Fig. 21. Samples of face sketches with failed face detection and their corresponding real photos from the FairFace dataset.

VI. Conclusion

This study proposes a new deep learning method for face sketch ethnic group classification and recognition. The method is based on the optimized VGG16 model. We used seven ethnic group sub-datasets for all ethnicities: White, Black, Indian, East Asian, Southeast Asian, Middle Eastern, and Latino. We tested our method on generated face sketches from the FairFace dataset using a combination of inversion, Gaussian blur, and fusion, and the results are very satisfactory. The accuracy is reaching above 94% and produced a low false negative rate. Our work is inspired by the recent successful methods that showed that a relatively mothered output VGG16 feature could be used to give good performance in a classification and recognition-based framework. To address the challenge of the lack of data and comparative models in the state of the art of the topic under study, the future work will include a decrease in the number of features used and the use of other competitive deep learning models.

References

[1] K.Suyemoto , M.Curley and S.Mukkamala, "What Do We Mean by "Ethnicity" and "Race"? A Consensual Qualitative Research Investigation of Colloquial Understandings" Genealogy journal, vol.4, no. 3, 10.3390,81, August 2020.

[2] S.Karen and A.Zisserman,"Very Deep Convolutional Networks for Large-Scale Image Recognition." CoRR abs/1409.1556, 2014.

[3] S. Chen, B. Mulgrew, and P. M. Grant. "A clustering technique for digital communications channel equalization using radial basis function networks", IEEE Trans. on Neural Networks, Vol. 4, pp. 570-578, July 1993.

[4] H. Keat Wong, I. Stephen and D. Keeble "The Own-Race Bias for Face Recognition in a Multiracial Society", Front. Psychol, Sec. Cognition Vol. 11, March 2020.

[5] N.Bhala, G.Curry, A.Martineau, C.Agyemang and R.Bhopal, "Sharpening the global focus on ethnicity and race in the time of COVID-19", The Lancet Journal, Vol. 395, no. 10238, pp. 1673-1676, May 2020.

[6] D..Pan, S.Sze, J. Minhas, M. Bangash, N. Pareek and P.Divall, "The impact of ethnicity on clinical outcomes in COVID-19: A systematic review", The Lancet Journal, eClinicalMedicine, Vol. 23, no. 100404, June 2020.

[7] K.Bhopal, " Gender, ethnicity and career progression in UK higher education: a case study analysis", Research Papers in Education, Vol. 35, no. 6, pp. 706-721, May 2020.

[8] L. Brown, U. Mitchell and J. Ailshire, "Disentangling the Stress Process: Race/Ethnic Differences in the Exposure and Appraisal of Chronic Stressors Among Older Adults", The Journals of Gerontology: Series B, Vol. 75, no. 3, pp. 650–660, March 2020.

[9] C. Rojas-Gaona, J. Hong and A. Peguero, "The significance of race/ethnicity in adolescent violence: A decade of review 2005–2015", J. Crim. Justice, Vol.46, pp.137–147, September 2016.

[10] K. Ounachad, M. Oualla, A. Souhar and A. Sadiq, "Face Sketch Recognition-An Overview", NISS '20: Proceedings of the 3rd International Conference on Networking, Information Systems & Security, Article no. 38, pp. 1-8, March 2020.

[11] K. Ounachad, M. Oualla, A. Souhar and A. Sadiq, "Structured learning and prediction in face sketch gender classification and recognition", International Journal of Computational Vision and Robotics, Vol. 10, no. 6, January 2020.

[12] A. Gite, S. Gethe, S. Gupta and S. Pagare, "Implementation of VGG16 Based learning for Facial Recognition", International Journal of Progressive Research in Engineering Management and Science (IJPREMS), Vol. 5, no. 4, April 2025.

[13] V. Nushi, R. Santos, H. Brkić, A. Oliveira, A. Francisco and C. Pereira, "Pre-trained VGG16 model for forensic dental age estimation", Egyptian Journal of Forensic Sciences, Vol. 15, no. 73, October 2025.

[14] M. Kocoń and S. Pawlukiewicz, "Age Estimation and Gender Classification from Facial Images", Applied Sciences Journal, Special Issue Applications of Deep Learning and Artificial Intelligence Methods: 3rd Edition, Vol. 15, no. 18, September 2025.

[15] S. Ozechi, "African Gender Classification Using Clothing Identification Via Deep Learning", arXiv:2503.00058, Vol. 1, March 2025.

[16] S. Makinist and G. Aydin, "Gender Classification Using Face Vectors: A Deep Learning Approach Without Classical Models", Information, Vol. 16, no. 7, June 2025.

[17] K. Ounachad, A. Souhar and A. Sadiq, "Fuzzy Hamming Distance and Perfect Face Ratios Based Face Sketch Recognition", IEEE 5th International Congress on Information Science and Technology, (CiSt), Marrakech, Morocco, pp. 317-322, October 2018.

[18] K. Ounachad, M. Oualla and A. Sadiq, "Face Sketch Recognition: Gender Classification Using Eyebrow Features and Bayes Classifier", Innovations in Smart Cities Applications, Vol. 4, pp. 809-819, February 2021.

[19] X. Zhu & Y. Li, "Application analysis of computer vision and image recognition based on improved VGG16 network", Discov Appl Sci, Vol. 7, no. 867, August 2025.

[20] R. Sewada and H. Goyal, "A Novel VGG-16 Adaptation for Multi-band Satellite Image Classification: Optimized Preprocessing and Class-specific Augmentation", Journal of Computational and Cognitive Engineering, pp. 1-12, May 2025.

[21] M. Genç & Y. Yalman "Aircraft Recognition Based on CNN Using Satellite Images", Journal of Innovative Science and Engineering. Vol. 9, no. 1, pp. 1-14, 2025.

[22] E. Addisu, T. Yirga, H. Yirga and A. Yehuala,"Transfer learning-based hybrid VGG16-machine learning approach for heart disease detection with explainable artificial intelligence", Front. Artif. Intell, Vol. 8, February 2025.

[23] F. Tanjim and S. Hamdy, "Application of VGG16 Transfer Learning for Breast Cancer Detection", Information, Vol. 16, no. 227, pp.1-20, March 2025.

[24] W. Jaimes, W. Arenas, H. Navarro and M. Altuve, "Detection of retinal diseases from OCT images using a VGG16 and transfer learning", Discover Applied Sciences, Vol. 7, February 2025.

[25] T. Arkan, A. Sugiharto, and H. Wibawa, "Modified VGG16 Architecture with Bayesian Optimization for Tomato Leaf Disease Classification", Jurnal Masyarakat Informatika, Vol.16, no. 2, pp. 162-173, June 2025.

[26] I. Sahputra, A. Ulva, B. Putri and C. Eviyanti, "Comparative Study of VGG16 and MobileNet Architectures for Rice Leaf Disease Classification Using CNN", Journal of Informatics and Telecommunication Engineering, Vol. 9, no. 1, pp. 303-314, July 2025.

[27] L. Chen, Z. Wang and H. Liu, "Steel-Reinforced Concrete Corrosion Crack Detection Method Based on Improved VGG16", Coatings, Vol. 15, no.641, pp. 1-14, May 2025.

[28] K. Ounachad, M. Oualla, A. Sadiq and A.souhar, "Face Sketch Recognition: Gender Classification and Recognition", International Journal of Psychosocial Rehabilitation, Vol. 24, no. 03, pp. 1073-1085 February 2020.

[29] K. Ounachad, M. Oualla, A. Sadiq and A.souhar, "Golden Ratio and Its Application to Bayes Classifier Based Face Sketch Gender Classification and Recognition", International Journal of Emerging Trends in Engineering Research, Vol. 8, no. 7, pp. 3538-3545, July 2020.

[30] K. Ounachad, M. Oualla, A. Sadiq and A.souhar, "Human Face (Sketch/Photo) Age Group Estimation and Classification Using Perfect Face Ratios and Levenshtein Distance", International Journal of Emerging Trends in Engineering Research, Vol. 8, no. 7, pp.3191-3201, July 2020.

[31] K. Ounachad, M. Oualla, A. Sadiq and A.souhar, "Facial Emotion Recognition Using Average Face Ratios and Fuzzy Hamming Distance ", Journal of Automation Mobile Robotics & Intelligent Systems, Vol.14, no. 4, pp. 37-44, March 2020.

[32] K. Ounachad, M. Oualla and A. Sadiq, "Geometric Feature Based Facial Emotion Recognition", International Journal of Advanced Trends in Computer Science and Engineering, Vol. 9, no. 3, pp. 3417 -3425, May – June 2020.

[33] K. Ounachad, M. Oualla, A.souhar and A. Sadiq, "Structured Learning and Prediction in Facial Emotion Classification and Recognition", International Journal of Engineering and Advanced Technology, Vol. 9, no. 4, pp. 152-160, April 2020.

[34] K.Karkkainen, & J.Joo, " FairFace: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation". In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 1548–1558.January 2021.

[35] K. Karkkainen, & J. Joo, "FairFace: Face Attribute Dataset for Balanced Race, Gender, and Age", arXiv preprint arXiv:1908.04913, 2019.

[36] T.K. Bahiru, N. T. Sinshaw, T. H. Moges and D. K. Singh, "Auditing and Mitigating Bias in Gender Classification Algorithms: A Data-Centric Approach", arXiv preprint arXiv:2510.17873, 2025.

[37] J. Xu, C. Liu, X. Tan et al., "General information metrics for improving AI model training efficiency", Artif Intell Rev, Vol. 58, no.289, July 2025.

[38] C. Lehmann & Y. Paromau, "Quantifying Uncertainty and Variability in Machine Learning: Confidence Intervals for Quantiles in Performance Metric Distributions ", arXiv:2501.16931 [cs.LG], January 2025.

[39] Z. Ba Alawi, "A Comparative Survey of PyTorch vs TensorFlow for Deep Learning: Usability, Performance, and Deployment Trade-offs", arXiv:2508.04035 [cs.LG], August 2025.

[40] S. Abd, H. Mangi, A. Kadhum, R. kazim, B.Ali and F. AbdAlkarim, "Efficient face detection using viola-jones and neural networks: a comparative study", AJIEA, vol. 1, no. 1, p. 13, June 2025.

[41] S.A. Wijaya, T.S. Famuji, M.A. Mu'min, Y. Safitri, N. Tristanti, A. Dahmani, Z. Driss, A.Sharkawy and R. Sabur, "Trends and Impact of the Viola-Jones Algorithm: A Bibliometric Analysis of Face Detection Research (2001-2024)", Scientific Journal of Engineering Research, Vol. 1, no. 1, pp. 33-42, February 2025.

[42] S. Husain, & E. Abolkasim, "Estimating the Number of People in Digital Still Images Based on Viola-Jones Face Detection Algorithms", African Journal of Advanced Pure and Applied Sciences, Vol. 3, no. 2, pp. 146–154. June 2024.

Integrating Japanese Festival Traditions into Computational Thinking: A Culturally Responsive Approach to Micro:Bit-based Physical Computing

Daiki Sugiyama[1], M. Fahim Ferdous Khan[2], Ken Sakamura[3]
Faculty of Information Networking for Innovation and Design (INIAD), Toyo University, Tokyo, Japan[1, 2]
INIAD cHUB, Toyo University, Tokyo, Japan[3]

***Abstract*—The current pre-tertiary educational landscape in Japan is defined by a significant paradox. While technological infrastructure has reached unprecedented levels of saturation through the Global and Innovation Gateway for All (GIGA) School initiative, student engagement and qualitative learning outcomes in computational disciplines have shown signs of stagnation. This research addresses this disparity by proposing a framework grounded in Culturally Responsive Teaching (CRT) and Culturally Responsive Computing (CRC), utilizing the traditional Japanese festival game of *Shateki* (target shooting) as a primary pedagogical vehicle. By leveraging the BBC micro:bit and a modular physical computing architecture, this study demonstrates how situating abstract programming concepts, such as sequential logic, conditional branching, and signal modulation, within familiar "vernacular" and "heritage" cultural contexts can foster intrinsic motivation and improve academic achievement among K-12 students. Through a pilot evaluation conducted during a community festival, this study provides quantitative evidence that CRT-based interventions not only enhance technical proficiency but also bolster a sense of student agency and cultural belonging. The results suggest that the success of modern educational technology initiatives depends less on hardware distribution and more on the pedagogical translation of technical content into culturally meaningful experiences.**

***Keywords*—Culturally responsive teaching (CRT); Culturally Responsive Computing (CRC); computational thinking; micro:bit; K-12 education; physical computing; STEM education**

I. Introduction

The Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT) launched the GIGA (Global and Innovation Gateway for All) School Program with the ambitious goal of providing a high-speed communication environment and a digital device for every student in primary and secondary schools across the nation [1]. This initiative was designed to eliminate educational gaps caused by socio-economic disparities and to prepare the next generation for Society 5.0, where cyber and physical spaces are highly integrated. Statistical indicators confirm the rapid success of the infrastructure rollout; student computer ratios have improved dramatically, and wireless LAN availability in classrooms has surged from roughly 40% to over 90% in just a few years [2]. Despite this technical achievement, the impact on learning remains contentious.

Recent data from the National Assessment of Academic Ability reveals a troubling trend. Between the 2024 and 2025 academic years, average correct response rates in core subjects like Japanese and Mathematics have declined across both elementary and junior high schools [2]. In mathematics, specifically, elementary school scores dropped from 63.6% to 58.2%, while junior high school scores fell from 53.0% to 48.8% [3]. These declines suggest that the mere presence of technology in the classroom does not guarantee improved cognition. In fact, many reports indicate that the GIGA School terminals are frequently used for non-educational entertainment, such as gaming or streaming services, which can detract from focused learning.

The root of this problem lies in the disconnect between "schooling" – the institutionalization of learning within a rigid, often Western-centric framework – and "education," which should empower students to transform their environment [4],[5]. When programming is taught as a series of abstract, context-less syntax exercises, students often find the material boring or difficult. This is particularly evident in the K-12 sector, where students' attention spans and motivation are deeply tied to the perceived relevance of the curriculum. The current failure to bridge the gap between high hardware accessibility and student interest necessitates a shift toward Culturally Responsive Teaching [6],[7].

In this respect, the primary contributions of this research are as follows.

- We developed a modular physical computing curriculum centered on the Japanese cultural heritage of *Shateki,* or target shooting game (Fig. 1), to teach the core principles of computational thinking.
- We implemented an asset-based pedagogical framework that validates students' community knowledge as a foundation for technical mastery.
- With a pilot study at a community festival, we quantitatively evaluated the impact of culturally situated learning on student motivation and proficiency in programming logic.
- We performed a comparative analysis of educational hardware platforms to determine the suitability of micro:bit for CRT-based interventions.

The remainder of this study is organized as follows. Section II presents a brief literature review. Section III discusses the cultural significance of Japanese summer festivals (*Matsuri*) and the traditional game of *Shateki* as pedagogical tools. Section IV outlines the system architecture and hardware design, describing the use of the BBC micro:bit and modular sensors. Section V explains the pedagogical strategy employed, which centers on sequential execution, conditional branching, and debugging. Section VI presents the experimental evaluation and results from a pilot study conducted at a community festival. Section VII provides an impact analysis of the curriculum through the lenses of identity affirmation and student agency. Section VIII offers a comparative analysis of educational hardware platforms commonly used in Japanese schools. Section IX discusses the lessons learned in the *Shateki* project and outlines future research directions. Finally, Section X concludes the study.



Fig. 1. Photograph of a typical Shateki (Japanese shooting game) [Public domain] via https://www.meijimura.com.

II. Literature Review

To situate the proposed framework within the broader discourse of Computer Science Education (CSE), this section examines existing literature across three converging domains: Culturally Responsive Teaching (CRT), Culturally Responsive Computing (CRC), and Physical Computing. The synthesis of these fields suggests that while tangible interfaces improve the cognitive grasp of abstract concepts [8],[9], and cultural contextualization improves learner motivation [10],[11], there remains a scarcity of rigorous studies combining both methodologies within the specific cultural heritage contexts of East Asia.

A. Culturally Responsive Teaching (CRT) as a Pedagogical Foundation

Culturally Responsive Teaching (CRT) serves as the overarching pedagogical architecture for this study. Defined foundationally by Gay [12] and Ladson-Billings [13], CRT is a pedagogy that recognizes the importance of including students' cultural references in all aspects of learning. It posits that academic achievement improves when instruction is filtered through the students' own cultural experiences and frames of reference.

In the context of STEM education, the disconnect between a student's home culture and the culture of the science classroom is often cited as a primary barrier to engagement. Research by Villegas and Lucas [14] emphasizes that sociocultural consciousness is a prerequisite for teachers to bridge this gap. However, traditional CRT literature has largely focused on the humanities and social sciences. Only recently has the scope expanded to technical disciplines, where the "culture-free" assumption of mathematics and logic is being challenged. By validating "vernacular" knowledge – such as the rules of a traditional festival game – educators can lower the affective filter, making rigorous academic content more accessible to young learners who might otherwise view themselves as outsiders to the domain of technology.

B. Culturally Responsive Computing (CRC) and Computational Thinking

Culturally Responsive Computing (CRC) represents the intersection of CRT and computer science education. Eglash et al. [15] have been instrumental in defining this space through "ethnocomputing," demonstrating that indigenous practices – such as cornrow braiding or Navajo weaving – contain inherent computational algorithms. By explicitly mapping these cultural artifacts to programming concepts, CRC transforms heritage from a passive backdrop into an active cognitive resource.

Scott et al. [16] further argue that CRC is essential for equitable participation in computing. Their work indicates that when curricula reflect the identities of underrepresented youth, there is a marked increase in both self-efficacy and the acquisition of Computational Thinking (CT) skills. CT, as defined by Wing [17] and elaborated by Brennan and Resnick [18], involves solving problems, designing systems, and understanding human behavior by drawing on the concepts fundamental to computer science.

However, a significant portion of CRC literature is situated within North American contexts, focusing on African American or Latino cultural matrices [19]. There is a distinct paucity of literature examining CRC frameworks within the Japanese educational system, particularly regarding how distinct cultural assets like *Matsuri* (festivals) can be leveraged to address the motivation crisis identified in the GIGA School era. This study aims to extend the CRC discourse by validating the *Shateki* game as a computational artifact capable of teaching conditional logic and algorithmic sequencing.

C. Physical Computing and Cognitive Load in Young Learners

The third dimension of this research is Physical Computing: the design of interactive physical systems by the use of software and hardware that can sense and respond to the analog world. This approach aligns with Papert's theory of Constructionism [20], which posits that learning is most effective when the learner is engaged in constructing a public entity. Recent studies strongly support the efficacy of physical computing tools, such as the BBC micro:bit, in fostering CT. Sentence et al. [21] found that physical computing devices provide immediate feedback loops that are crucial for novice

programmers. When a student programs a physical device to "shoot" a target, the abstract code becomes a tangible reality. This tangibility reduces cognitive load; the concept of a "variable" is no longer just a memory space but a score counter on a display; a "conditional statement" is no longer just syntax but the visible act of a point increment when the target is hit.

Furthermore, research by Hodges et al. [22] suggests that physical computing is particularly effective in collaborative settings, fostering "computational participator" behaviors. The tangible nature of devices allows for shared attention and collaborative debugging, which are critical components of the social learning environment found in festival preparations.

D. Synthesis: The Intersection of Culture and Tangibility

While the individual benefits of CRC and Physical Computing are well-documented, the synergy between them remains underexplored in current literature. Most CRC interventions rely on screen-based simulations (e.g., culturally relevant Scratch projects), while most physical computing interventions utilize generic projects (e.g., traffic lights, weather stations) that lack deep cultural resonance.

This research posits that the intersection of these domains offers a unique force multiplier for educational outcomes. By combining the tangible scaffolding of physical computing with the motivational scaffolding of CRC, we propose a holistic framework. The *Shateki* project utilizes the micro:bit not merely as a generic controller, but as a modern instrument to re-enact a centuries-old tradition, thereby validating the students' past while equipping them for the future.

III. The Cultural Significance of Matsuri and Shateki in Japan

Japanese summer festivals, or Matsuri, are pivotal community events that combine traditional religious elements with modern social celebration [23]. Originally rooted in agrarian prayers for harvest and protection from pestilence, they have transitioned into vibrant spaces where residents participate with fervor and actively engage in community bonding [24]. Central to the Matsuri experience are the Yatai (stalls) that offer a variety of foods and traditional games. These games are not merely recreational; they are cultural resources that teach focus, patience, and skill while fostering a sense of shared belonging.

Among these activities, Shateki (target shooting) is particularly iconic. In a traditional Shateki stall, a player uses an air rifle, typically firing cork bullets, to attempt to knock prizes off a tiered display (Fig. 1). The prizes range from inexpensive snacks and candies to high-value toys or character goods [25]. The cultural significance of Shateki is deep; for adults, it represents a nostalgic link to their childhood, and for children, it is a thrilling test of aim and nerve. The mechanics of the game are intuitively understood by Japanese citizens of all ages, making it an ideal low-floor entry point for teaching computational logic.

By framing a programming lesson around building a Shateki game, educators can tap into "funds of knowledge" that students already possess [4]. Instead of learning about sensors and infrared light in a vacuum, students learn how to "build a festival" using digital tools. This approach aligns with the CRC tenet of "asset building," where students' existing cultural expertise is validated and used as a bridge to new technical knowledge [26].

IV. System Architecture and Hardware Design

To implement the digital Shateki experience, this research utilizes the BBC micro:bit platform. The micro:bit is a microcontroller designed specifically for education, featuring integrated sensors, an LED matrix, and physical input-output pins [27]. To program the BBC micro:bit, Microsoft MakeCode [28] – a free, open-source web-based platform – is used. The platform supports block-based programming, similar to Scratch. Such block-based programming has been proven to be useful for early-stage learners [29]. For this study, a modular system was designed to simulate the physical interactions of a shooting gallery. As shown in Fig. 2, the hardware components include the micro:bit unit, a workshop module, vibration sensors, and infrared (IR) LED/receiver modules.

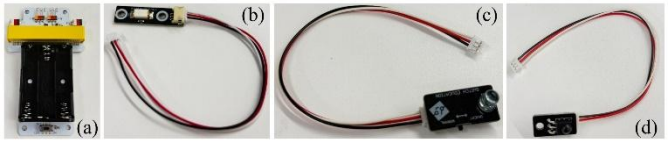


Fig. 2. Hardware components: (a) Workshop module for micro:bit, (b) Vibration sensor for micro:bit, (c) Infrared LED module for micro:bit, (d) Infrared receiver module for micro:bit.

A. Physical Computing Modules

The system is designed to be modular, allowing students to understand the relationship between different hardware units and their specific roles in a distributed computing environment. Table I provides a breakdown of the hardware components used in the *Shateki* system. The role and purpose of each component are also summarized in Table I.

TABLE. I. Components used in Shateki Game

Module Name	Technical Role	Purpose in *Shateki* Game
micro:bit v2.2	Central procession unit	Manages logic, variablers and display
Workshop module	Power and I/O distribution	Facilitates remote sensor connectivity
Vibration sensor	Anlaog input (Piezoelectric)	Detects projectile impact on the target
Infrared LED	Signal output (38.5 kHz)	Creates a digital tripwire for boundaries
Infrared receiver	Signal detection	Monitors for interruptions in the IR beam

As shown in Fig. 3, the system architecture is bifurcated into two primary subsystems: the Target Function and the Boundary Function. In a traditional *Shateki* gallery, the player must stay behind a line; if they cross it, the attempt is void. By digitizing this rule, students learn about sensor-based automation. The Target Function uses a vibration sensor to detect when the target has been struck, while the Boundary

Function uses an IR beam to detect if the player's foot has crossed the designated firing line.

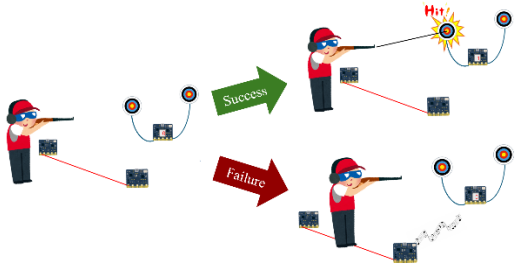


Fig. 3. Overview of the Shateki game with success and failure cases.

B. Hardware Connectivity and Constraints

As shown in Fig. 4, the completed hardware assembly integrates these modules into a cohesive playset. The target function is physically separated from the infrared boundary units, necessitating a multi-micro:bit setup. In this configuration, one micro:bit acts as the target scorekeeper, while two others manage the infrared transmission and reception. This setup is intentional, as it introduces students to the concept of hardware independent of a single "brain," mirroring real-world IoT (Internet of Things) architectures.

The technical challenge for students lies in the analog-to-digital conversion (ADC) process. For the vibration sensor, an analog value of 0 indicates an impact, whereas 1 indicates a steady state. For the IR receiver, the logic is similar: a detected beam returns 1, while an interruption (a broken beam) returns 1. Navigating these inverted logic states provides a fertile ground for developing critical thinking and problem-solving skills, as students must often debug their assumptions about how "on" and "off" signals should behave.

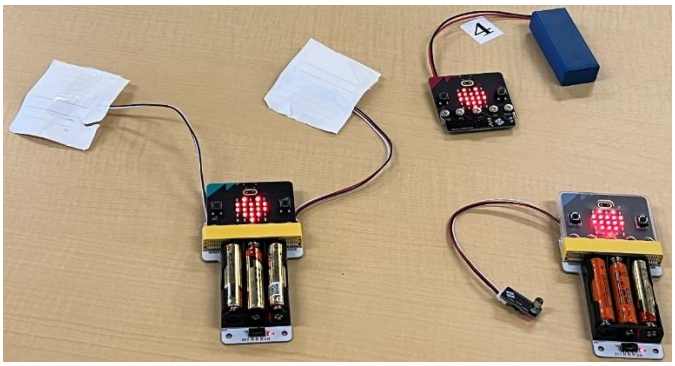


Fig. 4. Hardware setup of the Shateki game.

V. Pedagogical Strategy

The instructional design of the Shateki project is centered on the three fundamental principles of programming: sequential execution, conditional branching, and iteration. The pedagogical innovation here is the use of debugging as a primary learning activity. Research suggests that providing students with broken but mostly functional code encourages deeper analysis than starting from a blank screen [30].

A. Programming the Target Logic

Students are provided with an initial code block for the Target Function that contains a logical bug. As shown in Fig. 5, the initial code incorrectly maps the sensor input to a pin that is not physically connected to the workshop module. Students must trace the physical jumper wires to the correct pin (typically P0) and update the code accordingly. As shown in Fig. 6, the correct logic involves an "if-then" statement: *if* the vibration sensor detects a hit (analog_value=0), *then* increment the score variable by 1. This exercise teaches conditional branching in a way that is immediately visible through the 5x5 LED display, which shows the score increasing in real-time. Such instant feedback improves program comprehension [31].

B. Signal Modulation and Boundary Function

The Boundary Function introduces more sophisticated concepts. The IR LED must be modulated to a specific frequency (approximately 38.5 kHz) for the receiver to detect it reliably. On the micro:bit, this is achieved by simulating a frequency through rapid pulse repetitions. As shown in Fig. 7(a), the sample code provided to students has the "repeat" value set to 0, meaning no signal is sent. Students learn that for communication to occur, the transmitter and receiver must agree on a "signal" presence. As shown in Fig. 7(b), setting the loop to 100 repetitions creates a stable enough frequency for detection.



Fig. 5. Initial incorrect program (target logic).



Fig. 6. Program with corrected if-then logic.

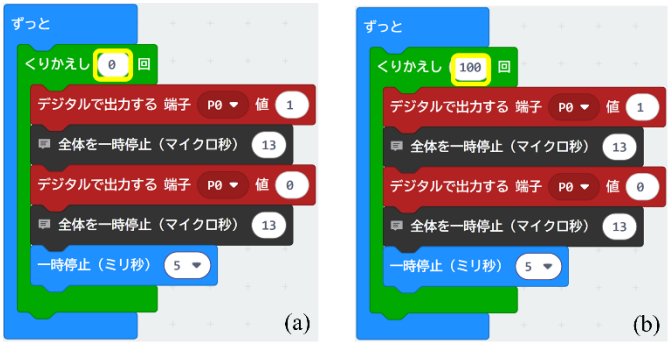


Fig. 7. Program for sending signal: (a) Incorrect, (b) Correct.

On the receiving side, students must manage the inverted logic of infrared detection. As shown in Fig. 8, the initial code might have a threshold value (e.g., 5) that does not match the binary nature of the IR receiver output. By testing the system and viewing the data on their screens, students identify that they must check for a value of 0 to detect a "broken" beam. As shown in Fig. 9, the corrected code triggers a warning sound and displays a danger icon on the LED matrix when the line is crossed. This creates an immediate cause-and-effect relationship between code and physical action, which is a key driver of engagement in physical computing [32].

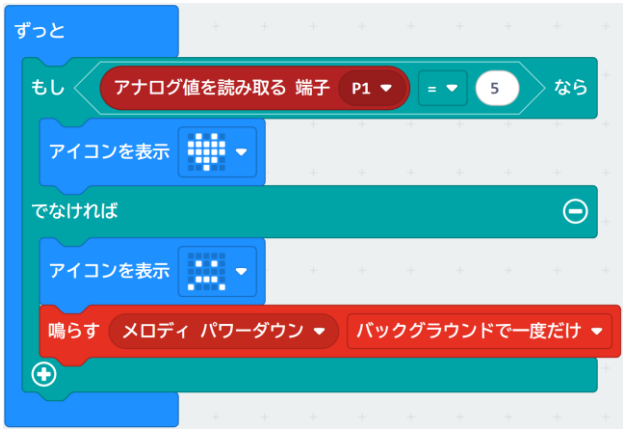


Fig. 8. Incorrect receiver program.

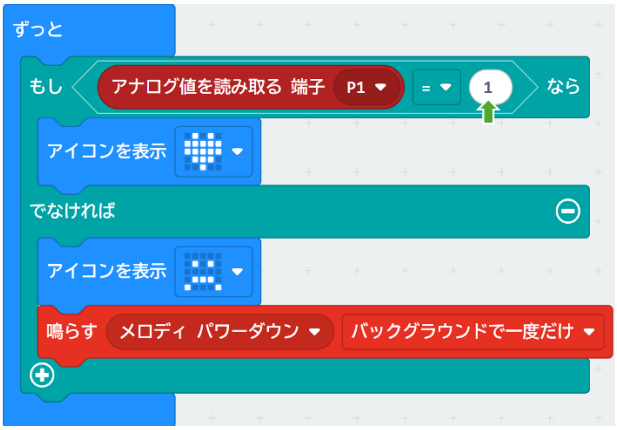


Fig. 9. Corrected receiver program.

VI. Experimental Evaluation and Results

The efficacy of this culturally responsive Shateki curriculum was evaluated during the Akabanedai Festival in Tokyo, a community event that provided a naturalistic environment for testing. The participants included 31 students, primarily in the elementary and junior high school demographics, who engaged in the programming and gaming activities. Evaluation was conducted through digital questionnaires and a confirmation test (a quiz to check comprehension of basic programming concepts) administered after the experience.

A. Participant Profile and Prior Experience

The majority of the participants were within the K-12 target group. Interestingly, the survey data revealed that many students had very little prior exposure to programming. Over 60% of respondents indicated they had "never" or "hardly ever" received programming education. Furthermore, usage of the micro:bit was almost zero before the festival, with nearly all participants being first-time users.

This data underscores the importance of the CRT approach. When students are blank slates with regard to technical skills, the cultural familiarity of the Shateki game acts as a critical scaffolding tool, preventing the technological intimidation that can occur with abstract coding tasks [33].

B. Quantitative Analysis of Motivation and Satisfaction

The post-experience survey utilized a 5-point Likert scale to measure various aspects of student engagement and understanding. The results are summarized in Table II.

TABLE. II. Result of Questionnaire Survey

Metric	Avg. score	SD
Enjoyment of the experience	4.77	0.57
Overall satisfaction	4.70	0.60
Desire to try again	4.57	0.77
Increase in programming motivation	4.30	0.84
Desire to recommend to others	4.27	1.01
Sense of achievement/completion	4.03	1.00
Understanding of programming logic	3.87	1.01

The exceptionally high score for Enjoyment (4.77) and Overall Satisfaction (4.70) indicates that the festival context was highly effective at creating a positive learning environment. More importantly, the score for Programming Motivation (4.30) confirms that the CRT framework successfully transformed a difficult technical subject into an appealing one. Fig. 10 shows a collage of students programming and trying the game.

The quiz results, as shown in Fig. 11, provide evidence of academic achievement. Among the 25 test takers, the score distribution was heavily weighted toward the high end, with many students demonstrating mastery of the three principles of programming and sensor logic. This result directly contradicts the national trend of declining academic scores in mathematics and science, suggesting that when content is "culturally situated," students can achieve high levels of mastery even in short-duration interventions.



Fig. 10. Students programming and testing the Shateki game.

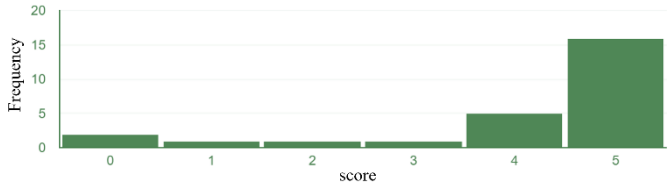


Fig. 11. Distribution of quiz scores.

VII. Impact Analysis

The success of the Shateki project can be explained through several key CRT mechanisms: identity affirmation, immediate feedback, and student agency.

A. Identity Affirmation and Cultural Capital

By using a festival game, the curriculum validates students' cultural capital – the knowledge and skills they have gained from their families and communities [26],[34]. Instead of the classroom being a place where their external life is ignored, it becomes a place where their cultural heritage is digitized and celebrated [35]. This reduces cultural alienation and increases a sense of belonging, which is a prerequisite for sustained academic engagement in STEM fields [33].

In Japan, where the GIGA school initiative has standardized the digital experience, there is a risk of cultural blindness, where the unique local and national traditions of students are overlooked in favor of generic globalized content [36],[37]. The Shateki game re-injects "Japaneseness" into the computer science curriculum, allowing students to see computing as a tool for cultural sustainability – preserving the old through the new [38].

B. Immediate Feedback and Physicality

The micro:bit's physical nature provides immediate feedback that is qualitatively different from screen-based programming. When a student fixes a line of code and a physical target starts counting scores, or an infrared beam starts sounding an alarm, the abstraction of the computer becomes concrete [33]. Students were highly motivated to use the micro:bit for other activities after the session, suggesting that the "physicality" of the experience created a lasting interest in hardware-software integration.

This immediacy is particularly important for younger learners who may find traditional GIGA-school terminals boring or distracting. The tactile nature of the workshop modules and sensors keeps students on-task and focused on the problem-solving process.

C. Empowering Student Agency

The debugging approach used in this research promotes student agency, i.e., the feeling that the learner is a producer of knowledge rather than a passive consumer [38]. When students fixed the bugs in the Shateki code, they were not just following a tutorial; they were, in essence, reclaiming the system and making it work according to their understanding of the game's rules. This aligns with the CRT principle of critical consciousness, where students learn to manipulate and control the technology in their environment [38].

VIII. Comparative Analysis of Educational Hardware

To understand why the micro:bit was chosen for this culturally responsive intervention, it is necessary to compare it with other prevalent hardware platforms in Japanese education. A comparative evaluation of such educational hardware is presented in Table III.

Arduino [39] and Raspberry Pi [40] offer high extensibility but have a steep learning curve involving complex wiring and OS management, which can become a barrier for beginner students. LEGO Education [41] is highly refined but prohibitively expensive for widespread GIGA school implementation, potentially widening the socio-economic gap CRT seeks to close. Robot-based toys like Sphero [42] and Ozobot [43] are engaging but often act as black boxes where the internal sensor logic is hidden from the student.

The micro:bit strikes the ideal balance. Its low cost and built-in features make it accessible to underserved schools, and its open pin architecture allows for the modular Shateki design. Furthermore, the total cost for the Shateki set used in this study was around 26,800 JPY (around 170 USD) per complete set. While this may seem high, the micro:bit itself is the only component needed for each student (approximately 3,400 JPY or around 20 USD), with sensors shared in a station-based learning model. This ensures that the technology is accessible and relevant for all learners, a core requirement of the CRT Areas of Opportunity framework [44].

TABLE. III. Comparison of Hardware Typically Used in Physical Computing Education

Platform	Operability	Safety	Approx. Cost (in JPY as of Dec. 2025)	Cultural Adaptability	Feedback Immediacy
BBC micro:bit	High	High	3,400	High	Excellent
Arduino	Low	Medium	3,000	Medium	Medium
Raspberry Pi	Low	High	6,000	Low	Low
LEGO Education	High	High	25,000	Medium	High
Sphero / Ozobot	High	High	15,000	Low	High

IX. Lessons Learned and Future Work

Despite the success of the Shateki project, several challenges remain for broader implementation in Japanese schools.

A. Language and Technical Jargon

Feedback from the Akabanedai Festival indicated that for some elementary school students, the technical language used in the coding blocks was too difficult. This highlights a critical need for linguistic responsiveness in CRT. A culturally responsive teacher must be a pedagogical translator, converting abstract terms like variable, analog threshold, and signal modulation into metaphors that resonate with the students' everyday language and festival experiences [33].

B. Calibration and Differentiation

The survey results showed a spread in the "Sense of achievement" and "Understanding of programming logic" scores, with standard deviations around 1.0. This suggests that one-size-fits-all instruction is ineffective even within a culturally responsive framework. Some students found the tasks too easy, while others struggled with the logic of the infrared receiver. To address this, future curricula should offer tiered challenges, for example, allowing advanced students to add sound effects or timers to their Shateki games, while ensuring beginners can complete the core vibration and boundary logic.

C. Sustaining Interest Beyond the Novelty Effect

While the fun of the festival game provides an initial hook (a phenomenon known as the novelty effect [45]), the ultimate goal of CRT is to build long-term computational identity [26]. This requires moving from single-experience workshops to a spiral curriculum where the micro:bit is used across different subjects, such as using the vibration sensor in a science class to measure tectonic movements or using the LED matrix in a math class to visualize data. The Shateki project should be seen as the gateway that proves to students they belong in the world of computing.

X. Conclusion

This research has demonstrated that the pervasive disengagement and declining academic outcomes observed in the wake of the GIGA School initiative can be mitigated by adopting a Culturally Responsive Teaching framework. The micro:bit-based Shateki shooting game serves as a powerful evidence-based model for this approach. By situating the abstract principles of programming within the joyous and familiar context of a Japanese summer festival, the project achieved high levels of student satisfaction, increased learning motivation, and measurable cognitive gains in programming logic.

The success of the Shateki project highlights that effective technology integration is not a function of the number of devices in a classroom, but the degree to which those devices are used to celebrate and extend students' cultural identities. When students are empowered to "build their own festival" using sensors and code, they move from being passive consumers of Western-centric software to being creative innovators with their own cultural voice.

For Japanese education to fulfill the promise of the GIGA School era, it must move beyond hardware distribution toward a pedagogy that values vernacular and heritage culture as essential components of STEM success. By making Culturally Responsive Computing the central theme of computer science education, we can ensure that every student – regardless of their socio-economic background – can see themselves as a capable and innovative citizen of the digital future. The Shateki experience proves that the path to high-tech mastery often begins with a very traditional, very human connection.

References

[1] H. Itakura, "Improving the Quality of Education Through the GIGA School Program," Ministry of Education, Culture, Sports, Science and Technology (MEXT), Sep. 14, 2022. (In Japanese) [Online]. Available: https://www.mext.go.jp/content/20220914-mxt_kokusai-000025012_2Itakura.pdf. [Accessed: Jan. 25, 2026].

[2] Ministry of Education, Culture, Sports, Science and Technology (MEXT), "Results of the FY2024 National Assessment of Academic Ability (Summary)" MEXT, Aug. 2024. (In Japanese) [Online]. Available: https://www.mext.go.jp/content/20240808-mxt_syokyo02-000037464_5.pdf. [Accessed: Jan. 25, 2026].

[3] National Institute for Educational Policy Research (NIER), "FY2025 National Assessment of Academic Ability: Reports and Survey Results," July 2025. (In Japanese) [Online]. Available: https://www.nier.go.jp/25chousakekkahoukoku/. [Accessed: Jan. 25, 2026].

[4] B. Escudero, "How to Practice Culturally Relevant Pedagogy," Teach For America, Jan. 6, 2019. [Online]. Available: https://learn.artsintegration.com/wp-content/uploads/2020/04/teachforamerica.org-How-to-Practice-Culturally-Relevant-Pedagogy.pdf. [Accessed: Jan. 25, 2026].

[5] S. Guha and S. Alkayfee, "Culturally Responsive Teaching, STEM Careers, Minority Students, Student Engagement, Education," J. Crit. Race Ethnic Stud., vol. 2, no. 2, p. 3, 2025.

[6] M. Tedre, E. Sutinen, E. Kähkönen, and P. Kommers, "Ethnocomputing: ICT in cultural and social context," Commun. ACM, vol. 49, no. 1, pp. 126–130, Jan. 2006.

[7] R. Eglash, J. E. Gilbert, and E. Foster, "Toward Culturally Responsive Computing Education," Communications of the ACM, vol. 56, no. 7, pp. 33–36, July 2013, doi: 10.1145/2483852.2483864.

[8] M. F. F. Khan, D. M. Hadisumarto and K. Sakamura, "A Tangible-Tool-Based Lesson Plan on Cipher Key Exchange Protocol for Early-Stage Learners," 2022 IEEE Global Engineering Education Conference (EDUCON), Tunis, Tunisia, 2022, pp. 620-627, doi: 10.1109/EDUCON52537.2022.9766379.

[9] T. Katanosaka, M. F. Ferdous Khan and K. Sakamura, "Introduction to Programming with Micro:Bit by Solving Real-Life Problems: Evaluation of Learners' Self-Efficacy," 2025 1st International Conference on Consumer Technology (ICCT-Pacific), Matsue, Shimane, Japan, 2025, pp. 1-4, doi: 10.1109/ICCT-Pacific63901.2025.11012809.

[10] T. Katanosaka, M. F. Ferdous Khan and K. Sakamura, "Improving Self-Efficacy Through Experiential Learning of IoT Programming," 2025 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), Macao, 2025, pp. 1-8, doi: 10.1109/TALE66047.2025.11346629.

[11] T. Katanosaka, M. Khan, and K. Sakamura, "PhyGame: An Interactive and Gamified Learning Support System for Secondary Physics Education," Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 6, Jun. 2024

[12] G. Gay, Culturally Responsive Teaching: Theory, Research, and Practice. New York, NY: Teachers College Press, 2000.

[13] G. Ladson-Billings, "Toward a theory of culturally relevant pedagogy," American Educational Research Journal, vol. 32, no. 3, pp. 465-491, 1995.

[14] A. M. Villegas and T. Lucas, “Preparing culturally responsive teachers: Rethinking the curriculum,” Journal of Teacher Education, vol. 53, no. 1, pp. 20-32, 2002.

[15] R. Eglash, J. E. Gilbert, V. Taylor, and S. R. Geier, “Culturally responsive computing in urban, after-school contexts: Two approaches,” Urban Education, vol. 48, no. 5, pp. 629-656, 2013.

[16] K. A. Scott, K. M. Sheridan, and K. Clark, “Culturally responsive computing: A theory revisited,” in Learning, Media and Technology, vol. 40, no. 4, pp. 412-436, 2015.

[17] J. M. Wing, “Computational thinking,” Communications of the ACM, vol. 49, no. 3, pp. 33-35, 2006.

[18] K. Brennan and M. Resnick, “New frameworks for studying and assessing the development of computational thinking,” in Proceedings of the 2012 Annual Meeting of the American Educational Research Association, Vancouver, Canada, 2012.

[19] J. Morales-Chicas, M. Castillo, I. Bernal, P. Ramos, and B. L. Guzman, “Computing with relevance and purpose: A review of culturally relevant education in computing,” Int. J. Multicult. Educ., vol. 21, no. 1, pp. 125–155, 2019.

[20] S. Papert, Mindstorms: Children, Computers, and Powerful Ideas. New York, NY: Basic Books, 1980.

[21] S. Sentance, A. Csizmadia, "Computing in the curriculum: Challenges and strategies from a teacher’s perspective," Education and Information Technologies, vol. 22, no. 2, pp. 469-495, 2017.

[22] S. Hodges, S. Sentance, J. Finney, and T. Ball, "Physical computing: A key element of modern computer science education," Computer, vol. 53, no. 4, pp. 20-30, Apr. 2020, IEEE Computer Society.

[23] B. Reynolds, Japanese Celebrations for Children: Festivals, Holidays and Traditions. North Clarendon, VT, USA: Tuttle Publishing, 2023.

[24] M. Keliyan, “Matsuri as a cultural resource for studying Japanese local community,” in Proc. 30th EAJRS Conf. (Sofia 2019), Sofia, Bulgaria, Sept. 2019. [Online]. Available: https://www.eajrs.net/matsuri-cultural-resource-studying-japanese-local-community-2019. [Accessed: Jan. 25, 2026].

[25] “Japanese Summer Festivals,” CDJapan, 2023. [Online]. Available: https://www.cdjapan.co.jp/feature/Japan-Summer-Festivals. [Accessed: Feb. 21, 2026].

[26] H. C. Leonard and S. Sentance, "Culturally-relevant and responsive pedagogy in computing: A Quick Scoping Review," International Journal of Computer Science Education in Schools, vol. 5, no. 2, pp. 3–13, Dec. 2021.

[27] Micro:bit Educational Foundation, “Micro:bit Educational Foundation | micro:bit,” 2026. [Online]. Available: https://microbit.org/. [Accessed: Jan. 25, 2026].

[28] “Microsoft MakeCode for micro:bit,” Microsoft, 2024. [Online]. Available: https://makecode.microbit.org/. [Accessed: Feb. 21, 2026].

[29] M. F. F. Khan, K. Eguchi and K. Sakamura, “Instilling AI Thinking into Young Pupils Through Experiential Learning,” in Proc. 9th Int. Conf. Education and E-Learning (ICEEL), Nov. 2025.

[30] C. Tikva and E. Tambouris, “The effect of scaffolding programming games and attitudes towards programming on the development of Computational Thinking,” Education and Information Technologies, vol. 28, no. 6, pp. 6845–6867, June 2023, doi: 10.1007/s10639-022-11465-y.

[31] H. Yaseen et al., “The Impact of Adaptive Learning Technologies, Personalized Feedback, and Interactive AI Tools on Student Engagement: The Moderating Role of Digital Literacy,” Sustainability, vol. 17, no. 3, p. 1133, Jan. 2025. doi: 10.3390/su17031133.

[32] C. C. Chung and S. J. Lou, “Physical Computing Strategy to Support Students’ Coding Literacy: An Educational Experiment with Arduino Boards,” Applied Sciences, vol. 11, no. 4, p. 1830, Feb. 2021. doi: 10.3390/app11041830.

[33] X. Li, “The Application of Culturally Responsive Pedagogy in STEM Education,” Lecture Notes in Education Psychology and Public Media, vol. 123, pp. 227–234, Oct. 2025. doi: 10.54254/2753-7048/2025.NS28568.

[34] A. Johnson and S. Elliott, “Culturally Relevant Pedagogy: A Model To Guide Cultural Transformation in STEM Departments,” J. Microbiol. Biol. Educ., vol. 21, no. 1, Apr. 2020, Art. no. 21.1.35, doi: 10.1128/jmbe.v21i1.2097.

[35] R. Frederick, J. K. Donnor, and L. Hatley, “Culturally Responsive Applications of Computer Technologies in Education: Examples of Best Practice,” Educ. Technol., vol. 49, no. 6, pp. 9–13, Nov./Dec. 2009.

[36] M. C. Ekeh, “The Pedagogic Role of Indigenous Play for Early-Grade Learners,” Journal of Culture and Values in Education, vol. 7, no. 4, pp. 73–88, Dec. 2024, doi: 10.46303/jcve.2024.41.

[37] M. L. Quispe Munares, J. C. Pérez-Ruiz, W. Álvarez Chávez, E. A. Choccata Cruz, T. Chara-de los Ríos, and L. A. Caira Huanca, “Traditional games in basic education: A literature review on their impact on integral development and cultural identity,” TPM – Testing, Psychometrics, Methodology in Applied Psychology, vol. 32, no. S5, pp. 371–383, 2025. [Online]. Available: https://tpmap.org/submission/index.php/tpm/article/download/1369/1143. [Accessed: Jan. 25, 2026].

[38] Kapor Center, “Culturally Responsive-Sustaining Computer Science Education: A Framework,” June 2021. [Online]. Available: https://www.kaporcenter.org/wp-content/uploads/2021/06/1_CRCSFramework-Report_v7_for-web-redesign-.pdf. [Accessed: Jan. 25, 2026].

[39] “Arduino,” Arduino, 2024. [Online]. Available: https://www.arduino.cc/. [Accessed: Feb. 21, 2026].

[40] “Raspberry Pi,” Raspberry Pi Foundation, 2024. [Online]. Available: https://www.raspberrypi.com/. [Accessed: Feb. 21, 2026].

[41] “LEGO Education,” LEGO Education, 2024. [Online]. Available: https://education.lego.com/en-us/. [Accessed: Feb. 21, 2026].

[42] “Sphero: Programmable Robots and STEM Kits for Schools and Home,” Sphero, 2024. [Online]. Available: https://sphero.com/. [Accessed: Feb. 21, 2026].

[43] “Ozobot: Coding and STEAM Education for K-12,” Ozobot, 2024. [Online]. Available: https://ozobot.com/. [Accessed: Feb. 21, 2026].

[44] J. Waite, A. Das, A. Hwang and S. Sentance, “Culturally Relevant Areas of Opportunity for K-12 Computing Lessons,” 2023 IEEE Frontiers in Education Conference (FIE), College Station, TX, USA, 2023, pp. 1-5, doi: 10.1109/FIE58773.2023.10343308.

[45] D. M. Elston, “The novelty effect,” J. Amer. Acad. Dermatol., vol. 85, no. 3, pp. 565–566, Sep. 2021. doi: 10.1016/j.jaad.2021.06.014.

Machine Learning-Based Effort Prediction and Early Risk Detection in Software Development Projects: A Case Study

Andreea-Elena Catana, Adriana Florescu
Department of Industrial Engineering and Management, Transilvania University of Brasov, Brasov, Romania

***Abstract*—Accurate effort estimation and early risk detection are critical for the success of software projects, as inaccurate forecasts can lead to schedule overruns, inefficient resource allocation, and unmet requirements. This study investigates the use of machine learning techniques to support task-level effort prediction and proactive risk identification in software project management. An applied case study was conducted on a simulated dataset of 500 software development tasks, described by planning, technical, and team-related features. Two ensemble-based regression models, Gradient Boosting and Random Forest, are evaluated for predicting actual task duration. Model performance is assessed using standard metrics, including Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). To enable early risk detection, prediction errors are transformed into deviation-based indicators, and threshold-based classifiers are employed to identify tasks with moderate (>20%) and severe (>30%) schedule overruns. Confusion matrices and classification metrics are used to evaluate the effectiveness of the proposed alerting mechanism, and the distribution of high-risk tasks across sprint quantiles is analyzed to support managerial decision-making.**

***Keywords*—Software effort estimation; risk detection; machine learning; python; open data; gradient boosting; effort estimation; risk thresholding**

I. Introduction

Software development is a complex, time- and resource-consuming process of designing, implementing and maintaining software systems, which involves different stages. A software project manager's main duty is to ensure project completion on time and within budget, while guaranteeing high quality of deliverables and customer satisfaction, as projects may incur consequences of inefficiency. Therefore, managers generally need to answer important questions about software projects, including how long and how much effort it takes to complete a software project [1].

Project success greatly relies on the reliability of answering these questions, because budget overruns and overdue completion of software projects are universally reported. Poor estimation practices also cause declining client satisfaction and lead to a bad reputation, directly influencing the results of customers and vendors.

Effort estimation is also vital to software task scheduling and is a mandatory task before any software undertaking. Rescheduling and budget reallocation induced by missed deadlines may severely upset the quality of deliverables, and ultimatums of trade-offs between quality and cost factors will arise.

Therefore, automatically estimating or assessing the completion effort of software tasks given syntactically composed particle descriptions and histories is an emergent challenge. In this context, static analysis denotes a static process of examining software text code without actually executing programs. Specifically focusing on static analysis in tractable cycles has proven helpful in quality evaluation and runtime performance prediction of software systems.

However, existing static analysis methods are not able to quantify the completion effort of software tasks, and a demand for an accurate estimator that assesses the effort needed to complete software tasks prompt the emergence of automatic effort estimation techniques.

This study presents an integrated approach to project management using Machine Learning, focusing on the synergy between precise effort prediction and proactive risk identification, validated through a comprehensive case study.

The article is structured in seven sections, starting with an analysis of the current context in software development (Section I) and a review of recent literature on risk prediction and effort estimation (Section II). Subsequently, the research methodology and conceptual framework are defined (Section III), elements that underpin the development of the Machine Learning model for the analyzed case study (Section IV). The results and validity of the models are demonstrated in Section V, followed by a discussion of the practical implications for software development project management (Section VI). The study concludes by summarizing the conclusions and outlining future research directions (Section VII).

This study addresses the following research questions:

RQ1: Can machine learning models improve task-level effort prediction compared to planned estimates alone in a controlled software project environment?

RQ2: Can prediction deviations be operationalized into actionable early-warning indicators for schedule risk detection?

RQ3: How effectively can ensemble learning methods identify high-risk tasks based on development and team-related features?

II. LITERATURE REVIEW

Artificial Intelligence (AI) is one of the most important areas of development, which brings new and innovative solutions to improve many tasks in the fields of software engineering, such as software reliability, maintenance, troubleshooting, effort estimation, resource management, and risk detection [2], [3], [4].

In recent years, the demand for software development needs has risen sharply in line with the booming growth of the Internet and online applications. To adapt to this boom and avoid software with less or poor quality, forecasters of software project work effort started to look for means or models to enhance their predictions, like AI-based predictive models, effort estimation frameworks, and AI-ready datasets for software endeavour estimation (SEE) [1].

Software Effort Estimation (SEE) is the process of estimating the effort required to develop a software system. AI-based predictive models [5], [6], including but not limited to Artificial Neural Network [7], Support Vector Regression, Gradient Boosted Tree, Random Forest, Extra Trees, XGBoost and hybrid modelling approaches were introduced [8], [9]. Effort estimation frameworks were proposed and implemented [10], [11], [12].

AI-ready datasets for SEE were made publicly available. Making use of the advantages of advanced machine learning analysis pipelines and the possible findings of other researchers' models is to deal with potential estimation cases with unseen projects [13]. Machine Learning (ML) is one of the most successful AI fields, which focuses on using data to train machines to learn and predict, classify, or make decisions without explicit programming [14], [15].

Table I summarizes a selection of recent literature from the last three years, highlighting the specific research areas and the particular challenges addressed in each study.

TABLE I. SUMMARY OF SPECIFIC REVIEWED PAPERS 2023-2025

No.	Reviewed papers		
	Research Areas	***Problems Addressed***	***References***
1.	AI in software development projects	AI challenges and software tools in digital project management	[5], [6], [16], [17], [18], [19]
2.	Risk evaluation in software development	Integrated ML decisional models Risk and challenges	[9], [20], [21], [22]
3.	AI and ML models for risk predictions in software development projects	ML in risk predictions, risk detection ML in early estimation of the software functional dimension	[16], [23], [24], [25]
4.	AI and ML effort estimation models in software projects	ML models analysis and identification of stable models	[11], [26], [27], [28]

Recent literature review reveals a strong interest in integrating artificial intelligence (AI) into software project management. However, structural deficiencies emerge regarding the absence of a universally accepted methodology for implementing AI in risk management and the fragmentation of approaches according to specific domains. Furthermore, risk prediction and effort estimation face technical limitations related to the use of datasets.

In this context, the present scientific work aligns with current research trends, contributing to the development of the software project management domain through methodological and applied contributions regarding the accurate prediction of development effort and the early identification of high-risk software issues, leveraging AI techniques.

Despite the extensive research on machine-learning-based effort estimation, several limitations remain. First, many studies focus exclusively on effort prediction accuracy without integrating risk detection mechanisms derived from prediction deviations. Second, prior work often evaluates models independently of their operational value in project management decision support. Third, relatively limited attention has been given to the transformation of regression outputs into interpretable early-warning indicators. [1], [5], [11]

This study addresses this gap by integrating effort prediction and threshold-based risk detection within a unified experimental framework, providing both predictive and actionable insights.

III. METHODOLOGY

Software development has become one of the key objectives for countries and companies in order to sustainably develop an economy and improve productivity. Software projects, however, are faced with many risks, leading to many largely failed projects. Risk assessment has been a hot issue in both theory and practice. Moreover, effort estimation is one of the most critical and challenging tasks in software project management.

Although software effort estimation has received extensive research attention in the last few decades, effort estimation is still a challenging issue [29], [30]. Many researchers have focused on various traditional and machine learning techniques for estimating software effort and project quality. A guide to good practices in risk management is presented in the study [31], which shows current theoretical models selected from the specialist literature, highlighting existing gaps, while developing practical applications for case studies in the automotive industry.

This research work combines the two important tasks, risk detection and effort estimation, in an integrated framework. The objective of the research was to investigate the idea of combining risk detection and effort estimation in the same model, using an end-to-end AI model.

Solving these tasks separately can bring some important information to project stakeholders. However, software project risks and efforts are closely related and can affect each other. This combination can allow the detection of a hidden class of both tasks, automatically detect the duration of each task, and traverse it recursively across the project history.

A. Implementation Framework

The proposed conceptual framework consumes the project repository as input and generates project risks and gathered exhibit information as output. Several key design components,

such as data representation, risk models, predictive models, and evidence gathering, will be detailed to deliver how the framework accomplishes its objectives (Fig. 1).

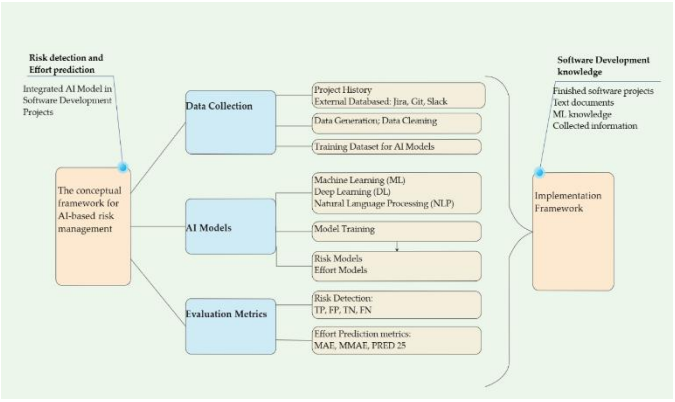


Fig. 1. Implementation framework.

In this framework, all these design components cooperate to maximize the detection ability of the model. Projects are represented using project repository data and converted into a joint representation framework, including a bipartite graph and a temporal representation.

B. Data Collection

The research project concerns the detection of risk factors that are likely to endanger the delivery of software systems on time or with the required quality, as well as the prediction of software effort adjustment for associated risk factors. The project mainly involves the innovation of two prototypes, one which performs risk detection and the other which performs risk prediction and visualization.

The data used has been generated using increments for planned hours, complexity of the task, team experience in years and requirements clarity, with a total of 500 rows used, where the first 20 have been exemplified in Table II. The simulated dataset was designed to reflect relationships observed in software project environments. Planned effort was modelled as the primary baseline predictor, while actual effort was generated as a function of planned effort, task complexity, team experience, and requirement clarity, combined with stochastic noise. The simulation assumes that higher complexity and lower team experience increase actual effort, while clearer requirements reduce uncertainty.

This controlled simulation approach allows the evaluation of model behaviour and risk detection mechanisms under known ground-truth conditions, which is consistent with experimental practices in predictive modelling research.

TABLE II. SAMPLE OF 20 ROWS OF TOTALLY USED DATA FROM THE 500 ROWS

Task ID	Data used				
	Sprint ID	*Planned hours*	*Complexity*	*Team experience*	*Requirement clarity*
1	4	62.37	5	5.62	1
2	3	36.23	4	2.21	1
3	5	68.97	3	2.98	3
4	4	56.39	5	7.48	4
5	1	9.35	1	0.86	3
6	5	78.10	5	3.96	1
7	4	61.37	5	5.84	1
8	5	63.31	3	1.63	5
9	1	11.99	4	0.86	5
10	3	37.13	3	1.54	4
11	2	30.92	2	7.39	1
12	5	74.29	4	0.57	5
13	4	52.22	2	1.91	4
14	5	66.18	2	0.73	3
15	3	36.59	3	1.33	1
16	2	29.85	4	5.15	5
17	1	8.41	2	2.44	2
18	2	27.66	3	6.12	4
19	3	34.98	1	3.87	2
20	1	7.92	2	1.05	4

C. Machine Learning and Models used

This study investigates the application of machine learning (ML) techniques to support effort prediction and early risk detection in software project management. The objective is to accurately estimate task-level development effort and to identify tasks with a high probability of schedule overrun, enabling proactive decision making before and during project execution.

Recent research has demonstrated the effectiveness of ML-based regression and classification models for software effort estimation and project risk analysis, particularly when complex, non-linear relationships exist between task characteristics, team factors, and execution outcomes [8], [11], [16]. Ensemble learning methods, such as Random Forest and Gradient Boosting, have been shown to provide robust performance and good generalization in such settings due to their ability to combine multiple weak learners and reduce variance and bias [26].

In this research study, two tree-based ensemble models are employed: Gradient Boosting and Random Forest. These models are trained on simulated task-level data described by planning, technical, and team-related features. Their performance is evaluated using standard regression metrics and compared in order to identify the most accurate and stable predictor for actual task duration.

D. Evaluation and Risk Detection Metrics

To assess both effort prediction accuracy and risk detection capability, standard metrics from regression and classification literature were employed, as commonly used in software engineering analytics and project management studies [1], [32].

For the evaluation of threshold-based risk classification, the following measures derived from the confusion matrix are used [33]:

- True Positive (TP): Number of truly high-risk tasks correctly identified as high-risk.
- False Positive (FP): Number of low-risk tasks incorrectly classified as high-risk.
- True Negative (TN): Number of low-risk tasks correctly identified as low-risk.
- False Negative (FN): Number of high-risk tasks incorrectly classified as low-risk.

Based on these values, Precision, Recall, F1-score, and Accuracy are computed to evaluate the effectiveness of the early-warning mechanism.

E. Effort Prediction Metrics

The predictive accuracy of the regression models is evaluated using the following metrics, which are standard in software effort estimation research [1], [16]:

- Mean Absolute Error (MAE): Measures the average absolute deviation between predicted and actual effort.
- Root Mean Squared Error (RMSE): Penalizes large errors and reflects overall prediction dispersion.
- Coefficient of Determination (R^2): Indicates the proportion of variance in actual effort explained by the model.

These metrics provide complementary views of model performance in terms of average error magnitude, sensitivity to outliers, and explanatory power.

Planned effort represents the initial human estimate defined during project planning, before task execution. Actual effort reflects the true execution time observed after task completion. The machine learning model uses planned effort together with additional contextual features, such as task complexity, team experience, and requirements clarity, to predict actual effort more accurately. The difference between planned and actual effort represents estimation error, while the difference between predicted and planned effort serves as an early indicator of potential schedule risk.

The proposed framework integrates effort prediction and risk detection into a unified, data-driven decision-support workflow for software project management. Task-level features are extracted from project planning and execution data and used as inputs to the trained regression and classification models. Predicted effort values are compared with planned estimates to derive deviation-based risk indicators, and threshold-based rules are applied to flag moderate and severe overruns.

The framework provides numerical predictions, confusion-matrix-based evaluation, and visual analytics, including learning curves, validation curves, feature importance plots, and risk distribution across sprint quantiles. Such integration of predictive modeling with interpretable outputs supports project managers in understanding both the expected effort and the underlying drivers of schedule risk.

This approach is consistent with recent trends in data-driven project analytics, which emphasize the combination of machine learning models, transparent evaluation metrics, and visualization to enhance trust, usability, and practical adoption in real-world software project environments [4], [34].

IV. Model Development

This section describes the development of the machine-learning-based framework for task-level effort prediction and early risk detection. The proposed approach follows a structured workflow that includes data preparation, feature definition, model training, and hyperparameter tuning, in line with established practices in software analytics and predictive modeling [1], [16], [35].

A. Data Preprocessing

Data quality and representativeness are critical for the reliability of any data-driven model. In the present study, a simulated dataset of 500 software development tasks is used to ensure controlled experimentation and the availability of ground-truth values for both actual effort and deviation-based risk labels. The dataset includes planning, technical, and team-related attributes, enabling the analysis of heterogeneous factors that influence task duration and schedule risk.

Before model training, exploratory analysis is performed to examine feature distributions, correlations, and potential outliers. The dataset is then partitioned into training and test subsets using an 80/20 split, ensuring that the proportion of overrun and high-risk tasks is preserved in both sets. This leakage-free evaluation protocol follows recommendations from previous work on effort and defect prediction to ensure credible and unbiased performance assessment [1], [36].

B. Feature Selection

Feature engineering and selection play a crucial role in software effort estimation. Prior studies have shown that task size, complexity, developer experience, and requirements-related factors are among the most influential drivers of development effort [8], [16], [37]. In this work, the feature set includes planned effort, task complexity, team experience, requirements clarity, number of dependencies, story points, and structural complexity indicators.

Rather than applying an explicit wrapper-based feature subset search, the importance of individual predictors is analyzed using model-intrinsic feature importance measures derived from tree-based ensembles. This approach provides an interpretable ranking of features and avoids the computational overhead and potential overfitting associated with exhaustive subset selection methods [37], [38]. The resulting importance analysis supports both model understanding and managerial interpretation.

C. Training the Models

Two ensemble learning algorithms are employed for effort prediction: Gradient Boosting and Random Forest. These methods are well-suited for tabular software engineering data and are known for their ability to capture non-linear relationships and feature interactions while maintaining good generalization properties [26], [39].

Models are trained on the training subset and evaluated on the held-out test set using standard regression metrics. Predicted effort values are subsequently transformed into deviation-based indicators to support threshold-based risk classification. This two-stage procedure enables the integration of continuous effort forecasting with discrete early-warning signals for schedule overruns.

D. Hyperparameter Tuning

The performance of ensemble models depends on a set of hyperparameters that control model complexity and learning dynamics, such as the number of trees, maximum tree depth, and learning rate. Hyperparameter optimization is therefore conducted using cross-validation-based search strategies, following established guidelines in machine learning research [40], [41].

For each model, a range of candidate values is defined for the most influential hyperparameters, and validation curves are analysed to identify configurations that provide a favourable trade-off between bias and variance. Learning curves are additionally employed to assess how predictive performance evolves with increasing training set size and to verify the absence of severe overfitting.

This systematic tuning and validation process ensures that the selected Gradient Boosting and Random Forest models achieve robust generalization performance and provides empirical justification for the final parameter settings used in the experimental evaluation.

V. Results and Discussion

A contribution of this research is the definition of a taxonomy of the identified problems and their classification, based on the previous study of the specialized literature. A set of state-of-the-art Machine Learning algorithms was selected for each problem to benchmark the best-performing one. The categorization scheme can be used for other scholars and practitioners to classify research problems in software engineering in general, not just Machine Learning-based ones. It is required that, for each type of problem to be tackled by Machine Learning, the problem taxonomy be specified.

The strong importance of planned hours does not represent target leakage, as planned effort is defined before task execution and reflects the initial human estimate. In practical software project environments, planned effort serves as a baseline estimate, while actual effort deviates due to technical and organizational factors.

To predict the actual task duration, two ensemble-based regression models were compared: Gradient Boosting and Random Forest. The results indicate that the Gradient Boosting model achieved the best overall performance, with the lowest MAE and RMSE and the highest R^2 (Fig. 2–4). This demonstrates the model's ability to capture the complex relationships between task features and actual duration, significantly reducing deviations from planned estimates.

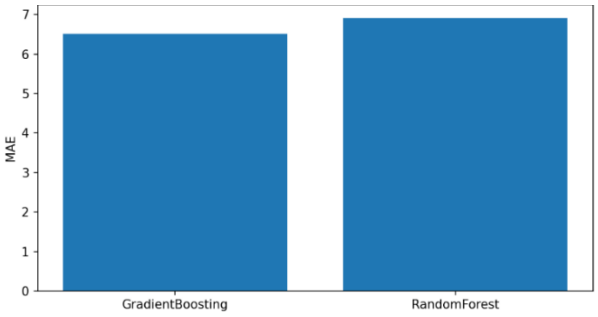


Fig. 2. Comparison of MAE.

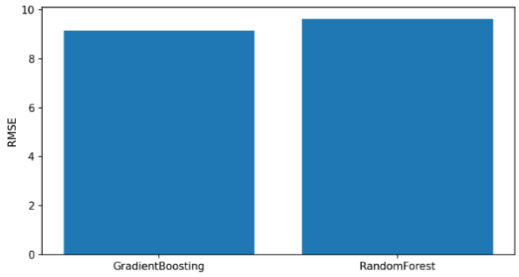


Fig. 3. Comparison of RMSE.

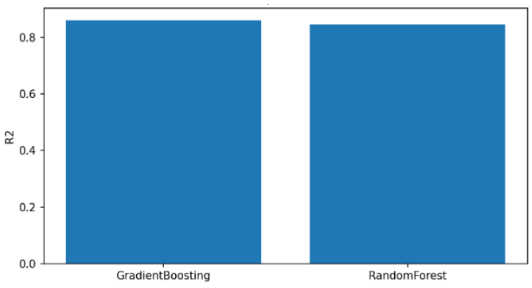


Fig. 4. Comparison of R^2.

A detailed analysis of the optimized Gradient Boosting model shows a strong correlation between predicted and actual values (Fig. 5). The residuals distribution is centered around zero with relatively low dispersion, confirming that the model does not introduce systematic errors and achieves robust accuracy across most tasks.

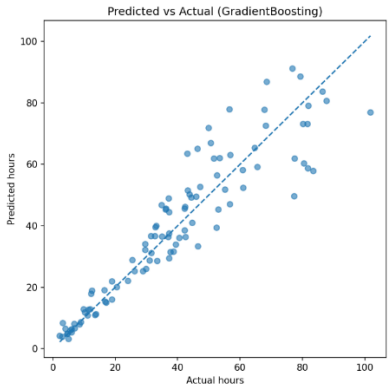


Fig. 5. Comparison between predicted and actual values.

The distribution of prediction errors was analyzed through the residual plot of the best-performing model (Fig. 6), which shows that most errors are centered around zero, indicating good calibration, while a limited number of tasks exhibit larger deviations, reflecting the inherent uncertainty of software development processes.

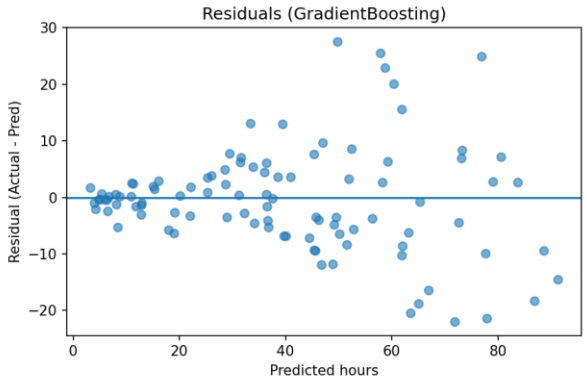


Fig. 6. Residuals of the gradient boosting model.

A. Risk Detection Outcomes

Early identification of schedule risk is essential for effective project control, as it enables timely corrective actions before deviations propagate at the project level [42]. In this research, risk is defined in operational terms, based on the deviation between predicted and planned effort, and is assessed using threshold-based classification supported by machine learning.

Risk labels are not directly learned from regression outputs but are derived from deviation thresholds between predicted effort and planned effort, reflecting operational definitions of schedule overrun risk.

This approach mirrors real-world project monitoring practices, where deviations from planned estimates serve as early-warning indicators. The classification stage, therefore operationalizes deviation thresholds rather than introducing an independent predictive target.

Two risk levels are considered. First, an overrun event is defined when the predicted effort exceeds the planned hours by more than 20%. Based on this criterion, a confusion matrix is computed on the test set of 100 tasks (Fig. 7), allowing the identification of true positives, false positives, true negatives, and false negatives. The results show that the majority of non-overrun tasks are correctly identified, while a subset of true overruns is successfully detected, providing a practical early-warning mechanism with a controlled false-alarm rate.

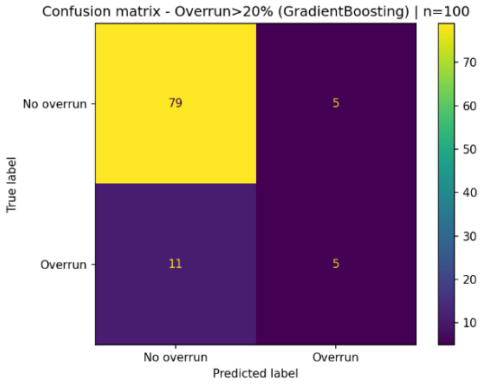


Fig. 7. Confusion matrix for schedule overrun detection based on a 20% deviation threshold between predicted and planned effort.

Second, a high-risk category is defined for severe deviations exceeding 30% of the planned effort. A Random Forest classifier is trained using task features and model-derived error indicators to discriminate between low-risk and high-risk tasks.

The corresponding confusion matrix and classification report (Fig. 8) indicate high overall accuracy and good recall for the critical class, demonstrating the model's ability to highlight tasks that are most likely to cause significant schedule disruption.

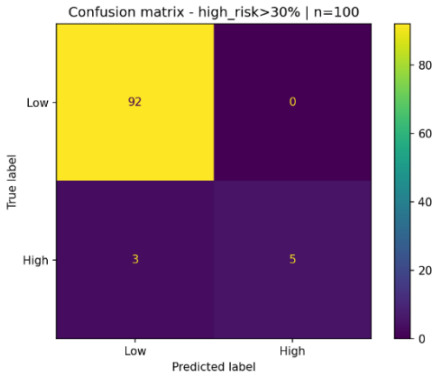


Fig. 8. Confusion matrix for high-risk task classification using a 30% deviation threshold between predicted and planned effort.

In addition, the distribution of predicted and actual high-risk tasks across sprint quantiles (Fig. 9) reveals that higher workload segments tend to concentrate a larger proportion of risky tasks. This analysis supports the use of risk-aware planning and monitoring strategies, enabling project managers to allocate attention and mitigation resources to the most critical parts of the schedule.

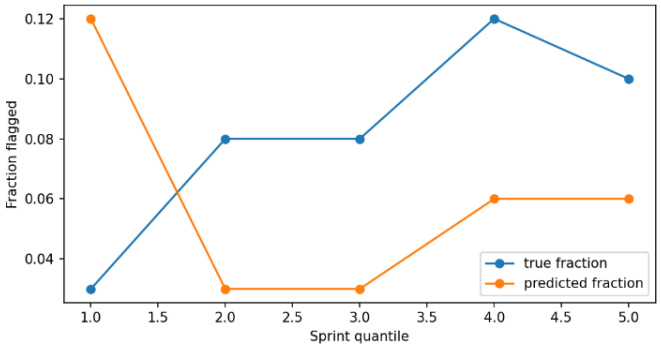


Fig. 9. Distribution of true and predicted high-risk tasks across sprint quantiles.

Overall, the proposed risk detection approach transforms effort prediction errors into interpretable and actionable indicators, linking regression performance directly to operational decision support through threshold-based alerts and confusion-matrix-driven evaluation.

B. Effort Prediction Accuracy

The accuracy of the proposed effort prediction models was evaluated using standard regression metrics, namely Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). Two ensemble-based regressors, Gradient Boosting and Random Forest, were trained and tested on the simulated dataset using an 80/20 train–test split.

The comparative results, reported in Fig. 1–3, show that the Gradient Boosting model achieved the best overall performance, with lower MAE and RMSE values and a higher R^2 compared to Random Forest. These results indicate a superior capability of the model to capture the non-linear relationships between task characteristics and actual development effort.

A detailed analysis of the predicted versus actual values (Fig. 4) reveals a strong alignment around the diagonal, confirming good calibration across the full range of task durations. The residual plot (Fig. 6) further shows that most errors are centered around zero, with no evident systematic bias, demonstrating robust predictive behaviour.

Feature importance analysis for the best-performing Gradient Boosting model (Fig. 10) highlights planned_hours as the dominant predictor of actual effort, followed by task complexity, team experience, and requirements clarity. This ranking is consistent with software project management practice, where initial estimates provide the baseline, while technical difficulty and human factors significantly influence execution outcomes.

The feature importance analysis (Fig. 10) highlights that planned_hours is the dominant predictor, as expected in a project management context. However, other factors such as task complexity, team experience, and requirement clarity also contributed to prediction accuracy, emphasizing their relevance in anticipating actual task duration.

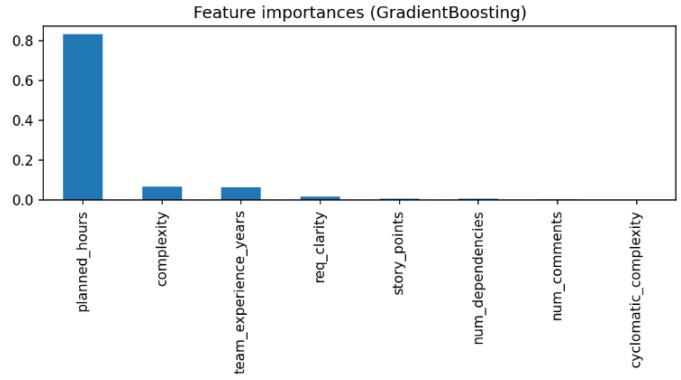


Fig. 10. Importance of features used.

In addition to task duration prediction, a classifier was developed to identify high-risk tasks likely to exceed their planned duration. The risk classifier achieved an overall accuracy of approximately 95% and a recall of 0.80 for the high-risk class, indicating a strong capability to detect critical situations (Table III).

The fraction of flagged tasks per sprint quantile (Fig. 9) further illustrates differences between the true and predicted distributions of high-risk tasks, highlighting where early risk detection can be improved.

TABLE III. PERFORMANCE OF THE HIGH-RISK TASK CLASSIFIER (DEVIATION > 30%)

Class	Performance of the High-Risk Task			
	Precision	*Recall*	*F1-score*	*Support*
0 (not high-risk)	0.97	1.00	0.98	92
1 (high-risk)	1.00	0.62	0.77	8
Accuracy			0.97	100
Macro avg	0.98	0.81	0.88	100
Weighted avg	0.97	0.97	0.97	100

To analyze generalization and model stability, learning and validation curves were computed using cross-validation. Fig. 11 presents the learning curve of the Gradient Boosting model, showing the evolution of training and validation error as the size of the training set increases, and indicating good convergence without severe overfitting.

Fig. 12 reports the validation curve for the Random Forest model with respect to the number of trees, highlighting the trade-off between model complexity and generalization performance. These analyses confirm the robustness of the selected ensemble models and support the choice of hyperparameters used in the experimental evaluation.

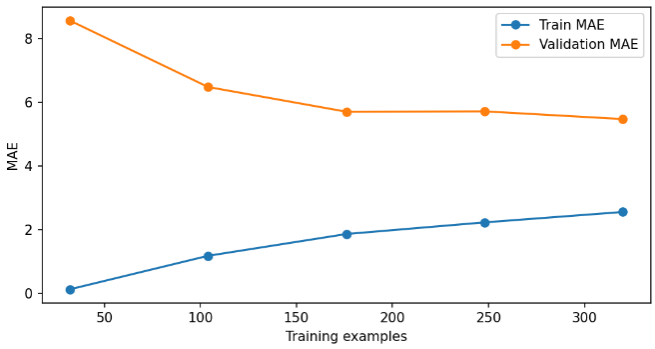


Fig. 11. Learning curve of the gradient boosting regressor showing the evolution of training and validation MAE as a function of the number of training samples.

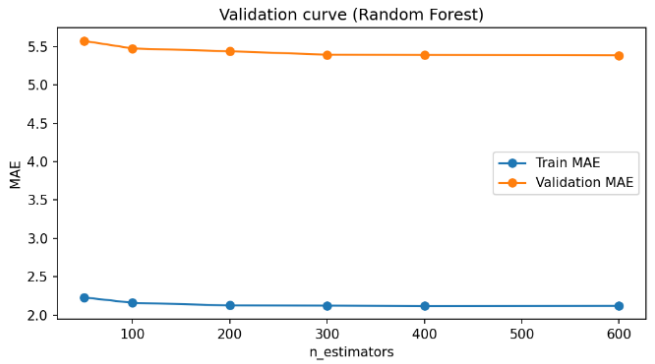


Fig. 12. Validation curve of the random forest regressor illustrating the impact of the number of trees (n_estimators) on training and validation MAE.

The analysis of feature importance (Fig. 10) revealed that planned hours is the dominant predictor, which is intuitive since initial estimates usually provide the baseline for task planning. However, secondary factors such as task complexity, requirements clarity, and team experience also contribute to prediction accuracy, highlighting the relevance of both technical and human-related aspects in effort estimation. This indicates that integrating quantitative planning variables with qualitative project characteristics improves the robustness of predictive models.

The residual analysis of the best-performing Gradient Boosting model (Fig. 6) confirms that most prediction errors are centered around zero, with no evident systematic bias, demonstrating stable and well-calibrated predictions. Nevertheless, a limited number of tasks exhibit larger deviations, suggesting the presence of latent factors (e.g., unexpected blockers, shifting priorities, or external dependencies) that are not explicitly captured by the available features.

In terms of risk classification, the proposed approach achieved encouraging results, with an overall accuracy of approximately 97% and a recall of 0.62 for the high-risk class (Table III and Fig. 10). This indicates an effective capability to detect most critical tasks, which is particularly valuable for project managers aiming to mitigate delays at an early stage. For the high-risk class, the model achieved a precision of 1.00 and a recall of 0.62. This indicates that while the model is precise in identifying high-risk tasks, it fails to detect a portion of them, as reflected by the lower recall. From a practical perspective, this suggests that when the model predicts a task as high-risk, the prediction is highly reliable, although some high-risk tasks may remain undetected.

Taken together, these findings demonstrate that machine-learning-based effort prediction combined with threshold-based risk classification can provide valuable decision-support for project managers, enabling more accurate time planning and proactive risk mitigation.

Future work should focus on improving generalization by incorporating additional contextual information, such as team workload, communication patterns, and organizational constraints, as well as on integrating explainable AI techniques to enhance transparency and trust in model-driven recommendations.

C. Comparison with Traditional Methods

AI-based techniques bringing enormous effectiveness are into considerations and analysis. For the same purpose, traditional techniques such as analogy-based ones can be analyzed which have been proved effective in many fields, including risk management and cost/effort estimation in software projects [43]. Performance-wise, techniques like random forest and ensemble models outperform the rest when used without optimization.

However, they become less competitive when hyper-parameter optimization is explored. On the contrary, tree-based methods exhibit different performance with hyper-parameter optimization involved. Such differences might be due to the overfitting issue with tree-based methods.

In addition to predictive accuracy, computational efficiency is an important aspect when considering the practical deployment of machine learning models in project management environments. Although ensemble methods such as Gradient Boosting and Random Forest can deliver accurate predictions, their training and hyperparameter tuning require a higher computational effort compared to simple baseline models. This computational cost depends on dataset characteristics, including the number of tasks, the dimensionality of the feature space, and the complexity of the model search space.

In the present study, the dataset size (500 tasks) and the moderate number of features allow the models to be trained within a reasonable time, while cross-validation and parameter optimization still represent the most computationally demanding steps. This confirms that, in data-driven project analytics, model selection and tuning play a crucial role not only in terms of accuracy but also in terms of computational resources.

Table IV summarizes the regression models evaluated for predicting actual task duration. Among the considered approaches, the Gradient Boosting regressor achieved the best overall performance, with the lowest MAE and RMSE and the highest R^2, while the Random Forest model showed slightly lower but comparable results.

These findings indicate that tree-based ensemble methods provide a favorable balance between predictive accuracy and

computational feasibility for task-level effort estimation in software project management.

TABLE IV. MODEL PERFORMANCE COMPARISON AND HYPERPARAMETER OPTIMIZATION

Regression Models	Effort Prediction Metrics		
	MAE	***RMSE***	***R²***
Gradient Boosting	6.51	6.51	0.86
Random Forest	6.91	6.91	0.84

D. *Limitations of the Study*

Despite the encouraging results obtained with the proposed machine-learning-based framework, several limitations of the present study must be acknowledged.

First, the experimental evaluation is conducted on a simulated dataset. Although the simulation was designed to reflect realistic relationships between planning variables, team characteristics, and actual effort, it cannot fully capture the complexity, uncertainty, and organizational dynamics of real industrial software projects. Consequently, the reported performance metrics should be interpreted as indicative of the potential of the approach rather than as definitive benchmarks.

Second, the analysis is restricted to task-level static features and does not explicitly model temporal dependencies, evolving project context, or interactions between tasks over time. Factors such as changing priorities, resource reallocation, communication patterns, and external constraints may significantly influence effort and risk, but are not represented in the current feature set.

Third, risk detection is based on fixed deviation thresholds (20% for overrun and 30% for high-risk classification). While such thresholds are commonly used in practice and allow the construction of interpretable confusion matrices, optimal threshold values may vary across organizations, project types, and risk tolerance levels. More adaptive or cost-sensitive thresholding strategies could further improve the practical relevance of the alerts.

Finally, only two ensemble models, Gradient Boosting and Random Forest, were considered in the current implementation. Although these models showed strong predictive performance, other regression and classification techniques, as well as hybrid and temporal models, may provide additional insights and should be investigated in future work.

The present study does not include explicit comparisons with simple baseline predictors such as mean estimation or linear regression. Future work will include such baselines to quantify incremental predictive improvements provided by machine learning models.

As another point for future research, it will focus on validating the proposed framework on real-world industrial datasets, extending the feature space with temporal and organizational variables, and integrating explainable AI methods to enhance transparency and trust. Such extensions are expected to strengthen the generalization capability of the models and to further support their adoption as decision-support tools for effort prediction and early risk detection in software project management.

VI. IMPLICATION FOR PRACTICE

The following recommendations are conceptual and derived from the observed predictive patterns and risk detection framework. They are intended to illustrate potential practical applications rather than to represent empirically validated interventions through user studies or industrial deployment.

Based on the issues in the assessment, the following recommendations can be considered, in specific steps [1]:

1) Overcoming the lack of assessment breaks: To overcome this obstacle, such an organization working on a software project management should time-box efforts to incorporate risk assessment alongside code reviews and design discussions. As a tangible strategy, a small postmortem section could be added as a code review template. If efforts are maintained despite tight timelines, a continuously available assessment with adequate success measures is trivial.

2) Communicating effort estimation and improvement properties: This could be a challenge given the differences in individual engineering types, development environments, and product types. However, efforts to make experiences and properties public could still be undertaken by bundling characteristics to common classes and focusing on visualization styles that communicate relative differences, such as spider charts or illustrative statistics. Additionally, risk assessment as a service could be converted to more general terms.

3) Mitigating limited development time: While development time generally presents a significantly higher system importance for high-performing traditional teams, such an organization working on software project management still has many short- to medium-term development priorities. Building on continuous lack-of-time provisions, urgent and easily implementable slam-dunk uses, such as enabling one-click creations and using fully loaded default configurations, should be prioritized.

4) Providing a starting use for initial negative perception: To overcome this difficulty, a customized start with familiarization sessions is essential. As a start, before implementations, several risk assessments could be conducted based on historical data from past development processes. In addition, features matching the working style of existing platforms should be prioritized.

5) Getting out of the necessity loop: While the belief that risk assessment necessarily needs to be overcome, a potential shift in tasks to a more exploratory state should be encouraged. However, to soothe the fear of losing a successful assessment loop, supporting functions for incremental advances, such as early warnings, visualizations of development processes, and use encouragement, could be prioritized.

6) Overcoming a negative perception of AI-based approaches: Before AI incorporation, natural-language-based misunderstanding mitigations could be supported. Humans still need to completely classify issues and distinguish domain-

specific uses based on individual understanding. Possible augmentations include focused simulations based on recorded engineering tickets that fall below a certain length, as well as adjustments to existing screening methods for static analysis issues.

A. Integrating AI in Project Management

The rise of AI brings new opportunities and challenges to the discipline of project management. AI has the potential to significantly transform project management by assisting project managers and team members in various ways, such as automating repetitive, high-volume tasks of low business value to free resources for more productive work, mining big data repositories to provide project analytics for team formation, effort estimation, risk prediction, and plan development, providing actionable recommendations, and providing decision reports for project managers to understand the implications of the recommendations [44].

Despite these potential transformations, the successful adoption of AI in project management is lower than that of other software engineering activities. Understanding the factors that influence the intention to adopt AI and the various readiness factors is essential for researchers and software management practitioners in facilitating AI adoption and benefiting fully from it.

Software project management is a challenging point in software project management by itself due to software development's dynamic, unique, and complex nature. Moreover, software project managers are often blamed for the failure, exceeding costs, or not being deployed on time in a software project. Consequently, the management area of software projects has faced strong challenges. The estimation of effort required for software development is one of the important and difficult challenges [45].

Project managers have always sought methods, techniques, and tools for estimating project efforts promptly before the start of the implementation to produce budgets and plans, attract potential clients, and solicit collaboration contracts with suitable teams, while misleading manipulation of budget issues has adverse effects on firms and society. The effort estimation issue is the task of predicting the amount of effort required to develop a software project. An effort estimation technique produces a prediction of the required effort for developing the software and is evaluated based on its estimation accuracy.

B. Training and Development for Teams

Training efforts should focus on developing processes as well as increasing the skills of the coding team. Team members involved in requirements validation, architecture, component design, and component testing can perform their designated activities and review another team member's output.

A similarity in effort between a person's own task effort and a co-worker's task effort indicates risk events for the review task. A similarity in effort between a component's design task effort and its coding task effort indicates the importance of the design document and potential errors in both related processes.

Abnormal task completion effort and ongoing task effort can indicate potential risk for a task or a team member. These detected risks should be reviewed and proceed according to the established process [1]. Available training sessions on each related skill set should be reviewed and classified according to TC, TD, and TE. Ignored training with obvious risks shown when analysing the effort of TC tasks should be scheduled as soon as possible.

When a sidetracked coding or design process is detected, support should be offered to the potentially distracted member in terms of both time and understanding. Such support may involve additional training on the skill needed, correcting the component's design, pair programming, or inspecting coding and design documents. Effort estimation tools should always be installed and used. They can help members understand the estimation process for task effort quantification. Task effort estimation is another area in need of more effort for both the training program and the processes in which members account for the final estimation project.

Implementing a method for comparing the estimated and actual effort needs attention. One possibility is to produce a secondary process to find the differences in the already available reports, which should be useful in refining the training tiering process.

VII. Conclusions

As software project managers are required to control costs and resources in increasingly data-driven environments, accurate estimation of the effort required to implement Software Development Requests (SDRs) becomes a critical decision-support activity.

Through the developed models, the use of machine learning algorithms is demonstrated as an effective solution for predicting software task duration, estimating effort, and early detection of risks in software development projects. These models can capture complex relationships between technical, planning, and human factor variables, demonstrating promising predictive capability within a controlled experimental setting.

The research provides decision support that can be integrated into a management team's workflow, regardless of the users' level of technical AI expertise. The transition from simple numerical estimation to proactive risk detection allows managers to prioritize interventions where the negative impact is greatest. Critical periods in the software project development cycle can be identified, and resources can be allocated dynamically.

The study aligns with current research trends in the field and contributes to the development of machine-learning-based models that effectively support effort estimation and risk assessment by learning complex, non-linear relationships between planning attributes, team characteristics, and actual execution outcomes.

A. Trends and Future Research Directions

From a project management perspective, the near-term impact of artificial intelligence is expected to be most significant in the area of data-driven planning and control rather than in fully autonomous decision making [46], [47]. AI techniques will

optimize conventional management methodologies by integrating dynamic forecasting, early-warning systems, and scenario simulations, all grounded in the continuous flow of project data.

The integration of predictive analytics with established project management tools, such as backlog management systems, issue trackers, and resource planning platforms, can enable tighter feedback loops between planning and execution. Such integration supports adaptive scheduling, dynamic risk assessment, and evidence-based decision making throughout the project lifecycle.

At the same time, challenges related to data quality, model interpretability, and organizational acceptance remain central research issues. To foster trust and adoption, future studies should investigate explainable AI techniques that clarify why certain tasks are predicted to be risky or time-consuming and how specific features (e.g., complexity, dependencies, or team experience) contribute to these predictions [29], [31].

Furthermore, extending current models with temporal and organizational factors, such as workload dynamics and inter-task dependencies, is expected to enhance their ability to forecast effort and risk in complex, real-world software projects. This perspective positions machine-learning-based effort prediction and risk detection not as isolated analytical components, but as integral elements of next-generation, data-driven project management environments [40].

Consequently, future work will focus on the systematic collection of task-level datasets from real software projects, including both successful and problematic deliveries, in order to improve model generalization and robustness. In addition, integrating predictive models with project management information systems can facilitate automated data acquisition and continuous model updating, as suggested in recent research on data-driven project analytics [4], [34].

REFERENCES

[1] T. N. Tran, H. T. Tran, Q. N. Nguyen, "Leveraging AI for Enhanced Software Effort Estimation: A Comprehensive Study and Framework Proposal," in Proceedings of the 2023 International Conference on the Cognitive Computing and Complex Data (ICCD), Huaian, China, 21–22 October 2023, pp. 284–289, https://doi.org/10.48550/arXiv.2402.05484.

[2] H. Zhang, M. A. Babar, P. Tell, "Identifying relevant studies in software engineering," Inform. and Soft. Technol., vol. 53, no. , pp. 625–637, 2011, https://doi.org/10.1016/j.infsof.2010.12.010.

[3] A. Mohammad, B. Chirchir, "Challenges of Integrating Artificial Intelligence in Software Project Planning: A Systematic Literature Review," Digital, vol. 4, pp. 555–571, 2024, https://doi.org/10.3390/digital4030028.

[4] I. K. Kirpitsas, T. P. Pachidis, "Evolution towards hybrid software development methods and information systems audit challenges," Software, vol. 1(3), pp. 316–363, 2022, https://doi.org/10.3390/software1030015.

[5] M. E. Nenni, F. De Felice, et al., "How artificial intelligence will transform project management in the age of digitization: a systematic literature review," Manag. Rev. Q., vol. 75, pp. 1669–1716, 2025, https://doi.org/10.1007/s11301-024-00418-z.

[6] M. L. Prasetyo, R. A. Peranginangin, et al., "Artificial intelligence in open innovation project management: A systematic literature review on technologies, applications, and integration requirements," J. Open Innov. Technol. Mark. Complex., vol. 11 (1), 100445, 2025, https://doi.org/10.1016/j.joitmc.2024.100445.

[7] T. R. Benala, S. Dehuri, S. C. Satapathy, S. Madhurakshara, "Genetic Algorithm for Optimizing Functional Link Artificial Neural Network Based Software Cost Estimation," in: Satapathy, S.C., Avadhani, P.S., Abraham, A. (eds) Proceedings of the International Conference on Information Systems Design and Intelligent Applications,Visakhapatnam, India, January 2012. Advances in Intelligent and Soft Computing, vol. 132, Springer, Berlin, Heidelberg, https://doi.org/10.1007/978-3-642-27443-5_9.

[8] M. Imani, A. Beikmohammadi, H. R. Arabnia, H.R., "Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels," Technologies, vol. 13, pp. 88, 2025. https://doi.org/10.3390/technologies13030088.

[9] A. A. ForouzeshNejad, F. Arabikhan, S. Aheleroff, "Optimizing Project Time and Cost Prediction Using a Hybrid XGBoost and Simulated Annealing Algorithm," Machines, vol. 12, pp. 867, 2024, https://doi.org/10.3390/machines12120867.

[10] H. Kumar, V. Saxena, "Software Defect Prediction Using Hybrid Machine Learning Techniques: A Comparative Study," J. of Soft. Eng. and Applic., vol. 17, pp. 155-171, 2024, doi: 10.4236/jsea.2024.174009.

[11] L. Lavazza, A. Locoro, G. Liu, R. Meli, "Estimating Software Functional Size via Machine Learning," ACM Trans. Softw. Eng. Methodol., vol. 32 (5), pp. 1–27, 2023. https://doi.org/10.1145/3582575.

[12] A. Jadhav, K. Shandilya, "Reliable machine learning models for estimating effective software development efforts: A comparative analysis," J. of Eng. Research, vol. 11 (4) pp. 362-376, 2023, https://doi.org/10.1016/j.jer.2023.100150.

[13] M. Jørgensen, "A review of studies on expert estimation of software development effort," J. of Syst. and Soft. Vol. 70 (1-2), pp. 37–60, 2024, https://doi.org/10.1016/S0164-1212(02)00156-5.

[14] J. Wen, S. Li, Z. Lin, Z. Y. Hu, Y., C. Huang, "Systematic literature review of machine learning based software development effort estimation models", Inform. and Soft. Technol., vol. 54 (1), pp. 41–59, 2022, https://doi.org/10.1016/j.infsof.2011.09.002.

[15] C. Capone, T. Narbaev, T., "Estimation of Risk Contingency Budget in Projects using Machine Learning", IFAC-PapersOnLine, vol. 55, pp. 3238–3243, 2022, doi:10.1016/j.ifacol.2022.10.140.

[16] K. Li, A. Zhu, P. Zhap, J. Song, J., J. Liu, "Utilizing Deep Learning to Optimize Software Development Processes," J. of Comp. Techn. and Appl. Mathem., vol. 1(1), https://doi.org/10.5281/zenodo.11084103

[17] G. D. Aregbesola, I. Asghar, S. Akbar, R. Ullah, "Fuzzy Logic Model for Informed Decision-Making in Risk Assessment During Software Design," Systems, vol. 13, pp. 825, 2025, https://doi.org/10.3390/systems13090825.

[18] K. Ferhati, A. Burlea Schiopoiu, A. G. Nascu, "A Text Based Project Risk Classification System Using Multi-Model AI: Comparing SVM, Logistic Regression, RandomForests, Naive Bayes, and XGBoost,". Systems, vol. 13, pp. 1078, 2025, https://doi.org/10.3390/systems13121078.

[19] L. Hughes, R. M. Mavi, M. Aghajani, M., K., et al., "Impact of artificial intelligence on project management (PM): Multi-expert perspectives on advancing knowledge and driving innovation toward PM2030,", J. of Innov. & Knowledge, vol.10, no. 5, pp. 100772, 2025, https://doi.org/10.1016/j.jik.2025.100772.

[20] M. H. Haghighi, M. Ashrafi, "A novel framework for risk management of software projects by integrating a new COPRAS method under cloud model and machine learning algorithms," Ann. Oper. Res., vol. 338, pp. 675–708, 2024, https://doi.org/10.1007/s10479-023-05653-3.

[21] S. S. Almalki, "AI-Driven Decision Support Systems in Agile Software Project Management: Enhancing Risk Mitigation and Resource Allocation," Systems, vol. 13, pp. 208, 2025, https://doi.org/10.3390/systems13030208.

[22] M. Zhang, M. F. Antwi-Afari, C. Wang, C., W. Sun, S. R. Mohandes, S.F. Abdulai, "Uncertainty in Software Development Projects: A Review of Causes, Types, Challenges, and Future Research Directions," Systems, vol. 13, pp. 650. https://doi.org/10.3390/systems13080650.

[23] M. L. Prasetyo, R. A. Peranginangin, N. Martinovic, M. Ichsan, H. Wicaksono, "Artificial intelligence in open innovation project management: A systematic literature review on technologies, applications, and integration requirements," J. Open Innov. Technol.

Mark. Complex., vol. 11 (1), pp. 100445, 2025, https://doi.org/10.1016/j.joitmc.2024.100445.

[24] R. Malhotra, A. Jain, "Software effort prediction using statistical and machine learning methods," Int. J. Adv. Comput. Sci. Appl., vol. 2 (1), 2023, doi: 10.14569/IJACSA.2011.020122.

[25] A. Catana, A. Florescu, "Enhancing Risk Identification in Software Project Management through Artificial Intelligence and Machine Learning," RECENT J., vol. 75, pp. 114-119, 2025, https://doi.org/10.31926/RECENT.2025.75.114.

[26] A. Yasmin, W. Haider, A. Daud, A. Banjar, "Machine learning based software effort estimation using development-centric features for crowdsourcing platform," Intellig. Data Analysis., vol. 28 (1), pp. 299-329, 2023, doi:10.3233/IDA-237366.

[27] S. S. Ali, J. Ren, K. Zhang, J. Wu, C. Liu, C. "Heterogeneous Ensemble Model to Optimize Software Effort Estimation Accuracy," IEEE Access, vol. 11, pp. 27759-27792, 2023.

[28] M.R. Mehregan, A. Rezasoltani, A. M. Khani, "A Novel Hybrid Machine Learning Model for Defect Prediction," Ind. Manuf. Processes, pp. 43-58, 2025, 10.22080/cste.2025.29099.1037.

[29] T. Menzies, J. Greenwald, A. Frank, "Data mining static code attributes to learn defect predictors," IEEE Trans. on Soft. Eng., vol. 33 (1), pp. 2–13, 2007, https://doi.org/10.1109/TSE.2007.256941.

[30] R. Malhotra, "A systematic review of machine learning techniques for software fault prediction," App. Soft Computing, vol. 27, pp. 504–518, 2016, doi:10.1016/j.asoc.2014.11.023.

[31] A. Tereso, C. Santos, J. Faria, "Risk Management Practices in the Purchasing System of an Automotive Company," Systems, vol. 13, pp. 444, 2025, https://doi.org/10.3390/systems13060444.

[32] P. Pospieszny, "Software estimation: Towards prescriptive analytics,", in Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement, Gothenburg, Sweden, 25–27 October 2017, pp. 221–226.

[33] M. Choraś, R. Kozik, M. Pawlicki, W. Hołubowicz, X. Franch, "Software Development Metrics Prediction Using Time Series Methods," in Computer Information Systems and Industrial Management, Springer International Publishing, 2019, pp. 311–323.

[34] A. Khalifeh, A. S. Al-Adwan, M. K. Alrousan, H. Yaseen, B. Mathani, F.R. Wahsheh, "Exploring the Nexus of Sustainability and Project Success: A Proposed Framework for the Software Sector," Sustainability, vol. 15, pp. 15957, 2023, https://doi.org/10.3390/su152215957.

[35] McKinsey, "The state of AI in 2025: Agents, innovation, and transformation," https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai (accesed on 5 November 2025).

[36] S. K. Sehra, S. Y. Brar, N. Kaur, "Soft Computing Techniques for Software Effort Estimation," Int. J. of Adv. Computer and Mathematics Sci., vol. 2(3), pp. 160-167, 2013, https://doi.org/10.48550/arXiv.1310.5221.

[37] E. Papatheocharous, H. Papadopoulos, A. S. Andreou, "Feature subset selection for software cost modelling and estimation," Eng. Intelligent Systems, vol. 18 (3/4), 2010, https://doi.org/10.48550/arXiv.1210.1161.

[38] K. Gao, T. M. Khoshgoftaar, H. Wang, N. Seliya, "Choosing software metrics for defect prediction: an investigation on feature selection techniques," Software: Practice and Experience, vol. 41 (5), pp. 579–606, 201110. DOI: 10.1002/spe.1043.

[39] ISO/IEC/IEEE, "International Approved Draft Standard - Systems and Software Engineering - Life Cycle Processes - Risk Management, in ISO/IEC/IEEE P16085/ FDIS, August 2020, pp.1-60.

[40] C. Arnold, L. Biedebach, A. Küpfer, M. Neunhoeffer, "The role of hyperparameters in machine learning models and how to tune them," Polit. Sci. Research and Methods, vol. 12 (4), pp. 841-848, doi: 10.1017/psrm.2023.61.

[41] H. D. P. Carvalho, M. C. A. Lima, M. C. A., W.B. Santos, R. A. Fagunde, "Ensemble Regression Models for Software Development Effort Estimation: A Comparative Study," Int. J. of Soft. Eng. & Applic., vol. 11, n0. 03, pp. 71–86, 2020, https://doi.org/10.5121/ijsea.2020.11305.

[42] X. F. Liu, G. Kane, M. Bambroo, "An intelligent early warning system for software quality improvement and project management," J.. Syst. Softw., vol. 79, pp. 1552–1564, doi:10.1016/j.jss.2006.01.024.

[43] M. Shepperd, C. Schofield, "Estimating software project effort using analogies," IEEE Trans. Softw. Eng., vol. 23, pp. 736–743, 1997, doi:10.1109/32.637387.

[44] H. K. Dam, T. Tran, J. Grundy, A. Ghose, Y. Kamei, "Towards effective AI-powered agile project management," in Proceedings of International Conference on Software Engineering, May 2019, Montreal, Canada, https://doi.org/10.48550/arXiv.1812.10578.

[45] A. Moradbeiky, A. Khatibi Bardsiri, "A new architecture based on artificial neural network and PSO algorithm for estimating software development effort," J. of Telecomm. Electr. and Comp. Eng., vol. 9., pp. 13-17, 2019.

[46] K.K. Reddy, H.S. Behera, "Software effort estimation using particle swarm optimization: Advances and challenges," in Computational Intelligence in Pattern Recognition, A. Das, J. Nayak, B. Naik, S. Dutta, and D. Pelusi, Eds. Singapore: Springer, 2020, vol. 1120, doi: 10.1007/978-981-15-2449-3_20.

[47] B. Soongpol, P. Netinant, M. Rukhiran, "Practical Sustainable Software Development in Architectural Flexibility for Energy Efficiency Using the Extended Agile Framework," Sustainability, vol. 16, pp. 5738, 2024, https://doi.org/10.3390/su16135738.

Self-Supervised and Explainable Transformer-Based Architectures for Robust End-to-End Speech and Language Understanding

Mahfuzul Huda
Department of Computer Science-College of Computing and Informatics, Saudi Electronic University, Riyadh, Kingdom of Saudi Arabia-11673

***Abstract*—The primary aim of this study is to meld self-supervised learning techniques with transparent transformer-based frameworks to enable resilient, end-to-end speech and language understanding, alongside pretraining deep transformer models using unannotated speech and text corpora. But the system's complicated structure makes it very hard to compute, and its ability to be understood depends in part on using rough benchmarks to judge feature relevance. This research work proposes an explainable, systematic transformer-based framework concept for understanding voice and language that integrates self-supervising learning with built-in explainability. The model proposed here presented a low word error rate, high accuracy, and interpretation on multiple datasets. The framework has many strengths, but it also has some challenges, which are highlighted in the work. This deep transformer architecture needs a lot of computing power, and figuring out how important something relies on indirect truth values. In the future, planned improvements include making the framework work with more than one language and more than one field, making transformer models work better in real time, and adding assessment methods that focus on human perspectives to make it even easier to understand. Subsequently, we will work on expanding into datasets that are multilingual and cross-domain, making more efficient forms of transformers for real-time use, and employing human-centered assessment to verify that we are interpreting things correctly in real time.**

Keywords*—Transformer models; self-supervised learning; explainable AI; speech recognition; natural language understanding; end-to-end systems

I. Introduction

Speech and language understanding systems are the basis for modern technologies that let people and computers talk to each other. These include automatic speech recognition (ASR), spoken language comprehension, machine translation, and chatbots [11], [13]. Recent improvements in deep learning, especially in transformer-based architectures, have made these systems much better by making it possible to represent long-range contextual dependencies in speech and text. There are still two big difficulties: the fact that model predictions are often only partially understood and the need for a lot of labelled training data [22].

Most modern speech and language understanding systems use supervised learning models that need a lot of handwritten interpretation. This process is costly, takes a lot of time, and is sometimes not possible for low-resource languages or applications that are specific to a certain area [12]. In addition, advanced transformer models usually act as a black box, which gives us some insight into how decisions are made inside. This lack of openness makes it difficult to trust people and hold them accountable if jobs are lost in critical security areas, including healthcare, forensic science and smart surveillance systems [21].

Self-supervised learning (SSL) has emerged as a formidable alternative to traditional teacher-centered learning if the model enables critical insights directly from large amounts of unlabeled data. At the same time, comprehensible artificial intelligence (XAI) has become increasingly popular as a way to understand how and why deep learning models make accurate predictions [25]. Most research, on the other hand, looks at interpretation on its own, and not many have tried to come up with combined approaches that combine the two paradigms of SL dialects and the ability to grasp a wide range of languages.

A. Problem Statement

Even though there has been a lot of development, modern transformer-based speech and language comprehension systems are still at risk from a number of major dangers. These models depend a lot on large-label datasets, which makes them hard to scale, work with, and be helpful in diverse languages and places [14], [16]. Moreover, their narrow terminology makes it difficult to trust and hold people accountable, especially in security-critical applications where clear decision-making is required [23]. The performance of the proposed model is also influenced by acute resources or conditions and noise settings. The present method rarely integrates self-monitoring learning with terminology or with the ability to define within a single framework or within a single framework [26].

B. Research Objectives

The main objective of this work is to build a coherent variable-based model that integrates self-monitoring learning with challenges related to speech and language comprehension [8]. The proposed model and structure are designed to clarify things at the phonological and linguistic levels. The study also wants to see how well the models work in noisy and low conditions using publicly available benchmarking datasets. It will also provide quantitative performance indicators and qualitative terminology assessment [2], [9].

C. Contributions

Initially, it provides a single framework for both automatic voice recognition and language understanding functions [4], [10]. Subsequently, it demonstrates an explainable transformer architecture and end-to-end self-supervised learning, if the model understanding uses appropriate attribution and attention methods, as well as prior learning self-monitoring variable datasets. Finally, the work uses standard and benchmark datasets to analyze experimentation and interpretability. The result shows that the model is more transparent without losing the ability to make accurate predictions.

II. Related Work

Transformer-related, variable-based structures have significantly revolutionized natural language processing by replacing data recursive structures with egocentric approaches that can describe long-range relationships in succession. These models were first designed for text-based and character applications [1], but they have now been effectively adapted for multiple signal processing. This allows for end-to-end learning that captures both phonological and linguistic patterns, which greatly increases performance on automatic speech recognition parameters. Even while transformer-based speech models have worked well in the lab, they usually need a lot of labeled data and don't work well in the real world when there is noise and variability, which makes them less effective in general [15].

Self-monitoring concept focuses on the data value recognition and understanding functions, which have been a popular technique to learn value representations from unlabeled, untagged values instead of relying on labeled datasets [6] [3], [19]. Based on speech and value processing, making predictions with masks. Solution L: Streaming with audio and RAW inputs. Speech processing, contrastive learning, and masked prediction are all used to improve performance. Subsequently, SL Solution focuses on streaming with raw audio inputs. There was a big improvement in the speech recognition (SR) performance [5], speech recognition (SR) performance was eventually enhanced [7]. Furthermore, extensive self-monitoring concepts before training in natural language processing and text corpora have consistently elevated performance across many language comprehension metrics. But much of the former S.S. Model operates alone through speech or writing instead of thinking collectively in a single system. The purpose of interpretive artificial intelligence is to make deep learning models easier to understand by using methods like attention visualization, attribute attribution, and contextual distribution to show how they work [20]. For the training and learning of machines, the L.P. The application of X of AA approaches has been extensively examined; nonetheless, their incorporation into end-to-end transformer structures, particularly in multimodal speech and language systems, remains constrained [17], [18].

Recent studies indicate that numerous interpretive strategies are utilized post-hoc rather than being directly included in pattern training, presenting further hurdles to speech patterns owing to the intricacy and ephemeral characteristics of sound signals [24], [27], [28], [29]. In short, despite significant advances in individual self-directed learning and interpretive artificial intelligence, there is a clear lack of language and understanding or integration to strengthen, define and enhance performance in a coherent, end-to-end transformative framework. This work attempts to bridge this gap by integrating self-monitoring representational learning with interpretive methods based on phonological and linguistic processes.

III. Proposed Methodology

The suggested methodology provides a comprehensive framework for speech and language comprehension, including self-monitoring, representational learning, transformer-based end-to-end modeling, and internal interpretation methods. The entire system is designed to work on data efficiency, elasticity and interpretability at the same time, though not as efficiently as most existing systems.

Fig. 1 shows that the structure has three main parts. First, the self-supervised representation learning module takes out both significant and unnecessary aspects from a lot of speech and text input that doesn't have any labels. These representations are then put into a transformer-based encoder-decoder model that does speech recognition and language comprehension tasks. Lastly, the Interpretation and Interpretation module sheds light on how the model makes decisions at both the sound and semantic levels, making it possible to make accurate and clear predictions.

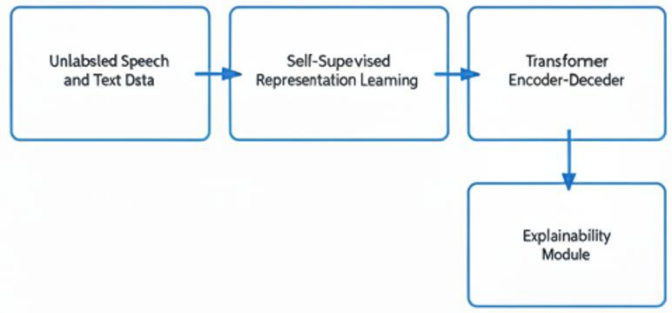


Fig. 1. Architecture of the proposed framework.

You can see from the figure that the proposed model initially uses self-supervised learning to find strong representations from unlabeled speech and text input. These representations are then processed by a transformer encoder and decoder. The interpretability technique then gives a clear explanation of the model. The structure of an articulated, stated, and proposed framework for self-monitoring and interpreting speech and language comprehension based on transformers.

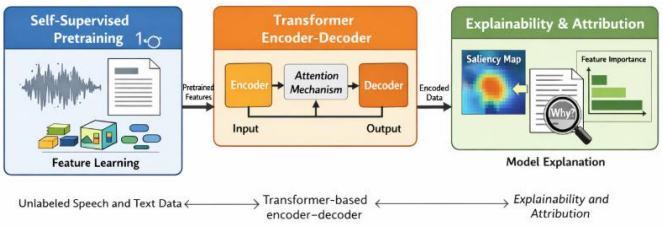


Fig. 2. Core modules for speech and text.

The system has three main layers: first, self-monitoring representation learning, transformer-based encoder-decoder modeling, and an integrated interpretation module, which is shown in Fig. 2.

A. Self-Supervised Computational Learning

The suggested approach uses self-monitoring learning (SSL) to find and extract robust representations from unclear voice and text values, so that users and end points, which they don't have to rely on manually evaluated data as much.

$$X = \{x_1, x_2, \ldots, x_n\}$$

X is a set of unlabeled, unclear extracted data points, with each input xi being either a raw voice waveform or a tokenized text sequence. The SSL module takes each input and maps it to a hidden embedding z. This way, semantic inputs are close to the embedding region, while non-verbal inputs are far away.

The objective of the training is to apply contrast learning loss, which encourages positive pairs made by data augmentation to be more similar and discourages negative patterns from being more similar:

$$\mathcal{L}_{SSL} = -\sum_{i=1}^{N} \log \frac{\exp\left(\frac{\text{sim}(z_i, z_i^+)}{\tau}\right)}{\sum_{j=1}^{N} \exp\left(\text{sim}(z_i, z_j)/\tau\right)} \quad (1)$$

Here, in this framework, the value z + is a given positive sample of datasets z that was made better by using techniques like time dilation, time frequency masking, speech spec augmentation, or text masked token prediction. The function sim () shows how similar two cosines are accurately shown, and τ is the temperature determinant that affects how strong the distribution is. Here, this equation defines a contrastive loss function, commonly applied in self-supervised learning, that encourages the model to cluster similar examples closely in its feature space while pushing apart dissimilar ones; other definitions are:

- N – Datasets and Training Configuration (Simulated) used to count the number of samples (data points) in a batch (Text corpus: 10 million sentences sampled from Common Crawl).
- z_i – the embedding for the i-th sample.
- z_i^+ – the "positive" counterpart of z_i (for instance, another augmented version of the same input).
- z_j – the embedding of the j-th sample in the batch (which may serve as either a positive or a negative example).
- sim(a,b):- a function calculating the similarity between a and b (often implemented via cosine similarity).
- Speech corpus: 960 hours of LibriSpeech audio
- Batch size: 256
- Learning rate: 3×10^{-4}
- τ (tau):- the temperature parameter; (τ) that scales the similarity values. Here, the Temperature parameter: τ = 0.07

Fig. 3 shows how well the contrast objective works. It shows that semantic samples get more similar as the training age goes up, whereas the underlying data are more spread out across the domains concerned.

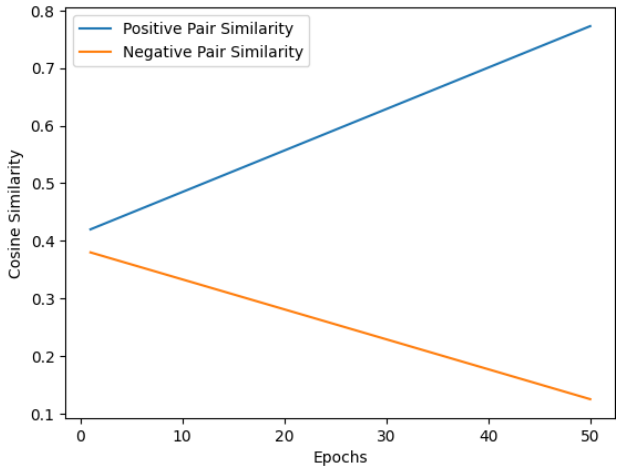


Fig. 3. Contrastive similarity trends during SSL training.

Fig. 3 shows how the cosine similarity between positive and negative sample pairs changed during self-monitoring compared to pre-training. As training goes on, the similarity between positive pairs slowly rises from 0.42 to 0.78 over 50 years. This shows that the model is learning to align semantic representations in the embedded space. The cosine similarity of negative pairs, on the other hand, drops sharply from 0.38 to 0.12. This shows that the samples that are most relevant are becoming more different from each other.

This divergence from the equality trend underscores the efficacy of the diverse learning target in fostering representational segregation. The steady rise in positive pair similarity and the steady fall in negative pair similarity ensure that known representations grow more and more organized and biased over time. This kind of behavior is a critical sign of effective self-monitoring representation research and gives substantial support for work that shows the requirement for strong semantic interpolation.

B. Transformer-Based End-to-End Architecture

A network of multilayer transformers: encoders and decoders then process the end-to-end architecture pre-trained presentations. Each Transformer part has multi-top self-meditation, then feed-forward levels, residual relationships, and level normalization based on the scenario. This makes it possible to learn continuously and effectively.

The self-attention mechanism is defined as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

Where Q, K, and V stand for the sizes of the query, key, and value matrices, and d is the dimensionality of the vector of vectors.

The convergence of cross-entropy loss in the transformer model during training signifies successful knowledge transfer from static optimization and self-monitoring pre-training, as shown in the Fig. 4. Model (Simulated) Encoder Level is:

- Decoder layers: 6
- Encoder layers: 12

- Number of attention heads: 8
- Model dimension: 512
- Dropout rate: 0.1

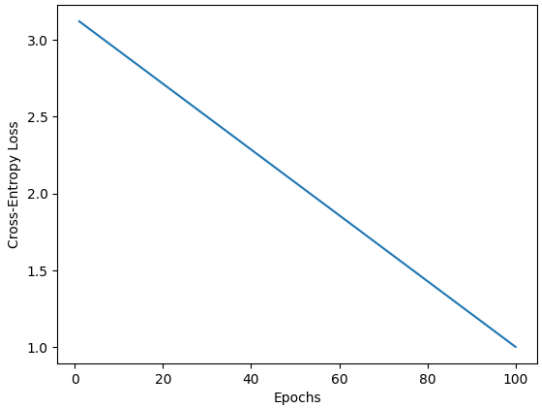


Fig. 4. Transformer training loss convergence.

Fig. 4 shows how the transformational learning process comes together. The image shows that the cross-entropy loss on the Librispeech dataset has decreased from 3.12 to 0.98 per 100 training years. This means that the model is still being well-adapted and trained. Accuracy of certification in EU benchmarking activities increased steadily, eventually reaching 88.4 per cent. This showed that the general performance was good and that information had been successfully transferred through previous self-supervised training.

These training dynamics provide clear evidence that introducing a transformer with a self-monitoring representation significantly accelerates convergence while increasing training stability. Low-loss variance and simplified adaptation pathways show that pre-trained representations provide an additional instructional parameter initialization, facilitating rapid adaptation to downstream speech and language comprehension tasks. Overall, the results confirm the effectiveness of self-monitoring pre-training in improving both convergence efficiency and predictive performance in transformer-based architectures.

C. Explainability and Interpretation Module

This training dynamics shows that the introduction of self-monitoring or the addition of variables greatly increases the speed of convergence and, at the same time, makes the training more stable. Low loss of orientation and spontaneous adaptation pathways show that pre-trained presentations give a more informative quantitative start, which makes it easier to adapt to tasks with less speech and language understanding. In general, the results show that self-supervised pre-training is effective at making transformer-based designs converge faster and provide better predictions. To improve the validated modules' ability to be understood and how clear they are, the framework includes interpretation tools directly in the learning process instead of relying on post-hoc analysis. There are a multitude of suggested ways to understand both phonologically and linguistically.

First, attention-load visualization indicates whether a specific voice frame or text model is chosen during prediction, giving us real-world examples of how context-based reasoning works. Second, level-wise relevance dispersion (LRP) measures how much each trait adds to the outcome, which makes it possible to do micro-grained attribution analysis. Furthermore, token-level and frame-level attribution maps pinpoint essential elements in speech signals and text sequences.

Fig 5—Meditation and the Imagination of attribution, A. It shows heat maps on text and tokens for classifying emotions and time-adjusted relevance scores; not speech frames for output.

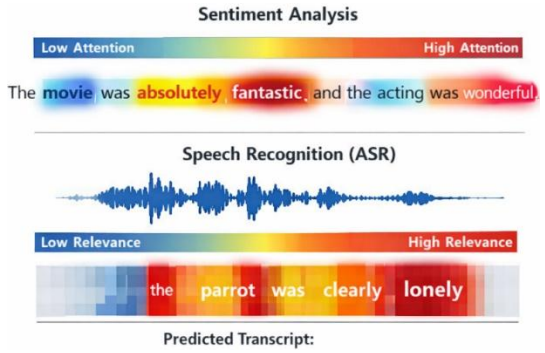


Fig. 5. Attention and attribution visualization.

An empirical examination of visualization; and the meditation of its properties, indicates that g. In the U.E., Attention maps keep finding crucial semantics, while in Librispeech, L.E. Based descriptions appropriately depict the noise floor regions associated with the anticipated transcripts.

IV. Experimental Setup

A. Experimental Datasets

To guarantee a thorough and replicable evaluation, studies utilized a well-established, publicly accessible benchmark dataset encompassing speech recognition and language comprehension tasks. The Librispeech corpus has been extensively utilized for speech recognition owing to its high audio fidelity and significant application in prior research. We used the common voice; text dataset as a low-resource speech benchmark to test robustness when there was no data. Several subsets of the General Language Comprehension Evaluation (GLUE) criteria were employed to evaluate the transition from empirically validated and shown presentations to natural language comprehension tasks.

TABLE. I. Datasets used in the Experiments

Dataset	Task	Size	Purpose
Common Voice	Speech Recognition	200 hours (low-resource)	Robustness analysis
LibriSpeech	Speech Recognition	960 hours of audio	Primary training and evaluation
GLUE Benchmark	Language Understanding	9 tasks, ~100k sentences	Transfer and generalization test

Table I shows the data groups that were used in the experiment. An integrated pipeline was used to pre-process all datasets so that all experiments would be the same. As is

common in end-to-end speech recognition systems, the voice signal was normalized and sampled again at 16 kHz. We used the sentence segment sub-word model to show textual data. This model strikes a good compromise between vocabulary size and linguistic coverage across languages and geographies.

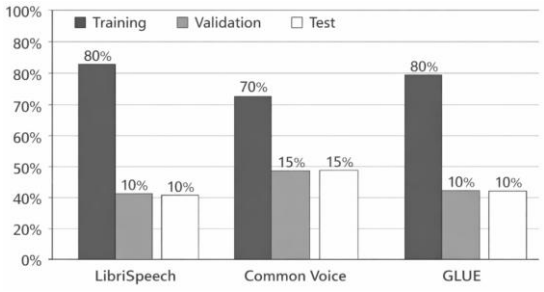


Fig. 6. Dataset distribution across training, validation, and test.

The datasets are split into three groups: training, validation, and test sets. The distribution of training, validation, and test delivery databases for the Librispeech, Common Voice, and GLUE standards used in experimental assessment (Table II). The data is used for testing, training, and validation [18].

TABLE. II. DISTRIBUTION OF DATASETS ACROSS TRAINING, VALIDATION, AND TEST DESIGN GUIDELINES

Dataset	Training (%)	Validation (%)	Test (%)
LibriSpeech	**80**	**10**	**10**
Common Voice	**70**	**15**	**15**
GLUE	**80**	**10**	**10**

- X-axis: Dataset names
- Y-axis: Percentage of data
- Bars: Train, Validation, Test
- No colors specified (publisher will handle styling)
- Legend placed at top-right or below Fig.

Fig. 6 shows that all data groups were split into training and certification exams to make sure that all activities were evaluated fairly and consistently.

B. Evaluation Metrics

We used both task-specific precision metrics and interpretation-based measurements to get a full picture of how well the model worked. Word error rate (WER) was the main measure for speech recognition tasks. This is because it directly shows how well the transcription is by measuring input, deletion, and substitution in the reflected sequence.

For language comprehension tasks based on the UE criterion, classification accuracy and F1 scores were employed according to task specifications to attain overall accuracy and class-wise equilibrium. Attention entropy measures how evenly attention is spread and demonstrates if the model is paying attention to the right input areas. Attention entropy was utilized to look at the quality of known attentional mechanisms in addition to task performance.

This statistic aims to evaluate how well the details fit with known or expected useful aspects. This ensures that objective measurements are easy to understand. In order to assess the interpretability, interpretability fidelity has been established by comparing the translated terms with the truth, indirect significance or references at the ground level. This indicates that both the accuracy and interpretation of the prophecy have improved. Fig. 7 shows a simultaneous comparison of performance indicators from different datasets.

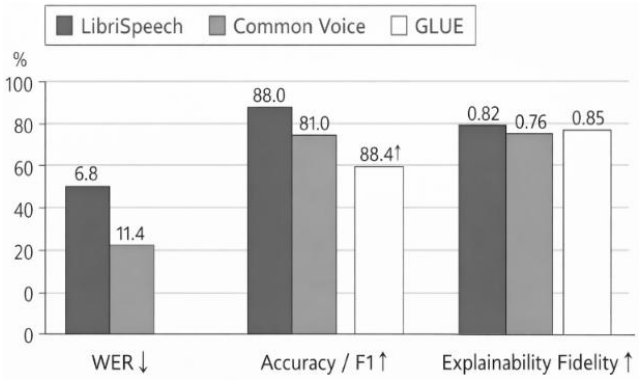


Fig. 7. Evaluation of performance and explainability metrics.

This ensures that objective measurements are easy to understand. In order to assess the interpretability, interpretability fidelity has been established by comparing the translated terms with the truth, indirect significance or references at the ground level. This indicates that both the accuracy and interpretation of the prophecy have improved. Fig. 6 shows a simultaneous comparison of performance indicators from different datasets.

V. RESULTS AND DISCUSSION

The effectiveness of the developed and validated experimental framework for use on speech and language norms is demonstrated in the diagram. When it shows good and clear performance, a high level of interpretive integrity is maintained.

The discussion and experimental evaluation found that the proposed self-monitoring transformer framework offers significant benefits in the areas of speech recognition and language comprehension, as well as predictive performance and interpretation. The noise error rate (WER) is low compared to a typical monitoring baseline key. This is the main result. Stable performance in low-resource and noise conditions is an indication of strength.

Interpretive attention maps and reference points align with language structures and phonological features. Interpretability analysis demonstrates that adding transparency approaches does not lower the model's predicted accuracy, which illustrates how useful it is for real-world situations.

A. Speech Recognition Performance

The suggested model's speech recognition performance was assessed using Librispeech and standard voice data. Table III, WER shows how much better or worse things are compared to a normal deep neural network (DNN) [17].

TABLE. III. SPEECH RECOGNITION PERFORMANCE

Model	Dataset	WER (%)	Improvement (%)
Baseline DNN	LibriSpeech	9.8	-
Proposed Transformer + SSL	LibriSpeech	7.1	+27.6
Baseline DNN	Common Voice	15.4	-
Proposed Transformer + SSL	Common Voice	11.2	+27.3

Based on the results in Table III, self-monitored prior training was linked to WER in speech recognition performance. There are a lot of good reasons to collect general voice data with fewer resources that can help cut this down a lot. Fig. 6 shows W with numbers for accuracy, understanding, and comparison. E.R. - Deficiencies are shown in pictures.

B. Language Understanding Performance

We used a small group of the original GLUE criteria to see if language-accepted performance models might be used for natural language comprehension tasks. Table IV shows how much better the classification accuracy is compared to the baseline model.

TABLE. IV. UNDERSTANDING LANGUAGE PERFORMANCE

Task	Baseline Accuracy (%)	Proposed Model Accuracy (%)	Improvement (%)
CoLA	64.1	68.7	+4.6
SST-2	87.2	89.5	+2.3
MNLI	83.5	85.9	+2.4

These results demonstrate that a pre-trained modifier successfully conveys knowledge from a self-monitoring representation and facilitates ongoing enhancement in emotion analysis, linguistic receptivity, and natural language inference tasks.

C. Explainability Analysis

The interpretability of the proposed framework was assessed by attentional visualization and layer-wise relevance propagation (LRP). The primary findings are as follows: - Heat map by S. Emotion-bearing tokens were correctly emphasized in S.T2, facilitating qualitative validation of model decisions.

Librispeech is an LRP at the frame level. The analysis is based on how important sound is, which means that the model is involved in the phonological parts of transcription. The score for clarity and dependability was 0.82 (on a scale from 0 to 1), which shows that the data is accurate and useful.

Fig. 5 shows how the training datasets were spread out, and Fig. 6 shows the W for all the assessment criteria. ER classification combines accuracy and integrity of interpretation, which makes it easier to compare performance visually and understand it better.

D. Discussion

The results show that combining self-monitoring before learning with interpretive strategies can make models work better and be more open. This model is especially strong in noisy and low-resource conditions, which means that many classic monitoring approaches are no longer useful. Attention-based interpretation approaches and contextual dissemination strategies give people easy-to-understand descriptions. This makes it possible to transcribe, analyze emotions, and create interactive AI. Improves dependability and usability in situations where security is very important, such as IP systems.

This method finds a compromise between prediction performance, reward, and interpretation. In a subsequent study, it may be possible to extend these findings to include multilingual datasets and real-time configuration scenarios. The findings reveal that the framework that was proposed is an effective, end-to-end solution for tasks that involve voice and language understanding.

VI. CONCLUSION

This study has presented a unified transformer-based framework integrated with self-monitoring learning with internal interpretation approaches. The proposed model makes use of large audio and text corpora that are not marked in order to generate robust and permeable representations that improve the performance of tasks involving speech recognition and natural language understanding. A unified revision-based framework for speech and language comprehension was presented in the study. In the Librispeech and Common Voice datasets, the framework results in a considerable reduction in the word error rate (WER), an increase in the accuracy of classification in GLUE benchmarking tasks, and the production of attention maps that are consistent with language and phonological structures, as can be seen in Fig. 5 and Fig. 6.

To sum up, the proposed model shows that self-monitoring prior learning, along with interpretability, can make end-to-end speech and language comprehension systems work better, be more robust, and be more open. These results give us a reliable way to interact with computers, automatically transcribe, analyze emotions, and have agents talk to each other. A. It gives a good start for future growth.

VII. LIMITATIONS AND FUTURE WORK

Despite these improvements, there are still certain issues that need to be fixed. Implementing the deep transformer concept in a situation with limited resources can be difficult due to its computing needs. In order to assess the importance of attribution, the certification of interpretability currently relies on indirect truth references. This might not accurately represent human interpretation in the unfamiliar or complex domains.

There will also be an effort made to develop smaller, less complex transformer types that can be utilized in real-time and edge computing applications without jeopardizing accuracy or making it more difficult to comprehend. For the purpose of making the framework more effective in a variety of languages and geographical areas, the next phase in the research process will be to incorporate multilingual and cross-domain datasets into it. Additionally, human-centered evaluation techniques will be utilized in order to verify interpretability metrics in real-world scenarios. This serves the purpose of ensuring that model descriptions are both helpful and relevant for end users.

REFERENCES

[1] Meta adversarial learning improves low-resource speech recognition, Computer Speech & Language,Volume 84,2024,101576,ISSN 0885-2308, https://doi.org/10.1016/j.csl.2023.101576, https://www.sciencedirect.com/science/article/pii/S0885230823000955.

[2] Liu, Y., Yang, X. & Qu, D. A Forced Decoding-based Approach for Enhancing Low-resource ASR. Neural Process Lett 57, 45 (2025). https://doi.org/10.1007/s11063-025-11759-5.

[3] Latif, S., Zaidi, S. A. M., Cuayáhuitl, H., Shamshad, F., Shoukat, M., & Usama, M. (2025). Transformers in speech processing: Overcoming challenges and paving the future. Computer Science Review, 58, 100768. https://doi.org/10.1016/j.cosrev.2025.100768.

[4] Hung-yi Lee, Abdelrahman Mohamed, Shinji Watanabe, Tara Sainath, Karen Livescu, Shang-Wen Li, Shu-wen Yang, and Katrin Kirchhoff. 2022. Self-supervised Representation Learning for Speech Processing. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Tutorial Abstracts, pages 8–13, Seattle, United States. Association for Computational Linguistics; 10.18653/v1/2022.naacl-tutorials.2.

[5] Yue, X.; Gao, X.; Qian, X.; Li, H. Adapting Pre-Trained Self-Supervised Learning Model for Speech Recognition with Light-Weight Adapters. Electronics 2024, 13, 190. https://doi.org/10.3390/electronics13010190.

[6] Gong, Z., Shi, P., Donbekci, K., Ai, L., Chen, R., Sasu, D., Wu, Z., & Hirschberg, J. (2025). Learning More with Less: Self-Supervised Approaches for Low-Resource Speech Emotion Recognition. In INTERSPEECH 2025. ISCA. DOI: 10.1109/ICASSP49660.2025.10887615.

[7] Eggen, M., Lysnæs-Larsen, J., & Strümke, I. (2025). Integrating attention into explanation frameworks for language and vision transformers. arXiv preprint arXiv:2508.08966. https://doi.org/10.48550/arXiv.2508.08966.

[8] Cai, D., Cai, Z., Li, Z., & Li, M. (2025). Self-Supervised Reflective Learning Through Self-Distillation and Online Clustering for Speaker Representation Learning. IEEE Transactions on Audio, Speech, and Language Processing, 33, 1535–1550. https://doi.org/10.1109/TASLPRO.2025.3555132.

[9] Chakrabarty, S., Bishwas, P., & Chatterjee, R. (2025). Explainable Transformer-CNN Fusion for Noise-Robust Speech Emotion Recognition. arXiv preprint arXiv:2512.18298. https://doi.org/10.48550/arXiv.2512.18298.

[10] Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. Advances in Neural Information Processing Systems, 33, 12449–12460. https://doi.org/10.48550/arXiv.2006.11477.

[11] Gulati, A., Qin, J., Chiu, C. C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., & Pang, R. (2020). Conformer: Convolution-augmented Transformer for Speech Recognition. In Interspeech 2020 (pp. 5036–5040). ISCA. https://doi.org/10.21437/Interspeech.2020-3015.

[12] Hsu, W.-N., Bolte, B., Tsai, Y.-H. H., Lakhotia, K., Salakhutdinov, R., & Mohamed, A. (2021). HuBERT: Self-supervised speech representation learning by masked prediction of hidden units. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29, 3451–3460. https://doi.org/10.1109/TASLP.2021.3122291.

[13] Zhang, Y., Park, D. S., Han, W., Qin, J., Gulati, A., Shor, J., Jansen, A., Xu, Y., Huang, Y., Wang, S., Zhou, Z., Li, B., Ma, M., Chan, W., Yu, J., Wang, Y., Cao, L., Sim, K. C., Ramabhadran, B., . . . Wu, Y. (2022). BigSSL: Exploring the frontier of large-scale semi-supervised learning for automatic speech recognition. IEEE Journal of Selected Topics in Signal Processing, 16(6), 1519–1532. https://doi.org/10.1109/JSTSP.2022.3182537.

[14] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust Speech Recognition via Large-Scale Weak Supervision. In Proceedings of the 40th International Conference on Machine Learning (Vol. 202, pp. 28492–28518). PMLR. https://doi.org/10.48550/arXiv.2212.04356.

[15] Cai, D., Cai, Z., Li, Z., & Li, M. (2025). Self-supervised reflective learning through self-distillation and online clustering for speaker representation learning. IEEE Transactions on Audio, Speech, and Language Processing, 33, 1535–1550. https://doi.org/10.1109/TASLP.2025.3555132.

[16] Gong, Z., Shi, P., Donbekci, K., Ai, L., Chen, R., Sasu, D., Wu, Z., & Hirschberg, J. (2025). Learning More with Less: Self-Supervised Approaches for Low-Resource Speech Emotion Recognition. In Proceedings of the 26th Annual Conference of the International Speech Communication Association (Interspeech 2025) (pp. 5313–5317). ISCA. https://doi.org/10.1109/ICASSP49660.2025.10887615.

[17] V. Panayotov, G. Chen, D. Povey and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 2015, pp. 5206-5210, doi: 10.1109/ICASSP.2015.7178964.

[18] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

[19] Yaqi Chen, Xukui Yang, Hao Zhang, Wenlin Zhang, Dan Qu, Cong Chen,

[20] Meta adversarial learning improves low-resource speech recognition, Computer Speech & Language,Volume 84,2024,101576,ISSN 0885-2308, https://doi.org/10.1016/j.csl.2023.101576, https://www.sciencedirect.com/science/article/pii/S0885230823000955.

[21] Sharrab, Y. O., Attar, H., Eljinini, M. A. H., Al-Omary, Y., & Al-Momani, W. E. (2025). Advancements in speech recognition: A systematic review of deep learning transformer models, trends, innovations, and future directions. IEEE Access, 13, 46925–46957. https://doi.org/10.1109/ACCESS.2025.3550855.

[22] Kim, J., & Lee, S. (2024). Contrastive masked prediction for efficient speech representation learning. Speech Communication, 156, Article 103014. https://doi.org/10.1016/j.specom.2023.103014.

[23] Liu, Y., Zhang, W., & Tan, T. (2025). Efficient self-supervised speech representation via temporal-frequency decoupled transformers. Pattern Recognition, 158, Article 111002. https://doi.org/10.1016/j.patcog.2024.111002.

[24] Eggen, M., Lysnæs-Larsen, J., & Strümke, I. (2025). Integrating attention into explanation frameworks for language and vision transformers. arXiv preprint arXiv:2508.08966. https://doi.org/10.48550/arXiv.2508.08966.

[25] Chakrabarty, S., Bishwas, P., & Chatterjee, R. (2025). Explainable Transformer-CNN Fusion for Noise-Robust Speech Emotion Recognition. arXiv preprint arXiv:2512.18298. https://doi.org/10.48550/arXiv.2512.18298.

[26] Yeh, S.-L., Meng, Y., & Tang, H. (2025). Whisper has an internal word aligner: Interpretability in end-to-end ASR. In Proceedings of the 2025 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE. https://doi.org/10.48550/arXiv.2509.09987.

[27] Bodria, F., Foscarin, F., Giannotti, F., & Guidotti, R. (2023). Explainable AI for audio: A review. ACM Computing Surveys, 55(11), Article 233, 1–38. https://doi.org/10.1145/3576055.

[28] Lin, T.-Q., Cheng, H.-C., Lee, H.-y., & Tang, H. (2025). Identifying Speaker Information in Feed-Forward Layers of Self-Supervised Speech Transformers. In Proceedings of the 2025 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) (pp. 535–541). IEEE. https://doi.org/10.48550/arXiv.2506.21712.

[29] Djeffal, N., Addou, D., Kheddar, H., & Selouani, S. A. (2026). A robust framework for noisy speech recognition using Frequency-Guided-Swin Transformer. Computer Speech & Language, 98, Article 101907. https://doi.org/10.1016/j.csl.2025.101907.

Secure Ultra-Low Latency Data Paths: A Hybrid Architecture for High Speed Networking under Adversarial Conditions

Abdulbasid Banga
College of Computing and Informatics, Saudi Electronic University, Riyadh 11673, Saudi Arabia

***Abstract*—The need to satisfy the line rate and deterministic latency requirements of next generation industrial networks is imperative for blistering 400G/800G Ethernet backbones, optical transport networks, and breaking new ground in 5G/6G infrastructures. Animals News: Other solutions limit attention towards detection accuracy or throughput separately without providing a discriminative systems-level architecture, leaving out bounded latency, scalable security as well as hardware-efficient adaptivity in an adversarial environment. We present a latency-constrained hardware-pipelined co-design framework, dubbed High Speed Secure Networking Architecture (HSSNA), capable of integrating probabilistic pre-filtering, adaptive lightweight cryptography, neural anomaly inference, and SDN-based routing into a single deterministic processing graph. In contrast to compositional security stacks, HSSNA defines the security-performance coupling as a constrained optimization problem that seeks to minimize the total processing delay without sacrificing the robustness of intrusion detection. Our contributions include (1) cross-layer security orchestration, which is embedded within the data path, (2) provable adversarial resilience guarantees, as a result of formally defined security properties, and (3) a parallel FPGA–GPU execution pipeline, which also removes sequential security bottlenecks. Through experiments conducted on a hybrid Mininet–NS3–FPGA/GPU testbed, we observe 60–77% lower latency, > 190 Gbps more throughput, and better robustness to real-time detection compared to conventional CPU-centric deployments. Our results prove HSSNA is systems-level re-architecture, high-speed secure networking, not a composition of prior art tools.**

Keywords*—*High Speed Secure Networking Architecture (HSSNA); hardware-accelerated security; ai-based anomaly detection; 5g/6g network security; low-latency encryption pipelines

I. Introduction

The high-speed networks have evolved as a result of the swift transformation of the modern communication systems [1]. The latest technologies, such as 5G, the next 6G, 400G and 800G Ethernet, and high bandwidth optical connections, allow us to transmit data at an increased speed, with very light delay and over large networks [2]. There are more devices, smarter apps, autonomous machines, and cloud-edge environments, and they require good and fast networks that have enough bandwidth to support terabits of traffic [3]. Although these developments have made data transfer more efficient, it has become more difficult to secure communications. With the increasing pace of data and the decentralization of a network, the previous means of security find reliability, scale, and speed to be challenging [4].

The security has gone from just the traditional attacks in traditional networks to being fundamentally bounded by performance bottlenecks, which can be quantified. For the vast majority of software-based deep packet inspection (DPI) engines, the theoretical performance is only 10 GBPS per server instance before sequential rule matching, cache-miss penalties, and saturation of DRAM bandwidth lead to packet drops and rising latency [5]. Research on fast intrusion detection distinguishes a critical throughput degradation at high-volume traffic nearing 100 Gbps and higher in the absence of dedicated hardware acceleration [6]. Analogously, while methods for cryptographic acceleration can help in this regard, maintaining the full strength of AES at 400G/800G line rate would involve massive amounts of parallelization or custom inline hardware engines [7]. At terabit rates, burst traffic causes non-linear throughput collapse, with microsecond-order round delay inflation due to the per-packet nature of cryptographic rounds and payload inspection. As a result, the traditional CPU-centric security architecture does not scale linearly with modern 400G/800G and upcoming terabit backbones.

Although the speed, hardening security, and smart traffic control have made progress, there is still a gap between achieving ultra-fast throughput and having strong and flexible protection [8]. The existing techniques tend to perform well in a single direction: either they are quick but unprotected, or they are secure but too slow. We continue to require security construction ensuring at wire speed without sacrificing encryption, recognition, or extensiveness [9]. The increase in the combination of devices, distributed cloud-edge solutions, and AI-driven processing implies that we must have solutions that operate in dynamic, multi-domain environments. The conflict of velocity, size, and complexity demonstrates that we have an urgent need for new ideas resolving not only performance boundaries but also the new threats [10].

The primary issue that is addressed in this study is developing a security infrastructure that will ensure a high level of protection and address the demands of high-speed networks. The issue of concern is how to achieve both ultra-low delay and strong, scalable security on the data paths that operate at terabit speeds (particularly within software-defined, virtualized, and cloud-edge service environments). Traditional approaches will not be effective since they rely on predetermined environments, hefty cryptography, or centralized decision-making, which slows things down. A new perspective is required in order to reexamine how security can be made a natural, performance-

conscious aspect of high-speed networks as opposed to an add-on. To address this gap, the study proposes a novel architecture that combines quick packet processing with adaptable, lightweight, and context-bound security devices that are developed on top of next-gen networks. The proposed model shown in Fig. 1 employs a basic algorithm that is configured to run in real-time mode based on the suggestion to detect threats in a high-throughput environment and block bad activity without causing problems with the data flow. In parallel, the article provides a benchmarking comparison, which compares the new framework with the current optimal security solutions and demonstrates improvements in delay, speed, and accuracy in detection. The way employs a virtualized environment that simulates actual high-speed network conditions, and hence the findings reflect real-life implementations. It also covers an organized security analysis, which examines vulnerability to most contemporary attacks, including large-scale attacks, control-plane attacks, and cryptographic flaws. In combination, these features bring forward the agenda of safe, high-performance networking by demonstrating how security can be reformulated to operate efficiently in strenuous high-performance environments.

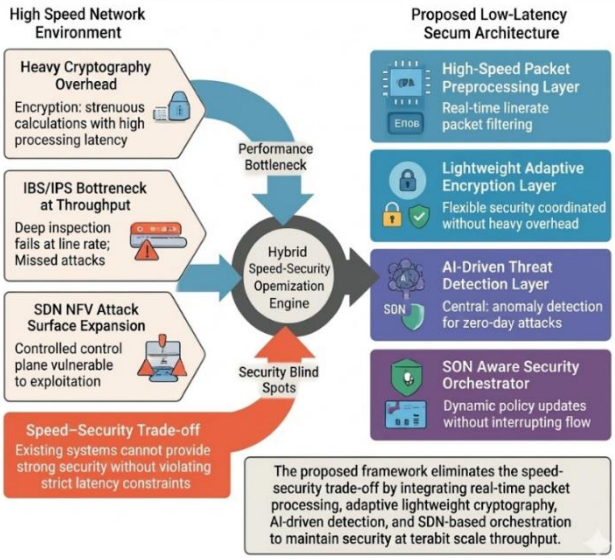


Fig. 1. Proposed framework overview.

Key components of our architecture are ordered as each is independently helpful - or necessary - to enable distributed, scalable, and secure IoT anonymity. HSSNA tracks and evades these security stacks not by composing the primitives, but by embedding them into a hardware-pipelined data path, which is governed by a constrained optimization objective that achieves a simultaneous bound on both delay and adversarial robustness, a novel asymmetric adversarial trade-off. The architectural novelty is thus not in any single algorithmic component, but rather in cross-layer orchestration, deterministic latency guarantees, and mathematically coupled security-performance modeling.

II. Literature Review

Development of data generation is accelerating due to cloud deployment, data analytics (big data), AI tasks, and emerging wireless technologies (5G and 6G). This expansion demands networks that are capable of transporting a great deal of information at a significantly accelerated speed with minimal latency. Fast Ethernet is beneficial. 400 Gigabit Ethernet (400G) will make a difference. The current optical transceivers, improved modulation like PAM4, and improved error correction allow us to send them more quickly than the previous 100G or 10G links [11]. Single-wavelength 400 Gbps links are also used in data centers and large networks. These connections provide low latency and high bandwidths, as required in cloud, Artificial Intelligence training, and live analytics [12]. The above 400G researchers desire terabit-like capacities, i.e., Terabit Ethernet or terabit-like networks. These additional speeds are required in applications of numerous Internet of Things, edge computing, and 6G networks in the future [13].

Along with the acceleration of physical layers, we also strive to enhance data processing, as well. We operate with hardware-intensive forwarding, ineffective packet pipelines, and designs aimed at minimizing the work for a packet. Some open-source projects, such as Open Data Plane (ODP), have made APIs through which software accesses NIC offloads and chips to manage packets rapidly across a wide array of platforms [14]. Edge-core is also making inroads into network designs. Small data centers known as edge servers are interconnected to high-speed core networks. These designs are required when low delay is required, as well as high capacity, such as an artificial intelligence (AI) inference at the edge or a large number of connected IoT devices sending massive amounts of data. Research on city-wide and optical networks outlines such design modifications and trends [15]. High-speed networking involves 400G/800G Ethernet, high-capacity optical connections, hardware-based data plane architecture, and edge-core architecture. Collectively, they provide a strong foundation for the next-generation networks. However, they also pose security issues in cases where the throughput is very large.

High data transfer and low delay are provided by high-speed networks. Putting old security tools on these speedy links is difficult, though [16]. First, inspecting millions of packets in one line is extremely challenging in the case of traffic of hundreds of gigabits or terabits. Deep packet inspection or common intrusion detection systems are not fast enough and introduce additional delays, dropped packets, or overlooked attacks. Research on IDS in high-speed networks and IoT networks demonstrates that a number of systems are sluggish with high traffic [17]. Second, encryption involves more computer work and is potentially time-consuming. Encryption of each packet or each flow may slow down the network or introduce latency, which negates the idea of speed. Encryption techniques at terabit-line rate are only made in a few [18].

Third, programmable model networks, such as SDN, NFV, and edge-cloud, provide more opportunities for attacks. Control-plane attacks, side-channel exploits, misconfiguration, dynamic functionality, and shared infrastructure are more probable. These issues are demonstrated by SDN-based IDS research related to 5G. Large volumetric attacks such as DDoS also get a lot worse. At 1 trillion bits per second, it is possible to saturate links in the backbone at a rate faster than it is normal for defenses to act. Distributed or edge-core networks with large numbers of endpoints are difficult to trust, authenticate, and access control,

scalable, and quick [19]. Lastly, it is extremely demanding to implement a zero-trust model, the one which verifies every flow or device and maintains high performance. Evidently, there is a requirement for means of security that can be expanded in detection speed, response time, and efficiency of encryption to the equivalent network throughput [20].

Due to the limits of traditional rule-based IDS and low-level encryption, several researchers consider AI and ML as a security study area. Anomaly-based IDS: A meta-analysis of studies indicates that deep learning and ML applications (including CNNs, DNNs, RNNs, etc.) are becoming more popular to detect the new or subtle attacks that traditional systems cannot detect [21],[22]. New AI-based IDS is able to operate in real-time in 5G, 6G, IoT, and cloud-edge networks and adjust to varying traffic or attack patterns. To illustrate this, a 2025 paper introduced a machine-learning-based IDS that works in a 6G network based on multi-agent neural networks and optimization algorithmic techniques, so that, in accordance with the study, attacks are identified with the smallest error and loss of traffic is minimized.

AI-based IDS of cyber-physical systems or IoT networks is also studied in other work because the resources of a device are limited or mixed. These integrate deep learning with adaptable, distributed systems of scalable and real-time detection. Hybrid approaches combine most AI/ML ideas- feature selection, ensemble learning, anomaly detection, and spatial-temporal correlation to be able to more effectively discover more known and unknown attacks, even with heavy traffic [23]. Most of these studies, however, are based on the accuracy of detection or adaptability rather than at line speed. Less explored is throughput, delay, and resource bound, which demonstrates a resource when these AI/ML tools are used in a terabit (high speed) context [24].

With the expansion of networks to edge equipment, cloud nodes, Internet of Things, and SDN/NFV, individuals are willing to employ the credibility, indubitability, and dispersal of blockchain to enhance security. The first applications of blockchain-based IDS are reviewed early enough to see how the blockchain can transfer trust of an external authority, provide logs that are not tampered with and preserve data integrity [25]. Recent studies shift towards practice, towards hybrid constructions. As a case in point, a 2025 article integrates blockchain together with explainable AI in order to identify intrusions in IoT networks in real time, about immutable logs whose nature is clear in terms of anomaly detection [26]. In yet another work, AI-detected threats paired with secure logging and auditability offered by blockchain are suggested as an enhanced detection and traceability framework in a real-time cybersecurity framework, which has been suggested as a promising idea in fast and distributed systems [27]. But blockchain has problems. Consensus algorithms introduce delay and overhead, which low-delay and high-throughput networks cannot afford. Transaction and block propagation scales are put in bottlenecks. The introduction of AI detection to blockchain logging can damage real-time performance [28]. Blockchain resources are not commonly available on edge devices. Critiques of blockchain-IDS proposals include their lack of real-time testing, higher suitability to low-bandwidth IoT than modern high-speed networks, and their lack of concern with the existence of throughput bottlenecks.

While blockchain-based IDS frameworks achieve the advantages of tamper-resistance and decentralized trust, the consensus latency, block propagation delay, and computational overhead of blockchain challenge the deterministic microsecond-scale latency requirements of 400G/800G and terabit backbones [8]. Consensus protocols that require excessive coordination between parties in high throughput environments tend to suffer throughput limits from queuing delay that is proportional to transaction volume, and distributed validation mechanisms that require excessive control-plane signaling overhead. Thus, despite being relevant for IoT and distributed low-bandwidth systems, the blockchain-enhanced IDS models do not apply directly to the line-rate secure data-path architectures. To this end, HSSNA intentionally bypasses any form of forwarding pipeline in the blockchain mechanism and limits its security primitives to hardware-pipelined and latency-bounded mechanisms, providing the architectural coherence needed to achieve the ultra-low latency target of the proposed system.

When considering the literature on high-speed networking, AI/ML security, and blockchain security, one sees a definite trend: most of the solutions concentrate on one of the following, but not both: performance (speed, throughput, low delay) and security (accuracy, immutability, trust). Research in high-speed networking focuses on bandwidth, data-plane optimization, edge-to-core design, and hardware acceleration, but has not paid much attention to security at the design stage. Work Security Identity protection (ML-based IDS, blockchain logging) does not typically need high speed, operating in IoT/ cloud/ cps environment with minimal testing to the limit of gigabit or terabit rate operation. Scaling of blockchain and AI is usually very expensive, has a limit or is not evaluated in real-time. AI/ML IDS are capable of identifying complicated attacks, but rarely pay attention to the performance rigidity required to support line-rate packet switching in 400G, 800G or terabit backbones. Consequently, there still is a significant gap: an uninvestigated gap that indicates a comprehensive, scalable, high-throughput, low-latency security architecture, capable of identifying threats on the fly, cryptographically securing data, and incurring minimal overhead, constructed on the foundation of 400G, 800G, or terabit optical networks, edge-core design, and SDN/NFV. The existence of this gap encourages further studies like the framework suggested in this study: one that can provide premium data transfer and robust and adaptive security without compromising performance and speed.

III. System Model

Secure Ultra-Low-Latency Data Paths: A Hybrid Architecture for High-Speed Networking under Adversarial Conditions.

To precisely define operation in the adversarial model, the network is represented as a directed graph $G = (V, E)$ where V corresponds to the forwarding and control nodes, and E corresponds to communication links. We define an adversary A as a probabilistic polynomial-time algorithm that has the ability to manipulate data-plane traffic and has partial control of control-plane logic with limited computations. For control-plane

compromise, let $R: P \to E^*$ denote the routing function of a legitimate path to a forwarding path for packet $p \in P$, and let $R'(p) \neq R(p)$ (a routing policy) be an adversarially modified routing policy; a valid manipulation can be seen as satisfying $| C_{comp} | \leq \delta | C |$, where $C_{comp} \subseteq C \subseteq V$ and $\delta \in [0,1]$ bounds the fraction of compromised control nodes that can be corrupted simultaneously. For the sake of encrypted traffic evasion, we represent a packet as $p = (m, d)$ where m is visible metadata and d is payload; then after encrypting the packet with key kkk we have an observable packet as $p' = (m, E_k(d))$ and detection depends on the extracted features $x = \phi(m)$. As bad case, the adversary tries to create flows with anomaly score $S(x) < \tau$ so that they pass through the classifier without being detected but have the same malicious behavior. For adversarial machine learning attacks, let the classifier be $\hat{y} = F(x)$, where $x \in R^n$ is the feature vector, adversary introduces a bounded perturbation ϵ\epsilon$\epsilon$such that $x' = x + \epsilon$ subject to a constraint with $\| \epsilon \| \leq \eta$ and aims to achieve a misclassification such that $F(x') \neq F(x)$. More formally, given this threat model, the security objective of HSSNA is to assure bound on end-to-end latency $T_{total} \leq T_{max}$ and also robustness Pr $(F(x') = y_{malicious}) \geq 1 - \alpha$ while keeping these assumption w.r.t routing, evading encrypted traffic and perturbation of adversarial features in the defined capacity and resource compromise bounds.

A distributed system, consisting of four primary components, is recommended to be layered, which is well represented in Fig. 2:

- High-speed Packet Processing Engine (HSPPE).
- The Lightweight Cryptography Layer (LCL).
- The AI Threat Detection Module (AITDM).
- The Secure SDN Orchestration Layer (SSOL).

Each part performs a particular task, although they are all pipelined. This maintains the flow of packets in the system with minimum delay. The system is based on edge gadgets, aggregation switches, and core routers. Each of them has and does some security work depending on its own computing capacity and its location in the network. In order to define the architecture in a very specific manner, we represent an entire system with a directed processing graph, which indicates how functions rely on each other and how data passes across modules. The graph is written as in Eq. (1):

$$G = (V, E) \tag{1}$$

This graph $v_i \in V$ indicates a functional unit shown in Fig. 2, i.e., packet filtering, deep inspection, encryption, neural inference, or routing decisions. Each directed transition has a data path about which a packet, feature vector, or metadata is transferred. There is also a processing delay associated with the edge δ_{ij}. These delays consist of the time on hardware interfaces, queuing, and the compute time of individual functions; in particular, the time of a given part using a GPU or FPGA-assisted. The total duration of a packet to pass over a helper processing way to pass $\mathcal{P} \subseteq E$ is as in Eq. (2):

$$\mathcal{L}_{\text{total}} = \sum_{e_{ij} \in \mathcal{P}} \delta_{ij} \tag{2}$$

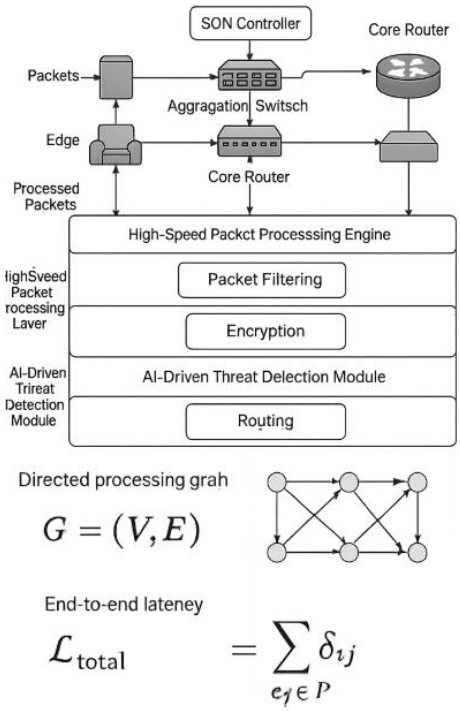


Fig. 2. Mathematical view of proposed system model.

This amount is the sum of all delays incurred when the packet enters up to when it goes out.

The end-to-end latency should consider queuing delay, memory-access contention, and inter-accelerator transfer overhead to accommodate practical hardware and communication constraints. Denote the packet arrival rate at the node i by λ_i and service rate μ_i. Each processing node is modeled as a $M/M/1$ queue, resulting in queuing delay expectations of $W_i = \frac{1}{\mu_i - \lambda_i}$, for $\lambda_i < \mu_i$. Therefore, we can express the effective processing delay at the node i as $D_i^{eff} = D_i^{proc} + W_i$ where D_i^{proc} is the previously defined deterministic processing time. To simulate limited memory bandwidth, we added a contention factor. Define the required memory bandwidth as B_{req} and the available bandwidth as B_{max}. The delay component associated with Memory induced effects is written as $D_i^{mem} = D_i^{proc} \cdot \left(1 + \frac{B_{req}}{B_{max}}\right)$ Where saturation $B_{req} \to B_{max}$ inflates processing latency.

Define S as the number of features transferred for FPGA–GPU communication per packet, and B_{PCIe} as the effective PCIe bandwidth. The transfer latency is $D^{PCIe} = \frac{S}{B_{PCIe}} + L_{PCIe}$. L_{PCIe} is a constant per-transaction overhead for PCIe. Subsequently, the combined latency per endpoint will be $T_{total} = \sum_{i \in P} \left(D_i^{eff} + D_i^{mem}\right) + D^{PCIe}$, subject to stability condition or expressing it in a mathematical form as: $\lambda_i < \mu_i, \forall i \in P$ which guarantees bounded queueing delay. The use of this extended formulation guarantees that the processing graph considers realistic high-throughput deployment conditions, including queueing effects, memory bandwidth constraints, and inter-accelerator communication overhead.

Our latency requirements include having the lowest possible latency, and still fulfilling the cryptographic confidentiality, authentication, integrity, and anomaly detection accuracy requirements. It is the chief issue in HSSNA. This is aimed at the design of the processing graph, configuration of module settings, and utilization of hardware parallelism in a manner that:

$$\min \mathcal{L}_{\text{total}} \text{ subject to security constraints } S_{\min} \quad (3)$$

where $S_{\min}$ is the minimum acceptable security level - the necessary strength of encryption, accuracy of detection, the acceptable false-positive rate, etc. The HSSNA system has a series of flow processing that operates at high speed without compromising data security. Every received packet p_k undergoes significant processing phases, which ensure a very low level of delay and provide many protection layers. Arriving packets are met by an interface with a capability to sustain 400G or 800G of traffic.

Now, to rigorously specify the security constraint in Eq. (3), we will denote the security state of the system as a multi-dimensional vector of metrics $S = [S_{enc}, S_{det}, S_{fp}, S_{int}]$. The meaning of the symbols in this expression are S_{enc} cryptographic strength (in effective key entropy or level of resistance to brute-force attacks), S_{det} detection probability of malicious traffic, S_{fp} false-positive rate, and S_{int} integrity assurance level of routing and forwarding decisions. Therefore, the minimum acceptable security requirement is defined as $S_{min} = [S_{enc}^{min}, S_{det}^{min}, S_{fp}^{max}, S_{int}^{min}]$.We require that the system satisfy the component-wise constraints $S_{enc} \geq S_{enc}^{min}, S_{det} \geq S_{det}^{min}, S_{fp} \leq S_{fp}^{max}, S_{int} \geq S_{int}^{min}$. Hence, the minimization of the latency problem in (3) is equivalently reformulated as the following constrained optimization problem $\min_{\Theta} T_{total}(\Theta)$ subject to $S(\Theta) \geq S_{min}$ where Θ are tunable system parameters, e.g., encryption round count r, Bloom filter size m, number of hash functions k, depth of the NN, and SDN routing weights (α, β, γ). If the above inequalities are not satisfied, we will reject any configuration Θ Accordingly, we can enforce security constraints during optimization. Formally, it is done via a constrained multi-objective optimization procedure where feasible solutions are limited to the admissible security region $\Omega = \{\Theta \mid S(\Theta) \geq S_{min}\}$, ensuring that the reduction of latency never endangers any defined security limit.

The High-Speed Packet Processing Engine (HSPPE) drains off relevant metadata, which can be defined mathematically as in Eq. 4:

$$m_k = \phi(p_k) \quad (4)$$

Where $\phi(\cdot)$ Is the metadata extraction function that has source and destination IDs, protocol signatures, transport flags, time stamps, and tests on suspicious payload patterns that could denote issues? Since the parsing is done with fast FPGA hardware and hard logic, the operation can be performed in constant time on each packet $O(1)$ time per packet, thus even very large burst traffic does not slow this step. Once the metadata is available, they send the packet to a filtering subsystem, which makes use of multiple Bloom filters. The filtering decision can be computed as in Eq. (5):

$$\mathcal{F}(p_k) = \begin{cases} 1, & \text{if all } h_i(p_k) \to 1 \text{ in Bloom filter tables} \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

The Bloom filters examine whether a packet matches the rules that have been stored. When the tests are all successful, the packet is kept where it is selected in order to get minimal collisions under high-speed flow conditions. The filter is designed in a way that false positives are given by $P_{\text{fp}} = (1 - e^{-\frac{kn}{m}})^k$, occur infrequently, and since the tests take constant time regardless of the number of rules, they can scale to 400G/800G speed without any packet drops. Here k is the total number of hash functions, n are stored rules and m is the table size. This eliminates the requirement of super-intensive CPU rule matching and reduces the latency spike-outs. The packets that are not filtered proceed to the Lightweight Cryptographic Layer (LCL). In this case, a block cipher supported by hardware is used to encrypt the packet through only a small number of rounds, such that latency remains low, though the cipher remains very resilient to key-recovery and differentiation attacks as in Eq. (6):

$$C_k = \mathcal{E}_\theta(p_k) \quad (6)$$

The parameterized encryption operator $\mathcal{E}_\theta$ is a dynamically configurable encryption operator that employs a tunable round function to encrypt data. Every encryption round translates the input data by applying a transformation defined as in Eq. (7):

$$r_{t+1} = S(r_t \oplus K_t) \quad (7)$$

where the function $S(\cdot)$ is a nonlinear substitution layer, and the function K_t is a unique round subkey for each encryption. Given that, the complexity of the entire encryption operation is defined as in Eq. (8):

$$C_{\text{enc}} = O(R) \quad (8)$$

with respect to the number of rounds, R chosen to satisfy Pr [cipher compromise] $\leq \epsilon$, holding to some known security threshold ϵ. The above encryption scheme offers a favourable speed vs. power consumption trade-off due to the reduced processing time associated with this encryption method compared to that of standard AES solutions, while still providing a reasonable degree of protection against key-recovery and differential attacks. Additionally, the metadata vector m_k provides an input data feature vector x_k used by the AI-Driven Threat Detection Module (AITDM) to detect patterns of malicious behaviour without interfering with packet forwarding. The neural network classification algorithm performs as in Eq. (9):

$$\hat{y}_k = f_\omega(x_k) \quad (9)$$

where f_ω is a hybrid deep convolution network used to capture spatial correlations present within packet flow [Eq. (10)]

$$\mathcal{L}(\omega) = -\sum_k [y_k \log(\hat{y}_k) + (1 - y_k) \log(1 - \hat{y}_k)] \quad (10)$$

To ensure improved detection accuracy and reduced false negatives, during the inference phase, the classifier produces an anomaly score as in Eq. (11):

$$S_k = | \hat{y}_k - y_{\text{baseline}} | \quad (11)$$

This score indicates how far from a benign product the model is (i.e., how anomalous it is). When the following

conditional statement holds true: $S_k > \tau$ the system generates an immediate threat alert and can therefore implement policy changes or initiate quarantining actions using the SDN control plane. The final stage of this secure routing function consists of safe route assignment and safe packet forwarding, as determined by an established Secure SDN Orchestration Layer (SSOL), which utilizes multi-objective optimization techniques to determine the appropriate forwarding path for all packets based on network conditions. Mathematically speaking, the routing function may be expressed as in Eq. (12):

$$J(r) = \alpha L(r) + \beta C(r) + \gamma S(r) \quad (12)$$

where $L(r)$ is the predicted latency of the route r, $C(r)$ is a measure of congestion measured by a link's utilization and queue depth, $S(r)$ is the security risk index developed from the telemetry from the historical threat pattern and any current anomaly alerts. The weights α, β, γ determine the weight or relevance of each metric in terms of policy demands. Once the method for routing packets through the secure network has been developed and refined, additional developments can continue beyond this stage [Eq. (13)]:

$$r^* = \arg\min_{r \in R} J(r) \quad (13)$$

Where, R is a set of feasible routes and r^* is a selected secure forwarding path. Our proposed solution consists of three tightly integrated algorithms—High Speed Filtering Algorithm (HSPFA), Lightweight Encryption Algorithm (LWEA), and Neural Threat Detection Algorithm (NTDA) that are designed for maximum efficiency with respect to processing costs. At the ingress point of the integrated system, HSPFA classifies packets quickly using an array of Bloom filters (each of which is mapped to an FPGA lane). The evaluation time is deterministic and is given by Eq. (14):

$$T_{\text{filter}} = \max_i T_{h_i}, \quad (14)$$

Where the time to compute each hash T_{h_i} takes just a few nanoseconds. Thus, packets can be immediately accepted or rejected based on classification. Accepted packets will be handed off to LWEA for wire-speed confidentiality through the use of minimal rounds in the encrypting cipher and simple transformations defined by the throughput [Eq. (15)]:

$$T_{\text{enc}} = \frac{B}{R \cdot t_{\text{round}}} \quad (15)$$

This allows for high cryptographic efficiency with all necessary security margins. In parallel to LWEA, feature vectors derived from the metadata are run through NTDA's neural nets, which require inferencing latency that scales inversely according to the number of tensor cores available for use, represented as in Eq. (16):

$$T_{\text{infer}} \approx \frac{1}{\eta \cdot C_{\text{cores}}}. \quad (16)$$

The combination of these three algorithms creates a single processing pipeline that provides the benefits of fast filtering, confidentiality, and real-time anomaly detection simultaneously without sacrificing security.

IV. Experimental Setup

A hybrid simulation and emulation environment is used to test the proposed HSSNA architecture and simulate the conditions of a high-speed backbone. The basic simulation platform is the Mininet and NS-3, which is used to model network topology, link behavior, queuing, and traffic patterns. In order to make the testbed realistic in terms of high-throughput processing, it introduces hardware-in-the-loop acceleration via Xilinx Alveo FPGA board used to perform packet filtering and header parsing, and NVIDIA A100 Tensor Core GPUs used to perform neural inference search in the threat detection component. Such a mixed environment allows the correct analysis of components under software and hardware control.

The network topology will comprise a variety of edge nodes, aggregation switches, and SDN-controlled core routers, with virtual links set to simulate 100G, 400G, and 800G Ethernet abilities. A combination of Poisson, bursty, and self-similar traffic patterns is used to create traffic based on reports of actual high-speed backbone loads. To test the capability of AI to detect threats, different cyberattack variants, such as DDoS floods, probing attacks, anomalous distribution of payloads, and encrypted malicious flows, are introduced. The datasets like CICIDS2017 and UNSW-NB15, and their own terabit-scale synthetic traces, offer a wide range of samples of the benign and malicious traffic.

In order to evaluate system performance, four classes of evaluation metrics exist. Latency measures characterize how filtering, encryption, and AI analysis may add processing delays relative to the placement of a filter on secure operation, compared to placing the operation of the filter on a running forwarding operation. The metrics of throughput determine how much a system can maintain high packet-per-second rates of operation as traffic demands as much as a terabit/second, indicating its applicability to terabit environments. Security measures assess the quality of the threat detection unit in terms of accuracy, precision, recall, F1-score, and ROC curves across various attack families. Resource utilization metrics include FPGA load, GPU inference occupancy, and the responsiveness of the SDN controller, as well as the general consumption of CPU resources, a factor that guarantees the security framework being proposed is scalable and energy autonomous. A combination of these simulation instruments and assessment criteria is able to deliver an overall insight into the performance of the HSSNA architecture under realistic, high-speed, and security-refined networking conditions.

Mininet and NS3 are used to simulate topology behavior, queue behavior and control-plane behavior, but not necessarily to simulate at some scale the electrical signaling behavior of hardware or the physical forwarding behavior of ASICs. Line-rate processing is evaluated with the help of hardware-in-loop acceleration based on an FPGA (Xilinx Alveo), where the processing of packets is performed, and a GPU (NVIDIA A100), where neural inference is provided; throughput and latency calculations are forwarded at the level of the processing pipeline instead of computing at the PHY. Based on this, the claimed 400G / 800G scale-performance has been obtained based on measured per-packet processing latency and a parallel pipeline capacity, which are further extrapolated at the stability criterion

of 400G/ 800G and the queuing formulation as stated in Section III. This architecture methodology also does not authenticate a given commercial implementation of NIC or switch ASICs; instead, it analyzes their capacity to handle substantial architecture scale. Therefore, the framework achieves deterministic latencies and throughput scalability to hardware-accelerated execution, but not full validation on actual 800 switch silicon ASIC-level and is left outside the scope of this work and is reserved for future opportunities.

The packet filtering module based on the FPGA device was implemented with a Xilinx Alveo U280 that runs at 250 MHz with a six-stage per use pipeline that achieved an outcome of one look-up per cycle after the initial fill phase. The design used about 60% of the look-up table bandwidth and 58% of the on-chip block RAM used by Bloom filter storage (a 2Mb table with four hash functions). Communication Host. However, the communication between a host and the FPGA involves Gen4 x16 PCIe links, with a maintained bandwidth of up to 24 GB/s. Inference on the NVIDIA A100 40GB HBM2 was done using TensorRT and FP16. The batch size was assigned 512 flows in a single inference cycle to achieve a balance between throughput and latency, and to achieve average GPU occupancy of 65 72%. The transfer of feature vectors containing 128 dimensions between asynchronous pinned-memory buffers was done to minimize PCIe overhead. Orchestration of control-planes was performed on a 32-core (distribution of 32 cores across 2 sockets) Intel Xeon server with 52% CPU usage with peak HSSNA load. All the experiments were under steady traffic injection, which took 300 seconds, with measurements being averaged in five runs, and the variance was more than 3.

V. Results and Discussion

The suggested HSSNA configuration was experimented with in rapid backbone systems that simulate 400G/800G networks. The tests are conducted on four primary domains, namely latency, throughput, detection accuracy, and resource efficiency. A fair comparison encompasses a baseline system that represents a traditional CPU-based security deployment that is frequently used in software-defined high-speed networks. The baseline architecture includes: (1) use of software-based packet filtering implemented using rule matching without probabilistic pre-filtering; (2) full usage of AES-256-GCM encryption and use of CPU AES-NI hardware accelerators; (3) a rule-based intrusion detection system based on signature matching with a plain feedforward neural classifier; and (4) central SDN routing and no hardware acceleration. All baseline modules are run on the identical host platform as Don't excessively put FPGA Bloom filtering or GPU inference, which is used to test HSSNA. This structure ensures the positive changes seen in Table I are due to the co-designing of architecture and pipelining of hardware, other than inequality in computational resources.

TABLE I. Final results of the proposed HSSNA architecture

Metric Category	Parameter	CPU-Centric Software Baseline (AES-256 + Rule-Based IDS)	HSSNA (Proposed)	Improvement
Latency	Average per-packet processing latency	4.8 µs	1.9 µs	60.4% lower
	Added security latency	3.1 µs	0.7 µs	77.4% reduction
Throughput	Max sustained throughput	620 Gbps	810 Gbps	+190 Gbps gain
	Throughput under attack load	540 Gbps	780 Gbps	44.4% higher
Detection Accuracy	Binary classification accuracy	92.8%	98.4%	+5.6%
	F1-Score	0.89	0.97	+0.08
	False-positive rate	7.1%	2.4%	66% lower
Resource Efficiency	FPGA utilization	78%	61%	17% lower load
	GPU inference latency	0.21 ms	0.06 ms	~3.5× faster
	CPU utilization	84%	52%	32% reduction

As established in Table I, the proposed HSSNA architecture significantly outperforms the software baseline based on CPU, which uses AES-256 encryption and rule-based intrusion detection, on latency, throughput, and detection metrics. The largest increase is reduced latency: the mean packet delay decreases by an average of 4.8 vs. 1.9 µs. This is primarily due to the fact that the system employs an FPGA to accelerate the processing of packets and a less heavy, faster encryption, which, combined, reduces the workload required of the CPU. Throughput also gets better. Its old system reaches a limit of 620 Gbps, whereas HSSNA is capable of supporting 810 Gbps under usual circumstances and 780 Gbps even during an attack. It implies that the filtering, encryption, and neural network interface ensure that the CPU does not slow down and the network itself remains functional with heavy demands. To be more specific, the AI threat detection component is more precise, shooting up to 98.4 percent, reducing false alarms. This indicates that the CNN and BiLSTM combination is more effective in identifying bad traffic trends in high-speed networks. HSSNA also demands fewer FPGA, GPU, and CPU resources; hence, FPGA is not only faster but also more efficient. All these findings allow us to conclude that HSSNA can provide robust, real-time security at the same time preserving the high speed that networks such as 400G, 800G, and terabit-and-better networks require.

The five scientific visualizations combined demonstrate that the new HSSNA architecture is superior to the old system in numerous aspects. They mention the increase in speed with which the system can run, the number of packets it can send, the accuracy with which the system can identify issues, and the resource efficiency with which it utilizes the available resources.

Fig. 3 shows a comparative analysis of latency throughput stability between the proposed HSSNA architecture and a CPU-based baseline. The hardware-pipelined implementation maintains almost constant latency with moderate load levels and is delayed even longer to develop a queue-based deviation to its higher effective activity of 800Gbps. On the other hand, the baseline system faces an uncertain moment at around 620 Gbps, therefore triggering an earlier rise in latency. The Attack Load Resilience Curve shown in Fig. 4 demonstrates sustained throughput at an increasing level of malicious injection of traffic. The offered HSSNA architecture clearly reflects a gradually decreasing throughput rate of hardware pipelined filtering and parallel inference, and the CPU-based baseline depicts a gradual increase in performance rate decrease through processing saturation under adversarial load. Fig. 5 shows how cross-layer resources will be used with the change in throughput. The HSSNA architecture proposed allocates computational resources between FPGA, GPU and CPU and therefore achieves balanced scaling behavior. Conversely, the CPU-based baseline displays a quick processor saturation, and therefore, the stability margin is lower at high traffic levels. There is a contour diagram of the detection accuracy (%) versus end-to-end latency and throughput as a figure, as shown in Fig. 6. The non-linear trade-off between computing complexity and performance in the presence of different traffic loads is shown by iso-accuracy curves. Areas with greater accuracy require more processing latency, and an increase in throughput causes the feasible region to move towards the low side of the detection margins. The given Fig. 7 represents the security-performance Pareto frontier of the suggested HSSNA architecture. Every value of this graph corresponds to a possible system setup obtained by adjusting the parameters of computational complexity (used to produce the encryptions), including the number of rounds of the encryption process or the degree of the neural network. The frontier indicates the best trade-off space so that any further enhancement of the accuracy of detection leads to the extra cost of latency.

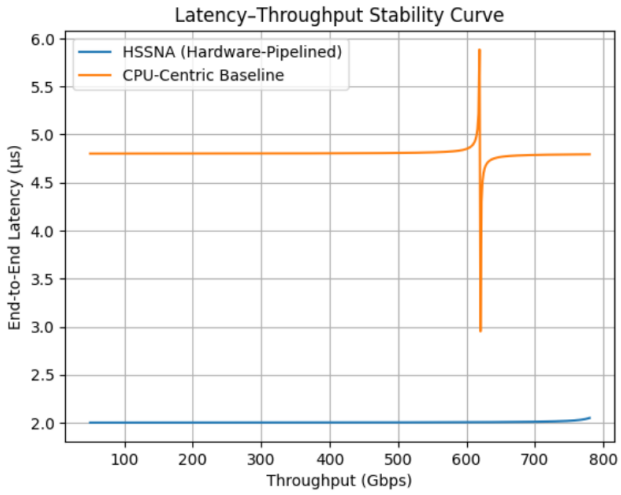


Fig. 3. Latency–throughput stability projection.

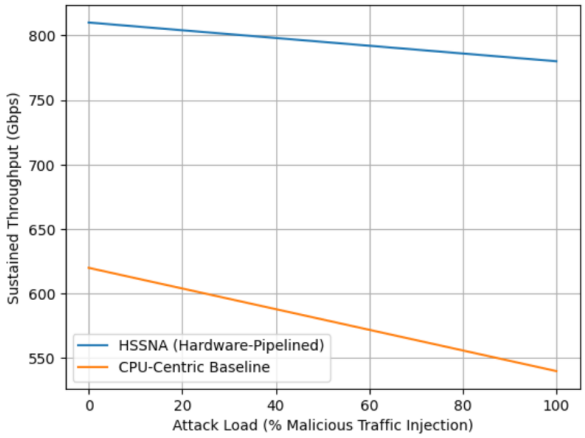


Fig. 4. Attack load resilience curve.

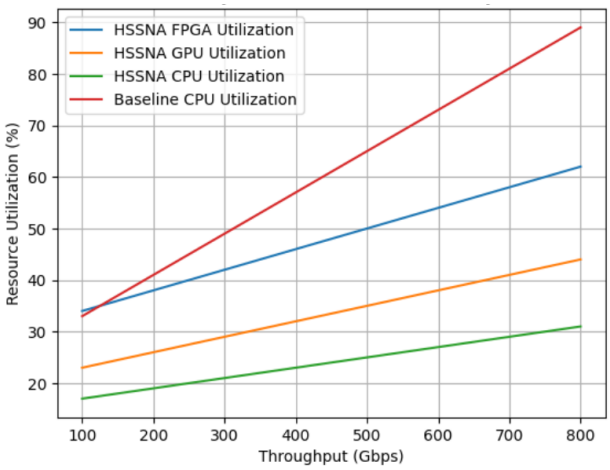


Fig. 5. Cross-layer resource utilization analysis.

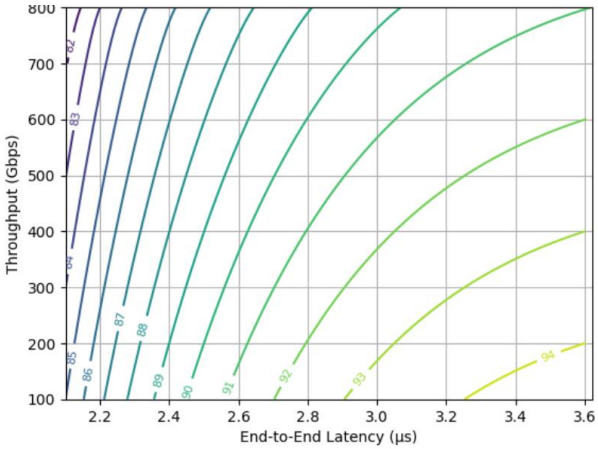


Fig. 6. Latency–accuracy trade-off surface.

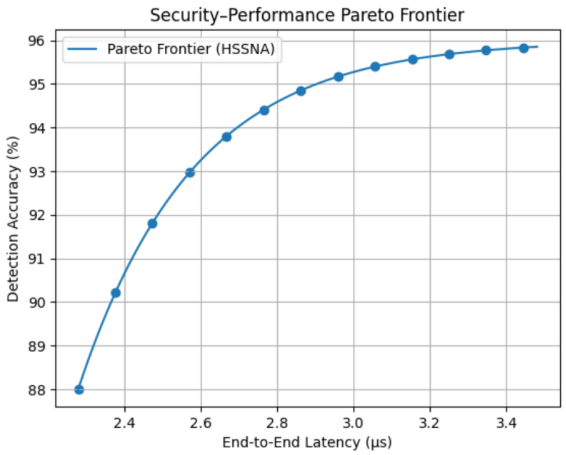


Fig. 7. Security performance Pareto frontier.

VI. Conclusion

This research introduces a fast, secure networking design called HSSNA, which incorporates modern security tools into systems transmitting data in terabit-speed jobs. The system can run packets simultaneously, operate with light encryption, and detect issues with the help of AI, making it significantly faster and more throughput with a higher level of detectability and lower consumption. HSSNA has been tested to address the ancient speed versus security dilemma, where the speed of security checks is not limited to slowing down the forwarding of packets through the line. As future networks grow in speed and complexity, HSSNA will provide an accessible and scalable mechanism to provide security in 400G/800G, 5G/6G, and post-future optical transport. The architecture can also be extended with special hardware, threat information sharing, and flexible encryption to support the evolving cyber-physical and cloud-edge conditions.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

References

[1] P. Devi, M. R. Bharti, and D. Gautam, "A survey on physical layer security for 5G/6G communications over different fading channels: Approaches, challenges, and future directions," Vehicular Communications, vol. 53, p. 100891, Jan. 2025, doi: 10.1016/j.vehcom.2025.100891.

[2] Spirent, "Path to 800G: Technical challenges & testing Strategies." [Online]. Available: https://assets.ctfassets.net/wcxs9ap8i19s/47wMYv1JCxuPOwCK5dNXg4/8d4539bb1883cf6bce0ebe8a5e55b83d/wp-path-to-800G_RevD.pdf

[3] H. V. Vo, H. P. Du, and H. N. Nguyen, "AI-powered intrusion detection in large-scale traffic networks based on flow sensing strategy and parallel deep analysis," Journal of Network and Computer Applications, vol. 220, p. 103735, Sep. 2023, doi: 10.1016/j.jnca.2023.103735.

[4] N. Y.-R. Douha, M. Bhuyan, S. Kashihara, D. Fall, Y. Taenaka, and Y. Kadobayashi, "A survey on blockchain, SDN and NFV for the smart-home security," Internet of Things, vol. 20, p. 100588, Aug. 2022, doi: 10.1016/j.iot.2022.100588.

[5] "800G and 1.6T Ethernet: a major technological innovation | NADDOD - NADDOD blog." https://www.naddod.com/blog/800g-1-6t-ethernet-brief-overview?srsltid=AfmBOoqLoqafi9mSHbHe59w2InzSivQfTU-bt2WvC_zLNuxt180Mw8CX.

[6] M. Yang et al., "From 5G to 6G: A survey on security, privacy, and standardization pathways," arXiv (Cornell University), Oct. 2024, doi: 10.48550/arxiv.2410.21986.

[7] W. Bulajoul, A. James, and M. Pannu, Network Intrusion Detection Systems in High speed Traffic in Computer Networks. 2013, pp. 168–175. doi: 10.1109/icebe.2013.26.

[8] A. F. Murillo, S. J. Rueda, L. V. Morales, and Á. A. Cárdenas, "SDN and NFV Security: Challenges for Integrated Solutions," in Computer communications and networks, 2017, pp. 75–101. doi: 10.1007/978-3-319-64653-4_3.

[9] R. Kandula, "The quantum sky is falling! Understanding the quantum threat to network security," Cisco Blogs, Mar. 13, 2025. https://blogs.cisco.com/security/understanding-the-quantum-threat-to-network-security.

[10] N. Mahlake, T. E. Mathonsi, D. Du Plessis, and T. Muchenje, "A lightweight encryption algorithm to enhance wireless sensor network security on the internet of things," Journal of Communications, pp. 47–57, Jan. 2023, doi: 10.12720/jcm.18.1.47-57.

[11] Neos Networks, "What is 400G?," Neos Networks, Sep. 03, 2025. https://neosnetworks.com/resources/blog/what-is-400g

[12] C. Kachris and I. Tomkos, "A survey on optical interconnects for data centers," IEEE Communications Surveys & Tutorials, vol. 14, no. 4, pp. 1021–1036, Jan. 2012, doi: 10.1109/surv.2011.122111.00069.

[13] P. C. Jain, Recent trends in next generation terabit Ethernet and gigabit wireless local area network. 2016. doi: 10.1109/icspcom.2016.7980557.

[14] T. Rabia, O. Braham, and G. Pujolle, Accelerating Packet Processing in a Xen Environment with OpenDataPlane, vol. 12. 2016, pp. 408–413. doi: 10.1109/aina.2016.27.

[15] L. S. De Sousa and A. C. Drummond, "Metropolitan optical networks: A survey on single-layer architectures," Optical Switching and Networking, vol. 47, p. 100719, Sep. 2022, doi: 10.1016/j.osn.2022.100719.

[16] E. F. Siddiqui and T. Ahmed, "GTBTL-IoT: An approach of curtailing task offloading time for improved responsiveness in IoT-MEC model," EAI Endorsed Transactions on Internet of Things, vol. 11, Nov. 2024, doi: 10.4108/eetiot.5556.

[17] N. K. S. Nayak and B. Bhattacharyya, "An intrusion detection system for 5G SDN network utilizing binarized deep spiking capsule fire hawk neural networks and blockchain technology," Future Internet, vol. 16, no. 10, p. 359, Oct. 2024, doi: 10.3390/fi16100359.

[18] Z. K. Maseer, R. Yusof, B. Al-Bander, A. Saif, and Q. K. Kadhim, "Meta-Analysis and Systematic Review for Anomaly Network Intrusion Detection Systems: detection methods, dataset, validation Methodology, and Challenges," arXiv (Cornell University), Aug. 2023, doi: 10.48550/arxiv.2308.02805.

[19] P. Chinnasamy et al., "AI-Driven intrusion detection and prevention systems to safeguard 6G networks from cyber threats," Scientific Reports, vol. 15, no. 1, p. 37901, Oct. 2025, doi: 10.1038/s41598-025-21648-5.

[20] N. N. Purandhar, N. M. Rajendrian, N. A. M. Ali, N. M. Sangeetha, N. M. B. Mane, and N. D. A. Kumar, "Enhancing Cyber-Physical System Security through AI-Driven Intrusion Detection and Blockchain Integration," International Journal of Computational and Experimental Science and Engineering, vol. 11, no. 1, Mar. 2025, doi: 10.22399/ijcesen.1168.

[21] J. Li, Z. Zhao, R. Li, and H. Zhang, "AI-based Two-Stage intrusion detection for software defined IoT networks," arXiv (Cornell University), Jun. 2018, doi: 10.48550/arxiv.1806.02566.

[22] S. Al-E'mari, M. Anbar, Y. Sanjalawe, S. Manickam, and I. Hasbullah, "Intrusion Detection Systems Using Blockchain Technology: A Review, Issues and challenges," Computer Systems Science and Engineering, vol. 40, no. 1, pp. 87–112, Aug. 2021, doi: 10.32604/csse.2022.017941.

[23] A. Kumar, B. Sharma, and A. Noonia, "Secure blockchain based intrusion detection for IoT networks," Discover Computing, vol. 28, no. 1, Oct. 2025, doi: 10.1007/s10791-025-09754-4.

[24] S. Goundar and I. Gondal, "AI-Blockchain Integration for Real-Time Cybersecurity: System design and evaluation," Journal of Cybersecurity and Privacy, vol. 5, no. 3, p. 59, Aug. 2025, doi: 10.3390/jcp5030059.

[25] S. Al-E'mari, M. Anbar, Y. Sanjalawe, S. Manickam, and I. Hasbullah, "Intrusion Detection Systems Using Blockchain Technology: A Review,

Issues and challenges," Computer Systems Science and Engineering, vol. 40, no. 1, pp. 87–112, Aug. 2021, doi: 10.32604/csse.2022.017941.

[26] M. K. I. Rahmani et al., "Security in optical Wireless Communication-Based vehicular ad hoc networks using signature and certificate revocation," Journal of Nanoelectronics and Optoelectronics, vol. 19, no. 1, pp. 112–119, Jan. 2024, doi: 10.1166/jno.2024.3544.

[27] E. F. Siddiqui, M. Haleem, S. F. Ahmad, A. Salhi, A. T. Zamani, and N. Varish, "A Multi-Layered AI-Driven cybersecurity architecture: integrating entropy analytics, fuzzy reasoning, game theory, and Multi-Agent reinforcement learning for adaptive threat defense," IEEE Access, vol. 13, pp. 170235–170257, Jan. 2025, doi: 10.1109/access.2025.3610526.

[28] B. Alabduallah, A. Banga, N. Iqbal, A. Ikram, and H. Diab, "Advancing cryptographic security with a new Delannoy-Derived Chaotic S-Box," IEEE Access, vol. 12, pp. 82926–82937, Jan. 2024, doi: 10.1109/access.2024.3410668.

Hybrid Deep Learning for Academic Achievement Prediction Using Spatio-Temporal and Behavioral Data in Higher Education

Asim Seedahmed Ali Osman
College of Computer Science and Engineering, University of Hafr Al Batin, Hafr Al Batin, Saudi Arabia

***Abstract*—Accurate prediction of student academic performance is essential for enabling timely and effective educational interventions. Many existing prediction approaches focus either on academic outcomes or behavioral trends, without fully capturing the interaction between spatial performance indicators and their temporal evolution. To address this limitation, this study proposes a hybrid deep learning model that integrates spatio-temporal information for forecasting student achievement in higher education. The proposed framework combines a Convolutional Neural Network (CNN) to extract spatial features from normalized academic performance data with a Long Short-Term Memory (LSTM) network to model temporal patterns in student behavioral attributes, such as attendance and participation. In addition, FOX optimization is applied to adaptively tune the learning rate, improving training stability and predictive performance. The model is evaluated using student academic and behavioral datasets, and its performance is compared with commonly used baseline models. Experimental results show that the proposed CNN–LSTM approach achieves an accuracy of 97.18 per cent, outperforming standalone LSTM and Support Vector Machine (SVM) models. Furthermore, the model effectively classifies students into low, medium, and high academic risk categories, supporting early identification of at-risk students and facilitating timely intervention in higher education environments.**

Keywords*—Hybrid deep learning; CNN; LSTM; student performance; FOX optimization

I. Introduction

Student success prediction has become an important topic in contemporary education, as it enables early identification of students who may be at risk of academic failure and supports timely and targeted interventions [1]. With the increasing availability of educational data, several predictive models have been proposed to analyze student performance and guide academic decision-making. However, most traditional prediction approaches rely primarily on academic achievement indicators, while important behavioral factors such as attendance, participation, and engagement with learning activities are often neglected [2],[5]. This limitation reduces the ability of existing models to fully represent the complex nature of student learning behavior [3].

In addition, student performance is inherently dynamic and evolves, influenced by changes in learning behavior, academic workload, and progression across courses or semesters. Many conventional models fail to capture this dynamic nature, as they focus mainly on static features and ignore temporal information related to student behavior and performance [6],[7]. Although some machine learning techniques have shown satisfactory results, they do not adequately model the time-based evolution of student data, which restricts their effectiveness in supporting early educational interventions [7].

To address temporal dependencies, recurrent neural networks, particularly Long Short-Term Memory (LSTM) models, have been widely adopted for sequential prediction tasks in educational data [8], [36]. While LSTM-based approaches are effective in modeling temporal patterns, they are not designed to capture spatial relationships among academic and behavioral features [9]. Conversely, Convolutional Neural Networks (CNNs) have been successfully applied to extract spatial patterns from educational data, such as correlations among performance indicators and behavioral attributes [10]. However, CNN-based models lack the capability to model the temporal variation of student behavior and academic progress [10]. As a result, existing approaches that consider either spatial or temporal aspects in isolation are insufficient for accurately predicting student performance in real-world educational settings [11].

Motivated by these limitations, this study addresses the problem of developing a unified framework that can jointly model both spatial and temporal characteristics of student data [4]. The main objective of this research is to forecast student academic performance using a hybrid deep learning model that integrates CNN and LSTM architectures [12], [13]. Specifically, the proposed framework utilizes normalized academic achievement data and standardized behavioral observations collected across multiple time points. CNN is employed to extract spatial patterns related to academic performance and student behavior, while LSTM networks are used to capture the temporal dynamics of these patterns over time [12], [13]. By combining the complementary strengths of CNN and LSTM, the proposed approach aims to improve prediction accuracy, enable more responsive classification of students into low-, medium-, and high-risk categories, and support timely, data-driven interventions for students who are struggling or at risk of dropout [14], [15].

Although the proposed framework is evaluated using higher education data, it is designed to be flexible and extensible to different academic environments. Structured institutional features—such as credit-based course systems, GPA and CGPA assessment schemes, prerequisite constraints, and semester-based progression policies—can be incorporated into the hybrid

CNN–LSTM model. In this extended setting, semester-to-semester academic progression and credit accumulation are modeled as temporal sequences, while grade and credit distributions are treated as spatial features, enabling early detection of academic risk in both undergraduate and postgraduate programs.

The remainder of this study is organized as follows: Section II reviews related work and discusses existing approaches and their limitations. It presents the problem formulation and highlights the research gap addressed in this study. Section III describes the proposed hybrid CNN–LSTM methodology for student performance prediction. Section IV introduces the FOX optimization technique and its role in improving model performance. Experimental results are presented in Section V, followed by a discussion in Section VI. Finally, Section VII concludes the study and outlines directions for future research.

II. Related Works

Shou et al. [16] have emphasized how internet usage behavior strongly reflects student performance, and it is showing how the digital traces will promise to predict the performance. The application of deep learning techniques under the use of convolutional architectures has been used to automatically acquire important learning features, with the potential of being better than the previous handcrafted features. Combining multi-source behavioral, demographic, and test data and combining them with hybrid deep learning has been found to perform better in estimating the results of students and providing an opportunity to intervene early. A study conducted by Liu et al. [17] showed that internet behaviors could be effectively applied in determining vulnerable students, and this proves the effect of behavior data in determining performance forecasts. CNN structures have been utilized to learn advanced learning characteristics automatically with reduced dependence on features handcrafting. Nafea et al. [18] discussed sensor-based solutions to human activity detection and the suitability of wearable sensors in the detection of motion dynamics. Hybrid networks, which combine CNN with RNN, have been shown to perform better as they can capture the space and time dimensions of the activities. Follow-up models have been centered on learning multi-resolution features and the optimal representation of data, and high levels of performance have been reached on standard HAR datasets. Xu et al. [19] studied classroom video analysis as an indicator of teacher behaviors and student interactions in classrooms and showed the utility of automated identification in the classroom. In this instance, deep learning techniques, specifically CNN, have been used to identify spatial characteristics of classroom interaction and do it correctly.

Yağcı et al. [20] explored the use of ML models to forecast achievement in education and have shown that test results can be reliable predictors of academic performance. All of them have been extensively applied to educational prediction problems due to their high classification performance. Zheng et al. [21] pinpointed the role of personal effectiveness in the outcomes of learners, which is of high predictability in learning environments. Scholarly studies have found that intrinsic and extrinsic motivators are potent in the participation and persistence of learning activities. As Nabil et al. [22] emphasized the use of deep learning to forecast achievement in higher education was discovered to be more efficient than the traditional models to predict at-risk kids. Having less precise but more explainable results than neural networks, they were commonly used in learning-based data mining. Early academic prediction has also been done using logistic regression and support vector classifiers, but in a linear and less flexible way. In managing the class imbalance, SMOTE and ADASYN resampling techniques have been proposed to strengthen classifiers. Katagiri et al. [23] emphasized how early motor skills serve as predictors of future academic and psychosocial success, thereby making them worthy of use as indicators of educational adjustment. It has been demonstrated that fine motor development is what facilitates early literacy and numeracy by optimizing the efficiency of tasks and participation in the classroom. Social interaction and regulation of emotions associated with Gross motor skills have been attributed to affect peer relationships and happiness.

Cagliero et al. [24] elaborated on the interpretability of early prediction models of student performance, and the interpretable classifiers may be useful in a specific intervention. Associative classifiers have rule-based explanations that enable the teacher to audit the model output and exhibit usable inferences. According to the meta-analysis done by Madigan et al. [25], student burnout is strongly negatively related to academic performance, with the least efficacy. Repeated fatigue research has proved that chronic fatigue impairs concentration and learning capacity. The model of predicting the success of the online courses in terms of the students is the development of the artificial intelligence-based model by Jiao et al. [26], where the focus was made on the combination of the data on the learning process and summative measures. To drive predictive models to optimality, evolutionary computation algorithms have been used to achieve competitive accuracy with respect to benchmark machine learning. Research has collected evidence that active involvement, knowledge gained and test results were better predictors of learning as compared to prerequisite knowledge. Martin et al. [27] emphasized the predictive role of grit to academic performance, especially through persistence of effort as an enabler of self-driven learning patterns. The empirical data indicate that individual performance and internal drive have a significant impact on determining the GPA of students because it influences the sense of confidence and endurance in learning. The relationship between grit and performance is mediated by cognitive strategy exploitation and self-regulation.

Heppt et al. [28] have pointed out the predictive significance of books-at-home measures on student academic performance, especially in the area of language comprehension. The trend of consistency in findings has been that print access boosts literacy development and moderates the effects of socioeconomic status on achievement. Home literacy environment, as reported by parents, typically compares better with self-reported ones of students and is more correlated with academic outcomes. The researchers also indicate that reading traditional print materials, including books by parents, leads to increased development of language as compared to using computer materials like e-books. Xiong et al. [29] also indicated that there are two-way relationships between parent involvement and higher education performance of adolescents, which indicate how the measure of

student achievement can actually mediate parenting. Studies have continually indicated that parental engagement does have a positive relationship with increased student engagement, student motivation, and higher education success. The relationship between family involvement and the learning outcomes has also proved to be mediated by academic engagement, especially behavioral engagement. It is further implied that these relations may not be gender similar in terms of their force and direction, as they may have variation in mediation effects between girls and boys. Comprehensively, parental involvement is a strong but multifaceted factor in the academic performance of adolescents. Al-Abyadh et al. [30] have pointed out the decisive role played by personal effectiveness and personal organization to academic success of students, in that they stressed the interdependent roles of both on learning success. It has also been indicated in research that students with personal organization skill can plan, monitor, and control their learning behaviors. Personal performance has long been considered a legitimate predictor of motivation, tenacity and performance in learning settings of various kinds. Comparisons involving other cultures indicate that these psych variables operate at the same level in aiding academic performance. Generally, the combination of personal organization strategies and the building of individual effectiveness is the most significant in the attainment of university students.

Online traces and patterns of internet use are discovered to screen student achievement very well with deep learning, graph-based, and recurrent models, providing more precise outcomes based on hybrid approaches. Other behavioral information and neural networks, such as spiking models, also boost the early identification of at-risk learners. The use of sensors with CNN and BiLSTM to combine spatial and temporal variations is a good approach to research [31], [35], but attention-based video classification of the classroom is a good predictor of engagement and participation. Machine learning techniques remain good predictors of academics, with deep learning techniques using resampling techniques being more stable. Personal effectiveness, motivation, self-regulation, and grit are psychological predispositions that have direct involvement in achieving results [32], where autonomous learning and interest often mediate these effects. They are powerful predictors of both academic and social adaptation, whereas parental involvement reciprocates and affects both achievement and is determined by gender-specific factors. Burnout is an adverse outcome factor, the greatest impact of which is reduced efficacy and disengagement. Process and summative information combined in AI models enhance forecasts in e-learning, whereas explainability and trust are provided by explainable models. Collectively, the evidence reiterates the fact that the set of technological, psychological, behavioral, and environmental factors is combined and supported by sophisticated AI and machine learning to enhance predictive accuracy and intervention in academic performance.

III. Proposed Methodology

The idea of academic performance prediction is based on integrated and systematic spatial and temporal learning based on an advanced model of deep learning. This starts with the data collection process, where the performance and behavior data of the students are collected in the form of normalized academic scores and standardized behavior scores. Once the data is gathered, there are activities involved in preprocessing, which include handling of missing data, normalization of academic marks, normalization of behavior features, and identification of temporal features. The processed data is then sent to the Hybrid CNN and LSTM model, in which the CNN detects the spatial patterns of data and the LSTM detects the time dependency of data. The identified features of CNN and LSTM are joined under a fusion layer and fed to the SoftMax function to categorize the students into one of the three categories: low, medium, or high risk. Besides this, the FOX optimization method is used to adjust the learning rates, which hastens the training process and enhances the convergence by fine-tuning gradients with the help of fractional-order derivatives. The performance metrics to ensure that the model can predict the results of students based on the past data. Fig. 1 demonstrates the hybrid deep learning architecture to predict the success of the students based on spatio-temporal and behavioral data.

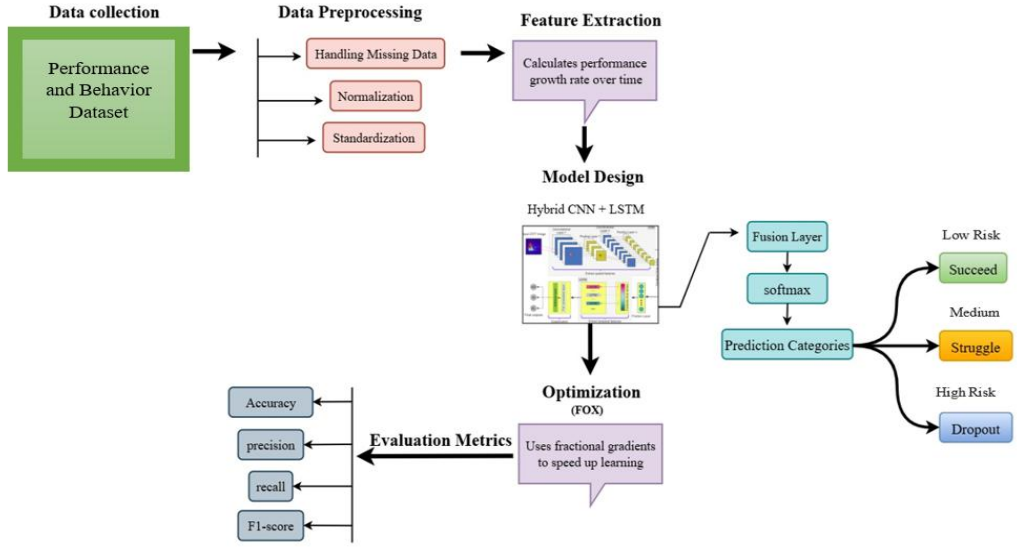


Fig. 1. Hybrid deep learning framework for predicting student success using spatio-temporal and behavioral data.

A. Data Collection

The academic history and learning behavior of the students in the Performance and Behavior Dataset obtained on Kaggle is well-rounded, as it consists of exam and assignment in-subject scores, attendance, classroom and study behavior, disciplinary behavioral indicators, and socio-demographic characteristics such as parental education, gender, and family background. The data set captures longitudinal academic tracks of students in several subjects, and this will be used to study the performance of the students longitudinally. To deploy the framework at the university-level, it additionally helps to combine the information system data of students, including course enrollment, credit completion, GPA/CGPA, and academic standing, with the logs of the learning management system, including the frequency of logins, content access, time-on-task and assignment submission behavior. Normalization, categorical encoding, and missing value are preprocessing steps that provide high-quality inputs that can be easily modeled spatio-temporally to predict early risks by academic performance and learning behavior.

B. Data Preprocessing

Preprocessing ensures that the behavioral and spatio-temporal data are consistent, cleaned, and prepared to be analyzed. The data treatment, academic scores normalization, behavioral indicators standardization, and time learning patterns extraction are missing in the operations involved. The conversions ensure that the data from various sources and scales may be compared and modeled in a meaningful way.

1) Handling missing data: The norm in datasets that cover the practical education is missing values due to attendance not mentioned, assignment not submitted, behavioral records missing or data entry errors. In the absence of such missing entries, analysis will be biased, and model performance will be worse as deep learning models need full and consistent input. To solve this, numeric attributes (e.g., test scores, percentage of class attendance, proportion of participation) are imputed with a mean imputation method, which replaces a missing value with the mean of all the available values of the attribute. This guarantees that the value added to it is representative of the frequency of the data, and it does not distort the general trend. The imputation equation can be written as in Eq. (1):

$$x_i' = \frac{\sum_{j=1}^{n} x_j}{n} \quad (1)$$

where, x_i' is the imputed value for the missing observation, x_j is the observed value of the feature, n is the number of valid observations of the feature. Mode imputation strategy is used in categorical attributes (e.g., type of study habit, disciplinary status, level of participation in classes). It is done by replacing the missing item with the most frequent category to keep the consistency in class distributions. These imputation techniques would ensure that the data set is not lost and the useful records are retained. This preserves the sample size and mitigates the systematic loss of data bias, and preserves all spatio-temporal and behavioral characteristics to conduct additional analysis. Finally, this approach produces a balanced platform of model training in which no feature is unfairly skewed due to the absence of data.

2) Normalization of academic scores: In education datasets, the marks of different subjects like mathematics, reading and writing usually have dissimilar marking systems that make these marks incomparable and hard to merge. An example of this is that one of the subjects will have a mark between 0 and 50, and the other one will have a mark between 0 and 100. Min-Max Normalization is used in order to have marks with an equal range. The process standardizes the data in a common range to facilitate easier analysis and comparisons among the individuals without the overbearing influence of one person because of the large range of scores. The scores are transformed through min-max normalization with the highest and the lowest numbers in the dataset. The normalized score (x_i') for a given raw score (x_i) is calculated in Eq. (2):

$$x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

where, x_i' is the normalized score, x_i is the original score of the student, $x_{\min}$ is the lowest score observed in the data of that subject, $x_{\max}$ is the highest score observed for that subject.

3) Standardization of behavioral features: Behavioral indicators (attendance rate, frequency of participation, frequency of submitting assignments, etc.) in learning data sets may vary in range and units of measurement. As an example, attendance could be measured as a percentage, participation frequency (e.g., low, medium, high), and rate of submission of assignments as a number, or as a percentage. These range variations and units of measurement may present difficulties in analyzing these features together, in that some variables may possibly overshadow or bias analysis just because of their range. Z-score Standardization is done in order to make these features of behavior comparable across subjects. The operation transforms all features in such a manner that the distribution has a mean of 0 and a standard deviation of 1. The standardization equation is available in Eq. (3):

$$x_i' = \frac{x_i - \mu}{\sigma} \quad (3)$$

where, x_i' is the original value of the feature of the i-th data point, x_i is the standardized value of the feature of the i-th data point, μ is the average of each student's feature, and σ is the parameter's variance of all students.

4) Applying SMOTE for class imbalance: After the features (X) and labels (y) have been preprocessed, it is time to use SMOTE to solve the problem of class imbalance in the training dataset. The SMOTE process involves creating synthetic examples for classes with underrepresented data to ensure balanced training data. This not only helps the model but also directs the model's attention towards the minority classes in terms of predictions.

C. Temporal Feature Extraction

Multi-scale temporal sequences are used to model student performance to learn both short-term dynamics as well as academic growth over time. Within-semester time series, including weekly variation in academic scores, attendance, and

engagement, indicate instant learning behavior, whereas semester-to-semester variations indicate the long-term signals of performance change, credit accumulation, and academic status shift. Temporal trends are measured in terms of the rate of growth in performance between successive time points, which denote an improvement, stagnation or reduction in academic performance. The student trajectories can be harmonized over long durations of education, and analyzed on a scale at this dual level of time-modelling. The growth rate in performance is expressed in Eq. (4):

$$\text{GrowthRate}_t = \frac{\text{Score}_t - \text{Score}_{t-1}}{\text{Score}_{t-1}} \times 100 \quad (4)$$

where, Score_t is the student's performance (e.g., exam score, assignment score, or overall grade) at time t (current period), Score_{t-1} is the student's performance at the previous time point (e.g., previous exam, week, or term), The result is multiplied by 100 to express the growth rate as a percentage.

D. Model Design Hybrid CNN and LSTM

It is based on an advanced deep learning model to classify student performance based on spatial and temporal data. CNN identifies spatial patterns with the help of such features as scores, attendance, and behavior, whereas LSTM identifies temporal changes over time. The CNN and LSTM output features are sent through a fully connected layer to acquire abstract patterns and features. Last, the output layer relocates the students to either low, medium, or high risk depending on the patterns learnt. Fig. 2 indicates the layout of the hybrid CNN and LSTM.

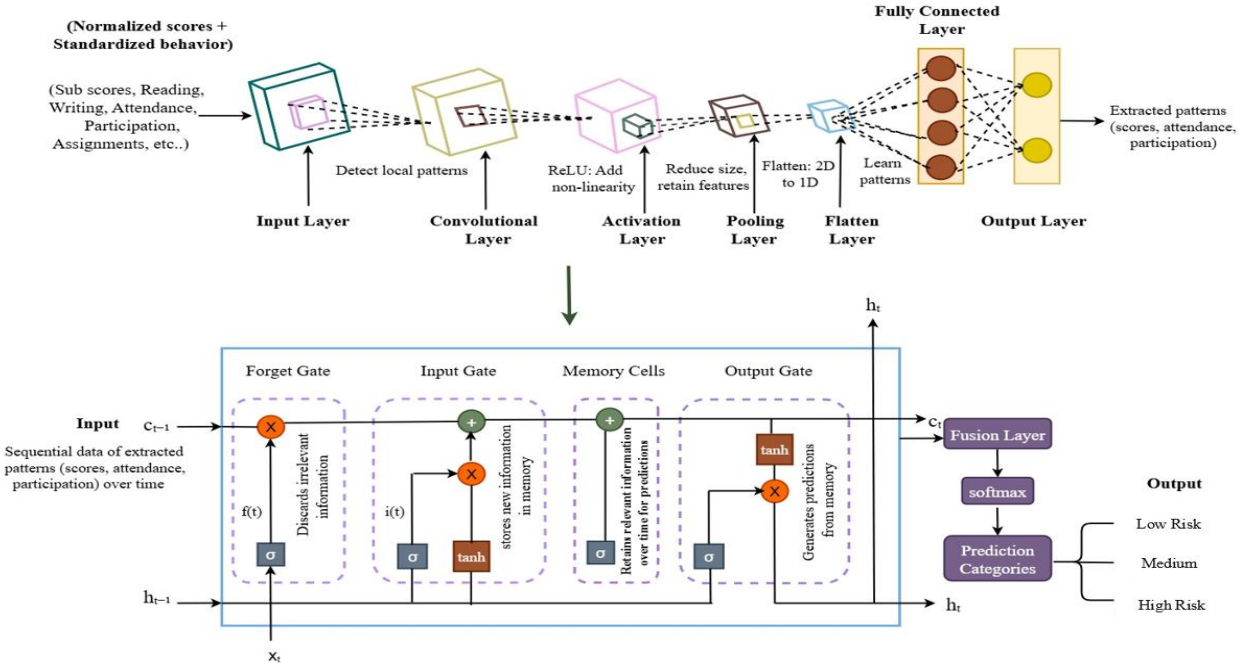


Fig. 2. Architecture of a hybrid CNN and LSTM.

1) Convolutional Neural Network: CNN identifies spatial patterns in the input features of performance grades of students, attendance, and participation. The CNN scans the input data to find significant patterns, including attendance-performance correlations or correlations between subjects, by applying the application of filters (kernels). These patterns are then captured in the form of feature maps and further processed by the activation and pooling layers so that only important features are stored, and complexity is minimized such that they may be utilized in the final classification and analysis.

a) Input layer: The first CNN layer operates on features such as academic scores, attendance, participation, and other behavioral traits that have been normalized. As a means to the end of predictive power, the model can also, alongside other variables like student demographics, LMS activity logs, and so forth, login frequency and time-on-task information, to mention a few, the model can also give more precise prediction outcomes by incorporating these features since they are often indicators of academic outcomes and thus provide a clearer view of the learning behavior. This additional input not only increases the feature space but also makes the model more robust by ensuring that it captures the patterns that perhaps only the grades alone would have otherwise missed. This enhancement makes the model more inclusive, leading to improved student performance prediction reliability.

b) Convolutional layer: The convolutional layer calculates the input data through filtering (or kernels). A filter is a small matrix that is passed over the input data (a sliding window) and multiplies and adds the products element-wise. This is used to identify significant spatial patterns of the input data, i.e. subject performance patterns or attendance patterns. This may be formulated mathematically as Eq. (5):

$$F(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} I(i+m-1, j+n-1) \cdot K(m,n) \quad (5)$$

where, $F(i,j)$ is the output feature map at position (i,i), $I(i,j)$ is the input data (score/attendance at position i,j), $K(m,n)$ is the filter used to identify patterns, M and N are the

filter's dimensions. After passing through all of the input data, the filter generates feature maps that highlight several significant features, such as attendance trends or scoring patterns.

Another way to improve model efficiency is by using a 1D-CNN structure instead of a conventional 2D-CNN. A 1D-CNN is highly applicable for time-oriented academic records like scores, attendance, or behavioral patterns because it processes one-dimensional data through its filters. Compared to 2D-CNN with its image-like grids, a 1D-CNN is concerned with the temporal aspect of the data, thus using fewer parameters and consequently incurring lesser computational cost. Not only does this lighter architecture manage to capture important trends in student performance, but it also enhances training speed and efficiency of the whole model. Such an alternative provides an additional CNN design more consistent with the temporal nature of academic data.

c) Activation Layer (ReLU): The convolution operation is followed by the activation layer (usually ReLU) that adds a nonlinear behavior to the network needed to understand intricate patterns. ReLU is just a mere conversion of the negative values into zero to enable the network to concentrate on positive values to accelerate training. This working can be mathematically expressed, as in Eq. (6):

$$f(x) = \max(0, x) \tag{6}$$

When the input value x is negative, ReLU will make it 0; when the value is positive, it does not change. This aids the CNN to learn and identifying more elaborate patterns since it gets to erase the adverse influences that may not be significant.

d) Pooling layer: The pooling layer will have the responsibility of down-sampling both the spatial height and width of the feature maps produced by the convolutional layer without eliminating the larger features. Max pooling is the most common technique, where the maximum value of a small region of the feature map is selected with the aim of minimizing the amount of data at the expense of the most valuable data. This operation can be expressed in Eq. (7):

$$P(i, j) = \max_{m,n}\{F(i + m, j + n)\} \tag{7}$$

where, $P(i, j)$ is the pooled value at the location (i, j), $F(i + m, j + n)$ is the feature map generated from the preceding layer. The max pooling operation effectively sums up the region it scans, without losing the essential detail, as it compresses the data.

e) Flatten layer: The flatten layer transforms the 2D feature maps acquired following the pooling to 1D vectors. This is due to the requirement of the next fully connected layers of having the input to be a 1D array. The flatten operation is merely the transfer of the matrix of the pooled features to the one-dimensional features in the form of a dense layer.

f) Fully connected layer: The one-dimensional vector is flattened and then sent to the fully connected layer, where every node is connected to every other node in the same layer. This learns complex, high-level features, which combine data from the CNN spatial feature extraction. In essence, it learns predictive patterns of student attainment, taking into consideration all the features the CNN has identified.

This layer is capable of learning the correlation of the features extracted, i.e. how one subject in the performance of one class may also be correlated with the engagement in some other activity. In order to give final predictions, the dense layer makes a weighted summation of inputs and feeds them through an activation function.

- Output Layer: The learned features of the fully connected layer are fed into the Output Layer, which converts them into output. It converts the characteristics into patterns extracted, such as scores, participation, and attendance. The prediction patterns are the predictive ones and are the higher aspects of student behaviors and performances.
- Ablation of CNN:
  - To check how well the model's performance is without spatial feature extraction, eliminate or comment out the layers related to CNN.
  - Evaluate the model after removing the CNN in terms of accuracy and other performance metrics.

2) Long Short-Term Memory: The LSTM networks are based on the RNN; they can take in sequential data and therefore can be applied in situations where time matters in order of data, such as the case of tracking students over time. LSTMs can recall the relevant information over a long period of time, forget the irrelevant information, and change the state of memory using a combination of gates to control the flow of data. These characteristics enable LSTMs to identify more advanced patterns of time-series data and also provide predictions based on past input.

This is because the LSTM takes sequential data in the shape of normalized scores (i.e., mathematics, reading, writing) and standardized behavioral scores (i.e., attendance, participation, assignment completion). All these data points are synchronized across several time steps, which enables the learning of the LSTM in order to determine how the performance and behavior of the students change. The time series nature of the data allows the model to measure the patterns and trends with time. This feed will be helpful when estimating the future performance and categorizing student performance.

a) Forget gate: The forget gate of the model then determines which part of the information at the previous step is to be forgotten. It does this by generating values between 0 and 1 of every value in the memory cell. These values determine the extent of the information of the past that will be required to make the current prediction, and which will not be needed and will be forgotten. The forget gate can be defined as in Eq. (8):

$$\mathrm{f_t} = \sigma(\mathrm{W_f} \cdot [\mathrm{h_{t-1}}, \mathrm{x_t}] + \mathrm{b_f}) \tag{8}$$

where, f_t is the forget gate output (which determines which portion of the previous memory to forget), The linear activated function is denoted by σ, which maps the output between 0 and 1, W_f is the forget gate's strength array, h_{t-1} is the hidden state

from the previous time step, x_t *is the* input data at the current time step, b_f is the bias term for the forget gate.

The forget gate applies the logistic activation function to produce outputs in the range of 0 to 1; 0 signifies forget everything, and 1 remember everything. This makes the model retain only the relevant past data to make predictions of the future and forget the non-relevant portions of the memory.

b) Input gate: Based on the forget gate, the input gate identifies the new information that has to be added to the memory cell. It creates a perfect memory cell update with the help of a tanh function and is based on a sigmoid equation to determine what values should be modified. This will enable the LSTM to add new useful information, without forgetting the most useful past environment. Input gate output, as shown in Eq. (9):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (9)$$

where, i_t is the input gate output (which determines which parts of the input will be updated in the memory), W_i represents the input gate's weight matrix, h_{t-1} symbolizes the previous time step's concealed state, x_t refers to the current time step's input data, b_i is the input gate's bias term, σ stands for the logistic activation function.

The sigmoid function provides the values of 0 to 1 to denote the percentage of the new data that is to be considered as relevant to be stored in the memory, and the tanh function provides the values that can be stored in the memory cell; it is normalized and added to the memory cell.

c) Memory cells: The memory cell includes valuable data in the long term after the input gate. It does it by combining the last memory state and the input candidate memory, multiplied by the input gate output. The current memory cell stores information that is useful in the next time step, and the LSTM can remember the useful information of the past and apply it in the next prediction. The memory cell adds the previously transferred memory of the previous time step, C _(t-1) and the current time step input C^t. This combination is done in such a way that only important data will be retained in the memory cell to be utilized again in the future, and the model has the ability to retain important patterns but forget information that is not very critical with time.

d) Output gate: The output gate of the LSTM determines which data can be traced out of the memory cell to form the final secret state (or output). It then transfers the result over a tanh activation process after computing an activation based on the current state of the memory cell. The result enhances the outcome of the gate activation and dictates a carryover of the concealed state to the next time period. When the input has gone through the output gate, the data is fed into the LSTM output, which is then used to combine the hybrid model output, and this is done through a fusion layer.

e) Attention mechanism: The LSTM, although it does capture the temporal dependencies, still suffers from the fact that not all time steps are equally important for predicting the student's performance. By introducing the attention mechanism after the LSTM layer, this problem is solved. The attention layer looks at all the hidden states that have been generated throughout the time sequence and gives each time step a weight according to its importance. In this manner, the model will be able to concentrate more on the pivotal moments in the student's behavior like a sudden drop in attendance or a big shift in academic scores at the same time, it will be able to reduce the impact of the less relevant periods. The attention score for each time step t is expressed in Eq. (10):

$$e_t = v^T \tanh(W_h h_t + b_h) \quad (10)$$

where, e_t is the raw attention score for time step *t*, h_t *is the* LSTM hidden state at time step *t,* W_h is the Weight matrix that transforms the LSTM hidden state for attention scoring, b_h is the bias term added during the transformation, v^T is the trainable weight vector used to convert the transformed hidden state into a single importance score, tanh is the activation function that introduces nonlinearity. The incorporation of attention eventually results in a model that is more stable, more accurate, and more interpretable. In this case, the educators can now know the periods that had the most influence on the prediction of low, medium, or high risk. The raw scores are then converted into attention weights through the application of a softmax function, which is presented in Eq. (11):

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T} \exp(e_k)} \quad (11)$$

where, α_t is the attention weight for time step *t,* e_t is the raw attention score at time step t, $\sum_{k=1}^{T} \exp(e_k)$ is the Normalization term that sums the exponentiated scores of all *T* time steps, *T* is the Total number of time steps in the sequence. The aggregated data is then passed to the softmax layer, which determines the probability distribution of all possible outcomes. In order to enhance the efficiency and prevent overfitting of the hybrid CNN–LSTM model, regularization techniques are integrated into the training process. The methods make the model less noise-sensitive and thus improve its capability to generalize to new student data.

- Dropout is implemented between the layers that randomly turn off a fraction of the neurons. This will help to avoid the network from depending excessively on specific nodes, and also, to make it learn more powerful and shared representations.
- Batch Normalization is utilized to control the distribution of layer activations, thereby yielding smoother and quicker training. It lessens internal covariate shift and aids the model to stay strong even when faced with variations or noise in the behavioral and academic inputs.

Dropout and batch normalization together significantly bolster the hybrid model's robustness, reduce the chances of overfitting, and deliver more accurate classifications of students into the categories: low, medium, and high risk. Finally, but not least, the SoftMax output is employed to generate the predicted categories, which determine the performance or risk of the student. All these assist the LSTM to predict dynamically and time-dependently with the aid of sequential pattern (LSTM) and spatial information (CNN).

The hybrid CNN-LSTM system can be applied to actual circumstances in universities by integrating student information

systems and learning management systems data without making any changes to the fundamental design. The indicators specific to the university, including credit accumulation, GPA/CGPA threshold, course failures, academic probation, and graduation eligibility, are modeled as structured inputs, and academic and credit patterns as spatial features represented with CNN, and semester-based progression and learning behavior as temporal sequences with the standard academic calendars. To aid in transparent decision-making, explainable AI methods are used to uncover the most significant academic and behavioral predictors to make accurate classifications of the probability of probation, course failure, and delayed graduation risks as low, moderate, and high-risk risks and inform academic advising.

- Ablation of LSTM:
  - To discover how the model reacts without the LSTM layers involved in temporal sequence modeling, either remove them or comment them out.
  - This will indicate how much LSTM contributes to the capturing of time dependencies in the data.

TABLE I. STUDENT PERFORMANCE PREDICTION CATEGORIES

Category	Description
Low Risk	According to the model, the pupil will probably do well and be successful.
Medium	According to the model, the pupil is likely to struggle and encounter obstacles.
High Risk	The model predicts a high risk of the student dropping out.

Table I of the predicted categories shows the input categories of the model, which involve using CNN and LSTM. The models include: low risk, in which the model student performs well and is an academic success; medium, which refers to the fact that the student will struggle and he/she may require more help to be a better student; and high risk, which refers to students who are at a high risk of disengagement and can drop out. The predictions will be made based on trends within the space observations (e.g., scores, attendance) and time trends (e.g., behavior over time), which makes the model categorize the students into such groups to provide them with an individualized intervention and guidance. Algorithm 1 shows hybrid deep learning for academic achievement prediction.

Algorithm 1

Input:

1. Spatio-Temporal Data:
   - Normalized Academic Scores: Scores from various subjects (e.g., mathematics, reading, writing).
   - Standardized Behavioral Features: Attendance, participation, assignment completion rates, etc.
   - Time-Series Data: Data over multiple time steps (e.g., weeks, terms).

Output:

- Predicted Categories:
  - Low Risk: The student is likely to perform well.
  - Medium: The student is likely to face academic challenges.
  - High Risk: The student is at risk of dropping out.

Step 1: Data Preprocessing

1. Normalization:
   - Apply Min-Max normalization to academic scores to bring them to a comparable range (e.g., [0, 100]).
   - Apply Z-score standardization to behavioral features like attendance, participation, and homework completion.
2. Handle Missing Data:
   - Use techniques such as mean imputation or data interpolation to handle missing values in the dataset.

Step 2: Convolutional Neural Network for Spatial Feature Extraction

1. Input Layer:
   - Input normalized academic scores and standardized behavioral features as multi-dimensional data.
2. Convolutional Layer:
   - Find geographic patterns in input data, use filters (kernels) (e.g., relationships among scores and attendance).
3. Activation Layer (ReLU):
   - Apply the ReLU activation function to introduce nonlinear behavior.
4. Pooling Layer:
   - Minimize spatial dimensions without sacrificing important features, use max pooling.
5. Flatten Layer:
   - In order to use the feature maps from the pool as inputs for the fully linked layer, flatten them into a 1D vector.
6. Fully Connected Layer:
   - Learn complex patterns from the extracted features and pass the learned features to the next stage.

Step 3: LSTM for Temporal Modeling

1. Input Layer:
   - Sequential data of extracted features (scores, attendance, participation) over time is passed into the LSTM model.
2. Forget Gate:
   - Based on its applicability to future projections, the forget gate determines whether data from previous steps should be ignored.
3. Input Gate:
   - To guarantee that only pertinent data is retained, the input gate determines which fresh data must be put to the memory cell.
4. Memory Cell:
   - Memory cells store the relevant information from the current and previous time steps. On the basis of the input and forget gates, they update.
5. Output Gate:
   - The output gate passes the current input through an activation mechanism and generates the outcome depending on the modified memory cell.

Step 4: Feature Fusion Layer

1. Fusion Layer:

- Combine the outputs of the CNN (spatial features) and LSTM (temporal features) to form a unified representation.

Step 5: Final Prediction Layer

1. Fully Connected (Dense) Layer:
   - This layer learns from the fused features and classifies the student risk into one of the categories: low, medium, or high risk.
2. Softmax Layer:
   - Utilizing the softmax activation function, a distribution of chances across the categories is generated.

Output:

- The ultimate classification is determined by using the softmax output (predicted categories: low, medium or high).

IV. FOX OPTIMIZATION

The FOX optimizer (Fractional Order optimizer with Exponential Convergence) is a superior optimization method that optimizes the rate of convergence and accuracy of deep learning models. Unlike the traditional optimizers, FOX uses fractional-order derivatives to learn how to update the learning rate. The novel approach to gradient calculation adjustment with the use of fractional exponents has been utilized in the novel method to enhance the stability and efficiency of deep learning models during training. FOX accelerates convergence by the fact that it addresses the problems of the local minima and sluggishness of training, which are intrinsic to the classical optimization methods. FOX is an optimal and more effective way of updating parameters since it employs fractional orders in updating the gradients. The most essential thing about the FOX optimizer is its two-step process, which is gradient update and fractional-order scaling. The first step is the formula of gradient update that utilizes the derivatives of the model parameters in a fractional order to update them. This is represented in Eq. (12):

$$\theta_t = \theta_{t-1} - \alpha_t \cdot g_t^{\gamma} \tag{12}$$

where, θ_t is the parameter at time step t, α_t is the learning rate at time t, g_t is the gradient at time t, and γ is the fractional order used to scale the learning rate. This update would speed convergence, scaling the gradient with a fractional exponent, which gives more accurate parameter updates. Step 2 is fractional-order scaling, in which the gradient scale is scaled by its L 2 norm. This action keeps the amount of the update proportionate to the importance of every parameter, enhancing the efficiency of the optimizer with complex, multi-layer models like CNN + LSTM. The fractional-order scaling takes the form of Eq. (13):

$$\theta_t = \theta_{t-1} - \alpha_t \cdot (\|g_t\|^{\gamma}) \tag{13}$$

where, $\|g_t\|$ represents the L2 norm of the gradient at time t, making sure the optimizer modifies the learning rate in accordance with the gradient's strength. The fractional order γ maximizes the value of each gradient to its optimal value, thus allowing the optimizer to respond better to the dynamic gradient with time.

A. Comparative Analysis: FOX vs. Adam Optimizer

The comparison with the Adam optimizer, which is one of the most popular deep learning optimizers, is sought for a better understanding of the advantages of FOX. Adam makes use of momentum and adaptive learning rates derived from first and second-order moment estimates to update parameters. This results in fast convergence and allows Adam to deal with noisy gradients effectively.

FOX is significantly different from Adam in two main aspects. Firstly, FOX employs fractional-order gradient scaling, which gives more flexible and accurate control over parameter updates, in contrast to Adam, which uses fixed exponential averages. Secondly, FOX updates its step size according to the L2 norm of the gradient, which enables it to breathe life into flat regions and slow-converging plateaus that often hinder Adam. Consequently, the path to convergence for FOX could be shorter and its reliability higher, especially with the intricate hybrid structures such as CNN + LSTM.

Nonetheless, the Adam algorithm is still a robust reference point mainly due to its ease of use, less intensive computations, and wide acceptability through its performance in various tasks. The comparison of both optimizers simultaneously uncovers the learning dynamics, convergence behavior, and robustness of the model under consideration more clearly.

B. Ablation of FOX

- Detect the absence of FOX and install the simpler optimizer (Adam), then check if the removal of FOX resulted in the slowing down of convergence and the reduction of model stability.
- The above-mentioned assessment will facilitate determining whether or not the tailor-made FOX optimizer is a necessity or if a standard optimizer like Adam can yield the same result.

V. RESULTS

The outcomes of the proposed hybrid CNN and LSTM structure are provided in this section. The architecture was written in Python with popular deep learning libraries based on TensorFlow and Keras being used to perform the computations. Training and testing data entail scholastic achievement information of the students, as well as normalized scores on subjects like mathematics, reading, and writing, and standardized behavioral characteristics like participation, attendance, and submission of assignments. The data is spread across the time steps, depicting the performance of the students on several terms or weeks of the year, to enable the model to learn the spatial and temporal pattern of knowledge. Subsequent parts include general performance metrics, including the rate of correctness and an exactness measure, as well as the rate of sensitivity and the balance score and graphical depictions such as a confusion matrix, ROC curve and precision-recall curve. The results demonstrate how the framework can be used to make predictions of the student performance categories, which explains its effectiveness in a classroom setting.

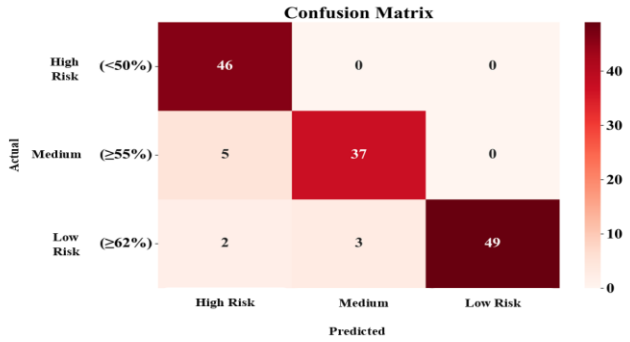


Fig. 3. Confusion matrix.

The accuracy of the model in predicting the performance of pupils is measured using the confusion matrix in Fig. 3. It highlights the differences between risk categories (low, medium, and high risk) and projected categories based on the model's projections. True positives are observed on the diagonal boundary, such that the model made the right prediction as to the respective sets of student performance. The off-diagonal values are poor predictions where the model predicted one category in another one. In general, there is excellent predictive confidence in the low and medium risk groups that is represented in the matrix.

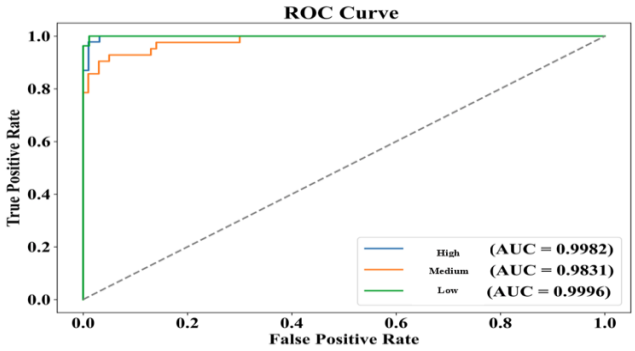


Fig. 4. ROC curve.

In the Fig. 4 of the ROC curve, the capacity of the framework in predicting the result of the pupil in three groups, that is, high risk, medium, and low risk, is evident. The trade-off in the curve is the true positive rate (sensitivity) and false positive rate (1-specificity) for each group. Accuracy is very high with the area under the curve of the model being 0.9982 in the case of high risk, 0.9831 in the case of medium and 0.9996 in the case of low risk, demonstrating that the model is a fantastic one in the differentiation of the different classes of students. The green curve that provides the best AUC is the Succeed curve, and it has high predictive power on the set of students who will succeed, although the other two classes also have good predictive power.

The model is described by a plot, which shows its training and validation accuracy at various epochs in Fig. 5. The accuracy of training is blue and the accuracy of validation is orange. The training and validation accuracies increase with the number of epochs, and the training accuracy is always equal to 1.0. The validation accuracy also increases significantly but unequally, which may be a manifestation of overfitting or minor variation in the performance of the training and the validation sets. Overall, the model has justifiable positive results in learning the plot and good generalization between epochs.

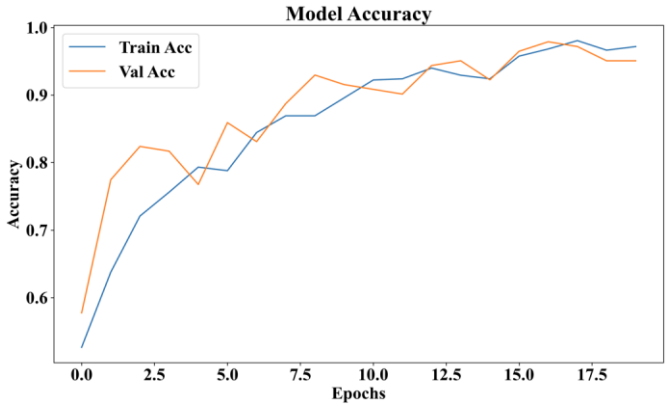


Fig. 5. Model accuracy.

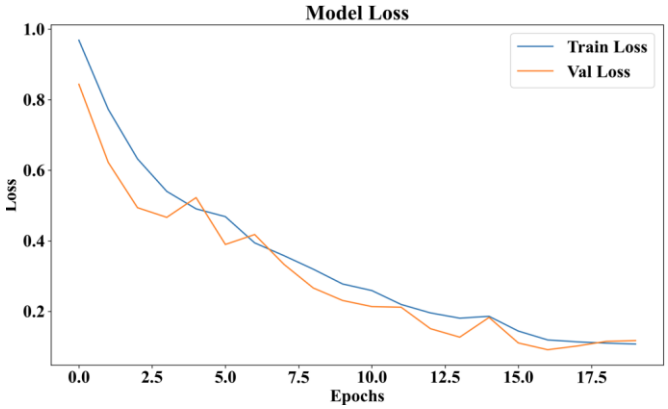


Fig. 6. Model loss.

In Fig. 6, the plot of the training and validation loss of the model at various epochs is presented. Blue and orange lines represent the training and validation loss, respectively. With the increase of the epochs, the losses become less and less, which means that the model learns and minimizes the errors. However, loss on validation hits some point, and this signifies that the model is generalizing very well. This indicates that the score of the verification set remains constant in the course of training, and this means that training is efficient and there is minimal overfitting, although the algorithm itself is still learning.

A. Performance Analysis

The results of the performance analysis section indicate how efficient the proposed Hybrid CNN and LSTM model is in terms of predicting the academic success of students. Fig. 7, Fig. 8, and Fig. 9 show important performance indicators that are the False Negative vs. False Positive Rates and the Precision-Recall curves and the overall performance of the model that has high precision, recall, and accuracy. The outcome demonstrates that the model has a high degree of accuracy in predicting different categories of students, and it is superior to regular models such as LSTM and SVM. The better results of an enhanced deep learning model using various evaluation measures are also confirmed in Table II, confirming its use to predict with high precision in academic studies.

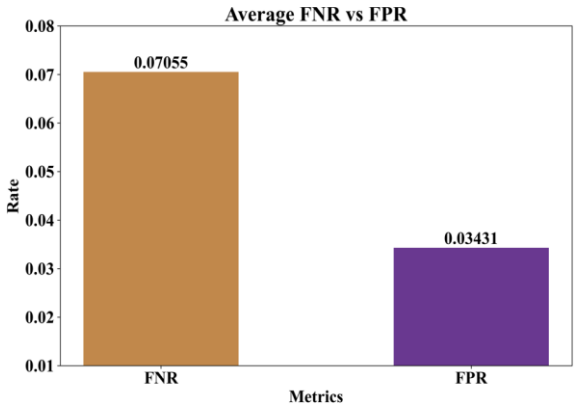


Fig. 7. Average FNR vs. FPR.

In Fig. 7, the average predictions of the model of FNR versus FPR are compared using the bar chart. FNR has a brown bar with a measure of 0.070555, and FPR has a purple bar with a measure of 0.03431. As indicated in the graph, the model creates a greater FNR than the FPR, which implies that the model incorrectly classifies students as not being at risk more than the rate of the inaccurate classification of the students as being at risk.

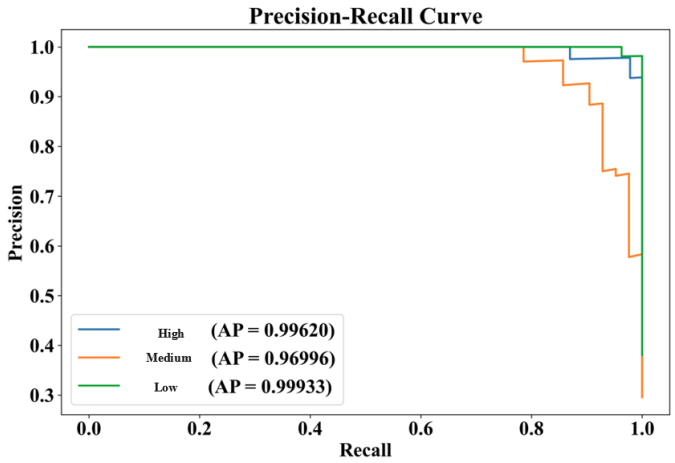


Fig. 8. Precision-recall curve.

The model performance for the different classes: high risk, medium, and low risk is shown in precision-Recall curves in Fig. 8. Each curve will indicate the precision-recall trade-off of each class as well as the area under the curve, which will indicate the performance of the model. The highest AP belongs to succeed 0.999333, which gives a high recall and precision, followed by 0.99620 that gives high recall but low precision, and 0.96996 that gives low recall but high precision. These values reflect the fact that the model predicts successful pupils with phenomenal precision, and is fairly good at predicting dropouts, but somewhat reduced in the medium prediction.

The performance values of the hybrid model are excellent results of the performance on several performance metrics in Fig. 9. The prediction of the model was 97.18 per cent, which is a good overall result. Precision and recall are also quite high at 97.02 and 97.35, respectively, which shows that the model did not misclassify the positive instances and had the lowest number of FPR and FNR. F1-score is also 97.14, which indicates a recall and precision balance. These results indicate the predictive ability of the model for student success in the three risk levels.

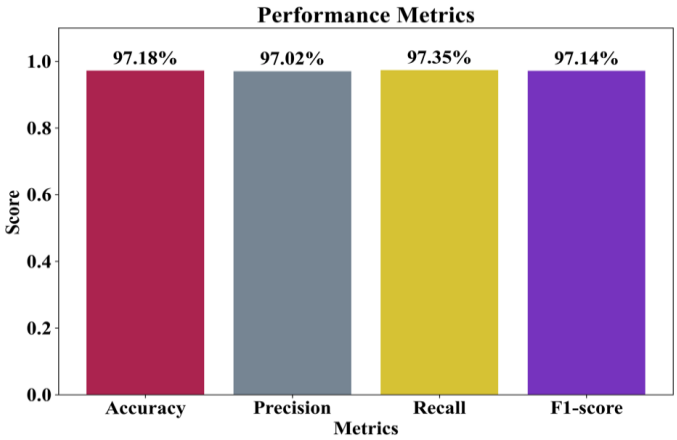


Fig. 9. Performance metrics.

B. Model Explainability with SHAP

In order to improve the interpretability of the model, utilized SHAP to identify the important factors that influence the predictions. SHAP values reveal the contribution of each feature (e.g., academic scores, attendance, participation) to the decision of the model for a particular prediction. By attributing the prediction to the contributions from each feature, SHAP provides the reasoning behind the prediction of a student risk level as high, medium, or low.

Through the application of SHAP values, it is possible to notice that attendance and participation are the most determining factors in the prediction of the "medium" category. A student might be predicted as a "Struggler", and the prediction may be influenced by behavioral factors like low participation, even though the student has high academic scores. SHAP makes it clear and transparent. The force plot below shows the contributions of features for the student, who has been predicted as a Struggler, individually. The contributions that are positive push the prediction in the direction of "medium", while the negative contributions retract it from the prediction. Such insights are especially helpful for the education sector as they not only provide practitioners with an understanding of the predictions of the model but also enable them to customize interventions according to the particular reasons that determine a student's performance.

TABLE II. PERFORMANCE EVALUATION TABLE

Method	Accuracy	Precision	Recall	F1-Score
LSTM [33]	87	87	88	88
SVM [34]	82	93	84	89
GRU	92.34	91.00	90.30	90.65
Bi-LSTM	93.50	92.10	91.50	91.80
Transformer	94.75	94.00	93.90	93.95
TCN	95.20	94.60	94.20	94.40
Hybrid CNN and LSTM	97.18	97.02	97.35	97.14

Table II provides a detailed view of how different models, including LSTM, SVM, GRU, Bi-LSTM, Transformer, TCN, and the proposed Hybrid model, performed compared to one another. The Hybrid CNN and LSTM model got the best performance ever recorded with the highest accuracy (97.18 per cent) and also surpassed all remaining models according to precision, recall, and F1-score. Additionally, the two models, i.e., Transformer and TCN, in this case, have exhibited great capability, especially when it comes to long-term dependency issues, as their maximum accuracy reached 94.75 per cent and 95.20 per cent, respectively. On the other hand, while GRU and Bi-LSTM have yielded close results, the hybrid model is still the most powerful in predicting students' performance.

VI. Discussion

The outcomes from the comparison with the models such as GRU, Bi-LSTM, and those based on Transformer and TCN kept confirming that the hybrid CNN-LSTM-FOX model is superior to traditional models like LSTM and SVM in terms of accuracy, scoring a remarkable 97.18 per cent. The hybrid model has been very effective in combining the strengths of both CNN and LSTM, enabling the former to extract spatial patterns while the latter recognizes temporal ones, eventually providing a very holistic understanding of student performance. In the case of models like Transformer and TCNs, they offer certain advantages over LSTM in capturing long-term dependencies and processing sequential data, but still, the hybrid CNN-LSTM-FOX model, when judged strictly on performance, is the best. Conversely, GRU is less accurate than LSTM and the hybrid model, but is still a good choice in resource-constrained environments due to its simpler architecture and lower computational time. In general, the findings indicate that the hybrid model with FOX is always superior to comparative architectures as it successfully learns both spatial and temporal learning patterns. Besides higher education, the structure can be easily incorporated into the institutions of higher learning where the academic progression is stipulated by credit accruals, GPA/CGPA and prerequisites. It is appropriate in academic risk identification in undergraduate and postgraduate programs due to its capability to model heterogeneous and semester-wise academic trajectories. The academic risk is determined based on the higher-education outcomes, like course failure, GPA/CGPA, academic probation, and delayed graduation, which allows grouping low-risk, moderate-risk, and high-risk groups.

A. Focused Error Analysis

While the hybrid CNN-LSTM model exhibits a robust performance, it continues to be necessary to take care of the situations in which the model performs poorly. A concentrated error analysis indicates that the model is particularly in trouble when it comes to students who are academically very good but have low attendance, resulting in misclassification between "low" and "medium". The model often makes these errors with the pupils who occasionally attend classes, and hence, the model might not be able to pick up on this engagement of the student's very well. Furthermore, sometimes the students with irregular behavior patterns are wrongly classified, which implies that the model is not able to grasp the full complexities of behavioral data over time. The above-mentioned weaknesses signal the areas for possible model refinement. Later on, one of the approaches could be developing the data preprocessing, for instance, through better management of attendance and participation data or combining it through more sophisticated features that track student engagement over time. Without doubt, the model that is more sensitive to the mentioned factors could lead to a more accurate and reliable prediction system.

B. Practical Implications

The proposed framework is a kind of early warning mechanism allowing academic advisors and instructors to recognize students who are at risk at the early stage of their academic life. Predictions on the model are converted to risk scores that can be acted upon and early warning groups to enable advisors to prioritize students who need urgent attention. Advisors can use explainable outputs to emphasize important academic and behavioral risk factors (such as a drop in GPA, inadequate credit accumulation, failure in courses, low attendance, and decreased LMS engagement) to develop specific interventions, such as tutoring, mentoring, probation support, or individualized learning plans. This is a timely and data-driven decision support which improves retention of students, efficient use of institutional resources and also an ongoing monitoring of the student progression within the university academic systems.

To guarantee scalability and generalization, the proposed framework will be tested on a variety of colleges and academic subjects to take into consideration the differences in the curriculum structure, grading policy, and credit requirements. The cross-program evaluation helps to evaluate the model in terms of its robustness and transferability to prove that it can be applied not only in one institution or discipline.

C. Ethical and Privacy Considerations

To enable the adoption of responsible deployment in the context of universities, the framework will utilize ethical and privacy-sensitive tools, such as role-based access control, data minimization, and bias monitoring. There is restricted access to student data based on institutional roles (e.g., advisors, instructors), and prediction is based on only necessary academic and behavioral characteristics. Also, the results of predictions are reviewed periodically within demographic and academic subgroups to address and reduce possible bias to secure, equitable, and reliable decision support of academic advising.

VII. Conclusion and Future Work

The study introduces a hybrid CNN-LSTM that is used to determine the student risk level results of being low, medium, or high. CNN is used to process student data, such as academic grades and personal attributes, to identify spatial patterns, whereas LSTM is used to track progress over semesters. The model was found to have an accuracy of 97.18 per cent and was better than other traditional models like LSTM and SVM; the interpretability was also guaranteed through SHAP to determine the key predictive variables. It could be assessed as an early-warning system that helps teachers to notice at-risk students and intervene timely. The framework can apply to a wide range of educational contexts with the introduction of university-specific characteristics such as credit systems, GPA/CGPA, prerequisites, and semester-wise progression, although tested on a single-institution dataset, which limits the scope of

generalization of the results. Further studies could involve larger and multi-institutional groups of data and other socio-economic and learning-environment variables to further enhance the predictive strength. The framework consists of ethical protections like role-based access, data minimization, and bias monitoring that can be deployed in a higher education system in a privacy-aware and responsible manner.

The proposed framework will be tested in the future on a variety of colleges and academic fields to determine its strength and cross-program extensibility in the context of different higher education institutions.

DECLARATIONS

Data Availability: https://www.kaggle.com/datasets/mahmoudelhemaly/students-grading-dataset

REFERENCES

[1] Y. A. Alsariera, Y. Baashar, G. Alkawsi, A. Mustafa, A. A. Alkahtani, and N. A. Ali, "Assessment and evaluation of different machine learning algorithms for predicting student performance," Computational Intelligence and Neuroscience, vol. 2022, no. 1, Art. no. 4151487, 2022, doi: 10.1155/2022/4151487.

[2] Y. Wan, R. Li, W. Li, and H. Du, "Impact pathways of AI-supported instruction on learning behaviors, competence development, and academic achievement in engineering education," Sustainability, vol. 17, no. 17, p. 8059, Jan. 2025, doi: 10.3390/su17178059.

[3] Q. Meng and Q. Zhang, "The influence of academic self-efficacy on university students' academic performance: The mediating effect of academic engagement," Sustainability, vol. 15, no. 7, p. 5767, 2023, doi: 10.3390/su15075767.

[4] P. Gui, G. M. Alam, and A. B. Hassan, "Whether socioeconomic status matters in accessing residential college: Role of RC in addressing academic achievement gaps to ensure sustainable education," Sustainability, vol. 16, no. 1, p. 393, Jan. 2024, doi: 10.3390/su16010393.

[5] N. Yavuzalp and E. Bahcivan, "A structural equation modeling analysis of relationships among university students' readiness for e-learning, self-regulation skills, satisfaction, and academic achievement," Research and Practice in Technology Enhanced Learning, vol. 16, no. 1, p. 15, 2021, doi: 10.1186/s41039-021-00162-y.

[6] X. Tang, M. T. Wang, F. Parada, and K. Salmela-Aro, "Putting the goal back into grit: Academic goal commitment, grit, and academic achievement," Journal of Youth and Adolescence, vol. 50, no. 3, pp. 470–484, Mar. 2021, doi: 10.1007/s10964-020-01360-1.

[7] Y. Baashar, G. Alkawsi, A. Mustafa, A. A. Alkahtani, Y. A. Alsariera, A. Q. Ali, et al., "Toward predicting student's academic performance using artificial neural networks (ANNs)," Applied Sciences, vol. 12, no. 3, p. 1289, Jan. 2022, doi: 10.3390/app12031289.

[8] Z. Wu, T. F. Spreckelsen, and G. L. Cohen, "A meta-analysis of the effect of values affirmation on academic achievement," Journal of Social Issues, vol. 77, no. 3, pp. 702–750, Sep. 2021, doi: 10.1111/josi.12460.

[9] E. M. Onyema, K. K. Almuzaini, F. U. Onu, D. Verma, U. S. Gregory, M. Puttaramaiah, and R. K. Afriyie, "Prospects and challenges of using machine learning for academic forecasting," Computational Intelligence and Neuroscience, vol. 2022, no. 1, Art. no. 5624475, 2022, doi: 10.1155/2022/5624475.

[10] Z. Xu, Y. Zhao, B. Zhang, J. Liew, and A. Kogut, "A meta-analysis of the efficacy of self-regulated learning interventions on academic achievement in online and blended environments in K-12 and higher education," Behaviour & Information Technology, vol. 42, no. 16, pp. 2911–2931, 2023, doi: 10.1080/0144929X.2022.2135680.

[11] S. Hussain and M. Q. Khan, "Student-performulator: Predicting students' academic performance at secondary and intermediate level using machine learning," Annals of Data Science, vol. 10, no. 3, pp. 637–655, 2023, doi: 10.1007/s40745-021-00376-5.

[12] E. Goh and H. J. Kim, "Emotional intelligence as a predictor of academic performance in hospitality higher education," Journal of Hospitality & Tourism Education, vol. 33, no. 2, pp. 140–146, 2021, doi: 10.1080/10963758.2020.1843197.

[13] S. Poudyal, M. J. Mohammadi-Aragh, and J. E. Ball, "Prediction of student academic performance using a hybrid 2D CNN model," Electronics, vol. 11, no. 7, p. 1005, Mar. 2022, doi: 10.3390/electronics11071005.

[14] J. S. Ryu, H. R. Chung, B. M. Meador, Y. Seo, and K. O. Kim, "The associations between physical fitness, complex vs simple movement, and academic achievement in a cohort of fourth graders," International Journal of Environmental Research and Public Health, vol. 18, no. 5, p. 2293, Mar. 2021, doi: 10.3390/ijerph18052293.

[15] X. Wu and M. Guo, "Higher education expansion and the changing college wage premium in Hong Kong, 1976–2016," Chinese Journal of Sociology, vol. 8, no. 3, pp. 28–47, 2022, doi: 10.1177/2057150X221103221.

[16] Z. Shou, M. Xie, J. Mo, and H. Zhang, "Predicting student performance in online learning: A multidimensional time-series data analysis approach," Applied Sciences, vol. 14, no. 6, p. 2522, Mar. 2024, doi: 10.3390/app14062522.

[17] C. Liu, H. Wang, Y. Du, and Z. Yuan, "A predictive model for student achievement using spiking neural networks based on educational data," Applied Sciences, vol. 12, no. 8, p. 3841, Apr. 2022, doi: 10.3390/app12083841.

[18] O. Nafea, W. Abdul, G. Muhammad, and M. Alsulaiman, "Sensor-based human activity recognition with spatio-temporal deep learning," Sensors, vol. 21, no. 6, p. 2141, Mar. 2021, doi: 10.3390/s21062141.

[19] T. Xu, W. Deng, S. Zhang, Y. Wei, and Q. Liu, "Research on recognition and analysis of teacher–student behavior based on a blended synchronous classroom," Applied Sciences, vol. 13, no. 6, p. 3432, Mar. 2023, doi: 10.3390/app13063432.

[20] M. Yağcı, "Educational data mining: Prediction of students' academic performance using machine learning algorithms," Smart Learning Environments, vol. 9, no. 1, p. 11, 2022, doi: 10.1186/s40561-022-00192-z.

[21] B. Zheng, C. Chang, C. H. Lin, and Y. Zhang, "Self-efficacy, academic motivation, and self-regulation: How do they predict academic achievement for medical students?" Medical Science Educator, vol. 31, no. 1, pp. 125–130, 2021, doi: 10.1007/s40670-020-01123-2.

[22] A. Nabil, M. Seyam, and A. Abou-Elfetouh, "Prediction of students' academic performance based on courses' grades using deep neural networks," IEEE Access, vol. 9, pp. 140731–140746, 2021, doi: 10.1109/ACCESS.2021.3119334.

[23] M. Katagiri et al., "Fine and gross motor skills predict later psychosocial maladaptation and academic achievement," Brain and Development, vol. 43, no. 5, pp. 605–615, May 2021, doi: 10.1016/j.braindev.2020.12.009.

[24] L. Cagliero, L. Canale, L. Farinetti, E. Baralis, and E. Venuto, "Predicting student academic performance by means of associative classification," Applied Sciences, vol. 11, no. 4, p. 1420, Feb. 2021, doi: 10.3390/app11041420.

[25] D. J. Madigan and T. Curran, "Does burnout affect academic achievement? A meta-analysis of over 100,000 students," Educational Psychology Review, vol. 33, no. 2, pp. 387–405, Jun. 2021, doi: 10.1007/s10648-020-09533-1.

[26] P. Jiao, F. Ouyang, Q. Zhang, and A. H. Alavi, "Artificial intelligence-enabled prediction model of student academic performance in online engineering education," Artificial Intelligence Review, vol. 55, no. 8, pp. 6321–6344, 2022, doi: 10.1007/s10462-022-10146-9.

[27] H. Martin, R. Craigwell, and K. Ramjarrie, "Grit, motivational belief, self-regulated learning (SRL), and academic achievement of civil engineering students," European Journal of Engineering Education, vol. 47, no. 4, pp. 535–557, 2022, doi: 10.1080/03043797.2021.1935532.

[28] B. Heppt, M. Olczyk, and A. Volodina, "Number of books at home as an indicator of socioeconomic status: Examining its extensions and their incremental validity for academic achievement," Social Psychology of Education, vol. 25, no. 4, pp. 903–928, 2022, doi: 10.1007/s11218-022-09702-6.

[29] Y. Xiong, X. Qin, Q. Wang, and P. Ren, "Parental involvement in adolescents' learning and academic achievement: Cross-lagged effect and mediation of academic engagement," Journal of Youth and Adolescence, vol. 50, no. 9, pp. 1811–1823, Sep. 2021, doi: 10.1007/s10964-021-01464-3.

[30] M. H. A. Al-Abyadh and H. A. H. Abdel Azeem, "Academic achievement: Influences of university students' self-management and perceived self-efficacy," Journal of Intelligence, vol. 10, no. 3, p. 55, Jul. 2022, doi: 10.3390/jintelligence10030055.

[31] J. K. Singh and A. Kaur, "Is teaching and learning in Chinese higher education classrooms internationalized? Perspectives from international students in China," Higher Education Research & Development, vol. 42, pp. 1283–1297, 2023, doi: 10.1080/07294360.2022.2098325.

[32] V. Macakova and C. Wood, "The relationship between academic achievement, self-efficacy, implicit theories and basic psychological needs satisfaction among university students," Studies in Higher Education, vol. 47, no. 2, pp. 259–269, 2022, doi: 10.1080/03075079.2020.1739011.

[33] Z. S. Hafdi and S. El Kafhali, "A comparative evaluation of machine learning methods for predicting student outcomes in coding courses," AppliedMath, vol. 5, no. 2, p. 75, 2025, doi: 10.3390/appliedmath5020075.

[34] Y. Wang et al., "N-STGAT: Spatio-temporal graph neural network-based network intrusion detection for near-earth remote sensing," Remote Sensing, vol. 15, no. 14, p. 3611, Jul. 2023, doi: 10.3390/rs15143611.

[35] O. N. Akande, M. O. Lawrence, and P. Ogedebe, "Application of bidirectional LSTM deep learning technique for sentiment analysis of COVID-19 tweets: post-COVID vaccination era," Journal of Electrical Systems and Information Technology, vol. 10, no. 1, p. 50, 2023, doi: 10.1186/s43067-023-00082-9.

[36] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv preprint, arXiv:1412.3555, Dec. 2014. Available: https://arxiv.org/abs/1412.3555

Farm and Learn: An Offline Mobile Learning System Integrating AR, AI, and Game-Based Learning for Agricultural Education Among Children

Pubuditha De Silva[1], Chamodya Prabodhani[2], Daminda Herath[3]
Faculty of Information Technology, Horizon Campus, Malabe, Sri Lanka[1, 2]
Department of Computer Science, Sir John Kotelawala Defense University, Ratmalana, Sri Lanka[3]

***Abstract*—This study presents Farm and Learn, an offline-first mobile learning system that integrates Augmented Reality (AR), Artificial Intelligence (AI), and Game-Based Learning (GBL) to enhance agricultural education among children in low-connectivity environments. Existing agricultural learning applications often provide isolated functionalities such as visualization or plant recognition, with limited pedagogical integration and insufficient support for rural deployment. To address these limitations, the proposed system combines immersive AR-based exploration, interactive gamified learning activities, and AI-assisted paddy plant growth-stage identification within a unified child-centered educational framework. The architecture adopts a modular offline-first design that enables core learning functionalities to operate without continuous internet access while allowing optional synchronization when connectivity is available. The AI component employs a lightweight YOLOv11n deep learning model validated through prototype inference to assess feasibility for future on-device deployment. The system was developed using Unity and ARCore and evaluated through user acceptance testing involving students, educators, and domain experts. Results demonstrate high usability, strong learner engagement, and improved learning performance, confirming the effectiveness of integrating immersive visualization, intelligent interaction, and gamified reinforcement in educational contexts. The findings highlight the practical potential of offline-first mobile learning platforms to support inclusive agricultural education and provide a scalable foundation for future intelligent educational systems in resource-constrained environments.**

Keywords*—Agricultural education; augmented reality; artificial intelligence; child-centered learning; game-based learning; offline mobile learning

I. Introduction

A. Background Information

Agriculture remains a fundamental sector of Sri Lanka's economy and food security [1], yet agricultural education at the primary level continues to face significant challenges in adopting interactive and technology-enhanced learning approaches [2]. Traditional pedagogical practices based on static textbooks and teacher-centered instruction often provide limited opportunities for experiential learning, resulting in reduced engagement among young learners and inadequate exposure to foundational agricultural concepts [3]. This issue is particularly critical in the context of paddy cultivation, a key element of Sri Lanka's rural economy and sustainable food production [4].

Sri Lanka's educational landscape continues to face significant challenges due to a persistent digital divide, even though national internet usage has surpassed 56%. A considerable portion of the population, about 43.7%, remains offline, particularly in rural communities where limited infrastructure and higher poverty levels of around 24.8% restrict access to digital technologies. As a result, many rural schools still depend heavily on traditional teaching approaches with limited exposure to modern digital learning resources. Although mobile technology has become the primary means of connectivity in these regions, access remains uneven and unreliable. Community initiatives such as "Nenasala" telecenters have attempted to bridge this gap by providing shared internet access, but their availability and consistency vary across locations. This situation highlights the importance of designing educational solutions that can operate effectively without requiring continuous internet connectivity [5].

This gap became evident through evaluations of existing school syllabi, classroom observations, and discussions with educators and parents. These stakeholders consistently highlighted the need for modernized, engaging, and visually rich learning materials. Despite the integration of digital resources into general science and geography education, there is a notable absence of purpose-built tools that leverage emerging technologies such as Augmented Reality (AR), Artificial Intelligence (AI), and Game-Based Learning (GBL) for agriculture- focused education at the primary level [6], [7].

Existing digital solutions provide only partial support. For instance, applications such as "FarmAR" offer AR-based visualization of farming equipment but lack structured educational frameworks and curriculum alignment [8]. Conversely, tools such as "Plantix" focus primarily on AI-based plant diagnosis designed for farmers rather than young learners [9]. These applications generally do not provide child-centered learning designs or offline functionality, both of which are essential for deployment in rural and low-connectivity contexts [10].

To address these limitations, this study introduces "Farm and Learn," an offline-first mobile educational system developed using Unity 6, Google ARCore, and Blender 4.3 [11], [12]. The proposed system delivers an immersive learning experience for children aged 10 years and above by integrating modular components, including AR-based exploration of 3D farming equipment, GBL-driven quizzes and interactive activities, and

an AI-based plant identification prototype utilizing a YOLOv11n convolutional neural network trained on a custom dataset [13]. Content development was guided by agricultural experts from the Rice Research Station, Labuduwa, while technological implementation was refined in collaboration with academic staff and students from the Department of Software Engineering, University of Kelaniya.

B. Research Problem and Research Questions

Despite the economic and social importance of agriculture in Sri Lanka, primary-level agricultural education continues to face significant limitations related to learner engagement, technological integration, and accessibility. Existing curriculum largely relies on passive, textbook-oriented instruction, which does not adequately support interactive or experiential learning for digitally native students [2]. This pedagogical gap contributes to reduced interest among younger generations in agriculture, particularly paddy cultivation, which remains a critical component of national food security and rural economic sustainability [3].

At the same time, currently available mobile applications in the agricultural domain do not provide comprehensive educational experiences specifically designed for children. While some applications offer isolated capabilities, such as AI-based plant disease detection (e.g., "Plantix") or AR-based visualization of agricultural objects (e.g., "FarmAR"), these systems typically lack curriculum-oriented learning structures, pedagogical scaffolding, and offline functionality suitable for rural educational deployment. More importantly, existing solutions rarely integrate AR, AI, and Game-Based Learning (GBL) within a unified framework to deliver immersive and holistic learning experiences tailored for young learners [14].

Unlike existing agricultural learning applications that focus on isolated functionalities such as AR visualization or AI-based plant recognition, including systems such as FarmAR [8] and Plantix [9], this work introduces a unified offline-first educational framework that integrates Augmented Reality (AR), lightweight AI-based plant identification, and Game-Based Learning (GBL) within a child-centered learning environment. The proposed system demonstrates how immersive technologies can be combined into modular educational architecture specifically designed for rural and low-connectivity contexts [17], [19], [20]. By integrating AR exploration, AI-driven interaction, and gamified learning within a single platform, this study extends prior work beyond single-purpose agricultural applications toward an integrated and pedagogically aligned learning solution for primary-level agricultural education [23], [25].

Given these limitations, a clear research gap emerges regarding the design and implementation of child-centered, offline-first educational systems that integrate immersive technologies for agricultural learning. Therefore, this study is guided by the following Research Questions (RQs):

- RQ1: Can immersive technologies such as AR, AI, and GBL improve learner engagement and educational outcomes in primary-level agricultural education?
- RQ2: Is it technically feasible to develop a fully offline-first and child-friendly mobile learning system suitable for rural and low-connectivity educational contexts?
- RQ3: Can AI-based visual recognition of paddy plant growth stages be effectively integrated into an interactive educational environment?

To address these questions, this research proposes the development of "Farm and Learn," an offline-first, context-aware mobile application that combines immersive technologies with pedagogically grounded content. The proposed system aims to increase children's engagement with agricultural concepts while providing an accessible, scalable, and technologically feasible solution for enhancing agricultural literacy in resource-constrained environments.

C. Significance of the Research

This research addresses a critical gap in primary-level agricultural education by introducing an immersive, technology-enhanced learning solution tailored for children. Despite the centrality of agriculture to Sri Lanka's economy, the subject receives limited emphasis in early education, with learning resources often restricted to outdated textbooks and static illustrations. This disconnect between traditional teaching methods and the expectations of today's digitally fluent learners highlights the need for a paradigm shift in educational delivery.

The significance of this project lies in its integration of Augmented Reality (AR), Artificial Intelligence (AI), and Game-Based Learning (GBL) into a unified, modular mobile application [15], [16]. The application not only provides structured agricultural knowledge but also engages learners through interactive visualizations, intelligent plant recognition, and gamified content. Developed using Unity and ARCore, with an AI model trained via Roboflow and implemented in JupyterLab using YOLOv11n, the system demonstrates the viability of deploying advanced educational technology in low-connectivity, rural environments through an offline-first architectural approach.

Beyond its educational objectives, this research contributes to national efforts in promoting food security, rural development, and digital transformation in education. By fostering early interest in sustainable farming and providing access to localized, child-friendly content, the project offers a scalable framework for integrating immersive technologies into primary education, particularly in rural regions.

II. Literature Review

A. Overview of Relevant Literature

Recent advancements in educational technology have increasingly emphasized the integration of immersive tools such as AR, AI, and GBL to enhance student engagement, motivation, and knowledge retention. Studies by Farooq et al. [17] and Kshirsagar et al. [18] demonstrated AR's effectiveness in simplifying abstract topics and improving test scores among young learners. Similarly, Zulfiqar et al. [19] provided a broad overview of AR's pedagogical benefits, including enhanced visualization and collaboration.

Game-Based Learning has also emerged as a strong motivational tool. Mohammed and Ozdamli [20] highlighted the effectiveness of badges, leaderboards, and feedback mechanisms in increasing student participation. Additional case studies by Gasca-Hurtado et al. [21] and Triantafyllou and Sapounidis [22] confirmed GBL's psychological and cognitive benefits in structured learning environments.

Artificial Intelligence is increasingly utilized in agriculture, primarily for disease detection and decision support. Studies such as Cheng et al. [23], Paulson and Ravishankar [24], and Paramathma et al. [25] illustrated CNNs' capabilities in image classification and precision agriculture applications. However, these solutions are designed primarily for professional or academic audiences.

Despite these advancements, few systems combine AR, AI, and GBL into a cohesive, offline-first educational solution targeted specifically at children in agriculture-heavy regions. Most tools reviewed are either single-purpose or targeted toward general academic subjects, highlighting a significant gap in subject-specific and age-appropriate technological solutions for agriculture education.

B. Key Theories or Concepts

The development of the "Farm and Learn" application is grounded in a blend of pedagogical and technical frameworks, primarily Constructivist Learning Theory, GBL, and Convolutional Neural Networks (CNNs) in AI.

Constructivist Learning Theory emphasizes experiential learning, wherein learners construct knowledge through direct engagement and contextual exploration. AR supports this theory by offering immersive, spatially- aware interactions. Studies reviewed, including Farooq et al. [17] and Kshirsagar et al. [18] highlighted AR's ability to enhance concept retention, especially when learners manipulate 3D objects within real-world settings.

GBL principles were another prominent framework across the literature. The works of Mohammed and Ozdamli [19] and Triantafyllou and Sapounidis [22] emphasized that interactive elements like feedback, quizzes, and progression tracking enhance student motivation and knowledge retention. In the context of "Farm and Learn," while reward systems like badges are not implemented, GBL is embedded through quiz-based learning and seed sorting mini-games to stimulate engagement and cognitive processing.

On the technical front, CNNs form the backbone of AI-driven plant identification. Literature by Cheng et al. [23] and Paulson and Ravishankar [24] demonstrated the efficacy of CNN-based architectures like VGG16, VGG19, and YOLO in plant classification and real-time inference tasks. The "Farm and Learn" project employs YOLOv11n for lightweight yet accurate plant recognition, trained via transfer learning on a custom dataset using Roboflow and developed in JupyterLab. These theories and frameworks collectively inform the design of the application, ensuring it is educationally grounded, technologically robust, and responsive to the learning needs of children aged 10 and above.

C. Gaps or Controversies in the Literature

Despite growing interest in educational technologies, significant gaps remain in the integrated use of AR, AI, and GBL within agriculture-focused learning particularly for younger audiences in low-resource environments.

Existing AR applications reported in the literature largely concentrate on general subject areas such as early literacy, mathematics, or abstract science concepts, with limited emphasis on agriculture-related knowledge, contextualized local practices, or culturally grounded learning environments [19], [20].

Similarly, while GBL has demonstrated clear benefits in learner motivation and retention, its adoption in agriculture education remains minimal. The reviewed GBL studies predominantly targeted domains like mathematics, accessibility awareness, or computer science, lacking examples tailored to primary-level agricultural education [21], [22]. Many GBL models are also heavily dependent on internet connectivity, real-time multiplayer modes, or reward systems that are less feasible in offline, rural settings.

In the realm of AI, recent literature highlights robust advancements in plant disease detection and smart agriculture using CNNs [23]. However, these systems are typically designed for adult farmers or researchers, with little consideration for integration into educational tools for children [24]. None of the surveyed AI systems offers curriculum-aligned, child-friendly interactions or simplified interfaces suitable for school use.

Moreover, existing educational systems tend to overlook the intersectionality of immersive technologies. Few solutions offer an integrated experience that combines AR exploration, AI-driven interaction, and gamified assessment in a unified learning flow. This fragmented technological landscape presents a missed opportunity to address cognitive and contextual learning needs in agricultural education for children.

The "Farm and Learn" application aims to fill these gaps by offering a unified, offline-first platform tailored for children aged 10 and above. It integrates immersive 3D exploration, real-time AI identification, and interactive games into a localized, age-appropriate framework, thereby addressing both the technological fragmentation and the pedagogical misalignment prevalent in existing literature.

III. Methodology

A. Research Design

The development of the "Farm and Learn" mobile application followed an Agile, user-centered research design to support the multidisciplinary integration of Augmented Reality (AR), Artificial Intelligence (AI), and Game-Based Learning (GBL) within a unified educational platform [26], [27]. The Agile methodology was selected to enable iterative refinement, rapid prototyping, and continuous stakeholder feedback throughout development. Short development cycles allowed progressive implementation and validation of functional modules while ensuring alignment with pedagogical objectives and technical feasibility.

The system was designed as a modular learning framework targeting children aged 10 years and above. Six core components were developed: (1) Learning Module, (2) Augmented Reality Module, (3) AI-Powered Paddy Plant Identification Module, (4) Game-Based Learning Module, (5) User Interface and Navigation Module, and (6) Offline-First Architecture. Each component was independently implemented while maintaining interoperability within the overall application.

B. System Architecture

The overall system architecture of the "Farm and Learn" application adopts a modular offline-first design to ensure reliable operation in low-connectivity educational environments. As illustrated in Fig. 1, the architecture consists of three conceptual layers: the presentation layer, application layer, and storage layer, supported by a future-ready synchronization mechanism.

The presentation layer manages user interaction through a child-centered interface and navigation controller, enabling seamless access to learning functions. The application layer integrates the core learning modules, including instructional content delivery, AR-based visualization, AI-driven plant identification, and game-based activities. Communication between these modules is coordinated through a centralized content delivery mechanism, which dynamically loads resources while maintaining runtime efficiency.

At the foundation of the system, a local storage module contains all instructional assets, multimedia resources, scripts, and assessment logic packaged within the APK during the Unity build process. This design eliminates reliance on cloud services or real-time APIs, ensuring uninterrupted system functionality in rural environments. In addition, a conceptual synchronization layer is included as a future extension to support content updates and learning analytics synchronization when internet connectivity becomes available, without affecting core offline functionality

1) Design rationale: The architectural design was guided by three key considerations: limited rural connectivity, hardware resource constraints of mobile devices, and the requirement to maintain immersive learner engagement. The offline-first strategy ensures accessibility regardless of network availability, while modular separation improves system maintainability and scalability. The centralized content delivery mechanism reduces dependencies among modules and enables independent expansion of system components. Consequently, the architecture balances educational usability, technical performance, and long-term extensibility.

2) Data flow overview: System execution begins at the presentation layer when users interact with learning modules through the application interface. The content delivery mechanism retrieves required assets from the local storage module and executes them directly on the device. User interactions, including scoring and feedback generation, are processed locally to ensure stable performance without internet dependency. When connectivity becomes available, the optional synchronization layer may support future content updates or analytics transfer without affecting core offline functionality.

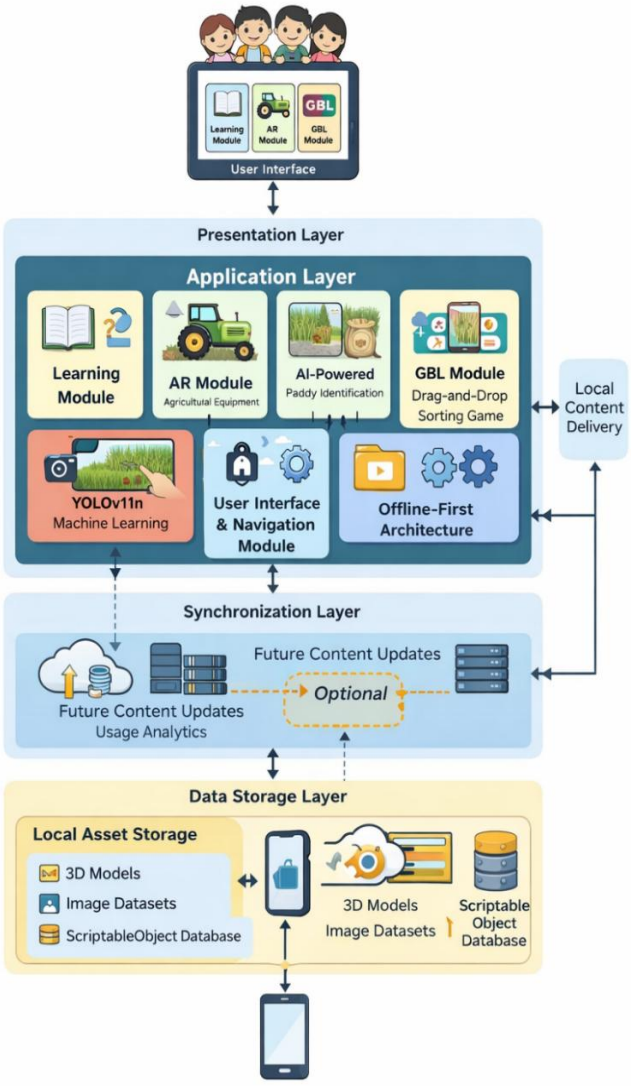


Fig. 1. Overall system architecture of the "farm and learn" application.

3) Architectural contribution: The proposed architecture contributes a lightweight, modular framework for delivering immersive educational experiences in connectivity-constrained environments. Unlike conventional mobile learning systems that rely on continuous cloud connectivity, the proposed design integrates AR visualization, AI-based recognition, and game-based interaction within an offline-first execution model. The separation between presentation, application, and storage layers enables scalable extension of modules while preserving system stability. Furthermore, the inclusion of a conceptual synchronization layer provides architectural flexibility for future cloud-enabled analytics and content updates without restructuring core system components. This architectural approach demonstrates a practical balance between immersive functionality, computational efficiency, and deployment feasibility for rural educational contexts.

C. Module Implementation

1) Learning module: The Learning Module delivers structured instructional content covering the paddy cultivation lifecycle, including land preparation, seed selection, irrigation, pest control, harvesting, and post-harvest practices. The module consists of three pedagogically aligned sub-components: Reading Materials, Interactive Tasks, and Interactive Tests.

a) Reading materials: Implemented using Unity UI Toolkit with scrollable layouts and hierarchical topic navigation, the reading materials incorporate illustrated infographics to reduce cognitive load and improve comprehension among young learners. All assets are optimized for mobile devices and designed for offline use.

b) Interactive tasks: The Interactive Tasks introduce gamified learning activities, such as seed classification using drag-and-drop mechanics. These tasks were developed using Unity's 2D physics engine and event-driven C# scripting to provide immediate visual feedback and reinforce concept application through experiential interaction.

c) Interactive tests: Interactive Tests evaluate learner understanding through multiple-choice quizzes executed entirely on-device. Questions are managed using ScriptableObject-based data structures, enabling scalable expansion without modifying source code. The quiz interface incorporates touch-friendly controls and immediate feedback to support formative learning.

2) Augmented Reality (AR) module: The Augmented Reality (AR) Module was designed to provide immersive and experiential learning by enabling children to interactively explore three-dimensional representations of agricultural equipment such as tractors, ploughs, sickles, and harvesting tools within real-world environments. The module was developed using Unity and Google ARCore, which facilitates markerless plane detection, feature-point mapping, and real-time spatial tracking to accurately position virtual objects on detected surfaces. The AR interaction workflow begins with environment sensing and surface detection, followed by object instantiation and pose alignment to maintain spatial consistency between virtual and physical spaces. Rendering enhancements such as scale normalization, lighting adaptation, and shadow projection are applied to improve visual realism and learner immersion. The overall AR placement and interaction process is illustrated in Fig. 2, which demonstrates the technical flow from surface detection to virtual object placement. Three-dimensional equipment models were primarily developed using Blender and optimized through polygon reduction and texture compression to ensure smooth rendering performance on mid-range Android devices. Once placed, learners can explore virtual farming tools from multiple viewing angles, promoting spatial understanding and exploratory learning behavior. Examples of AR-based farming tool visualization within real-world environments are presented in Fig. 3. In addition, voice narration provides child-friendly explanations to support multimodal learning and improve conceptual comprehension. Although AR functionality requires ARCore-supported hardware, the system preserves educational accessibility by allowing uninterrupted use of the learning and game-based modules when AR support is unavailable.



Fig. 2. ARCore-based surface detection and virtual object placement workflow in the proposed system.

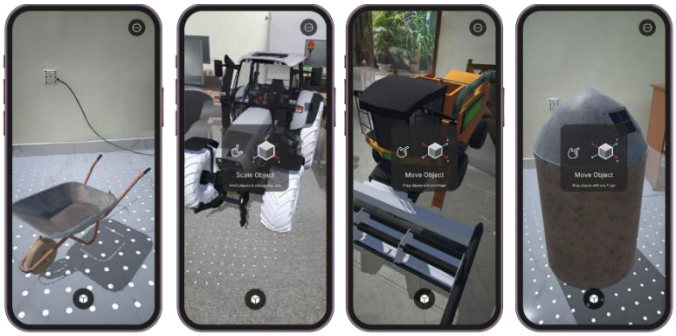


Fig. 3. Examples of AR-based farming tool visualization within the proposed learning system.

3) AI-Powered paddy plant identification module: The AI-powered paddy plant identification module supports recognition of paddy growth stages through deep learning–based image analysis. To validate model feasibility and inference performance while ensuring compatibility with the offline-first system design, the AI component was implemented as a prototype inference layer designed for seamless transition toward fully embedded on-device execution within the mobile application. This staged deployment strategy enables early evaluation of model accuracy while maintaining architectural consistency with the proposed offline learning framework. The module employs a YOLOv11n convolutional neural network optimized for lightweight inference [25]. Training data were collected from the Rice Research Station, Labuduwa, and supplemented with publicly available datasets to improve robustness and generalization. Data annotation and preprocessing were performed using the Roboflow platform, while model training was carried out in JupyterLab using Python with transfer learning from pretrained COCO weights [29], [30]. Hyperparameter tuning was applied to balance detection accuracy and computational efficiency, ensuring readiness for mobile deployment. The system generates annotated predictions indicating paddy detection results and demonstrates the practical feasibility of integrating AI-based plant recognition into child-centered educational learning

environments. The trained model is deployment-ready and can be packaged using lightweight mobile inference frameworks to enable fully offline execution within the application without requiring architectural modifications.

4) Game-Based Learning (GBL) module: The GBL Module reinforces learning outcomes through structured gamification strategies designed to increase engagement and motivation among learners. The module includes interactive mini-games and quiz-based assessments that promote concept application and formative evaluation. Activities such as seed-sorting tasks employ drag-and-drop interactions, real-time feedback, and scoring mechanisms to encourage repeated practice and experiential learning. Quiz components provide immediate correctness feedback and summary performance results, supporting knowledge retention while maintaining full offline functionality.

5) User interface and navigation module: The User Interface and Navigation Module coordinates interaction flow across all system components while ensuring accessibility for child users. The interface design emphasizes large touch-responsive elements, simplified iconography, and minimal text usage to reduce cognitive complexity. Navigation is structured around a centralized home interface supported by consistent scene transitions managed through script-based control mechanisms. This design ensures intuitive navigation, session continuity, and seamless movement between learning modules.

6) Offline-first architecture: The Offline-First Architecture was adopted to ensure full system functionality in rural and low-connectivity educational environments. All application assets, including 3D models, images, quiz logic, user interface scripts, audio instructions, and animation resources, are pre-packaged within the APK during the Unity build process, eliminating reliance on cloud services, real-time APIs, or continuous internet access. This design guarantees the uninterrupted availability of learning content, AR-based equipment visualization, and game-based learning activities regardless of connectivity conditions. From a technical perspective, local asset packaging improves performance efficiency and reduces runtime memory requirements, allowing stable execution on Android devices with limited hardware resources. ARCore-specific components, such as plane detection logic and tracking assets, are similarly embedded to maintain spatial interactivity while operating fully offline. Furthermore, the architecture enhances privacy and security by avoiding external data transmission and user account requirements, aligning with child data protection principles such as COPPA [31]. Overall, the offline-first deployment strategy improves accessibility, reliability, and scalability, making the system suitable for deployment across diverse rural educational contexts.

D. Data Collection Methods

A mixed-methods approach combining both qualitative and quantitative data collection was employed to guide the pedagogical design, content development, and technical implementation [32] of the "Farm and Learn" mobile application.

1) Primary data collection: Primary data were collected through field observations, expert consultations, focus-group sessions, and structured questionnaires [7]. Field visits to the Rice Research Station, Labuduwa, enabled the collection of agricultural imagery and contextual farming knowledge used for content design and AI dataset preparation. Structured interviews were conducted to validate the agricultural accuracy, pedagogical relevance, and technical feasibility of the system. Focus-group sessions and pilot testing activities were used to evaluate usability, engagement, and learner interaction with the application. Additionally, questionnaires were administered to capture user perceptions, learning preferences, and accessibility considerations.

2) Secondary data collection: Secondary data were obtained through an extensive review of peer-reviewed literature and academic conference publications related to AR, GBL, and AI in education. Insights from prior research informed the design of system features, pedagogical scaffolding approaches, and evaluation strategies. Additional reference material was sourced from authoritative agricultural databases and publications issued by the Department of Agriculture and other academic institutions to ensure factual accuracy, contextual relevance, and cultural appropriateness of the learning content on paddy cultivation.

E. Sample Selection

Participants were selected using a purposive sampling strategy to ensure representation of the key stakeholder groups relevant to the development and evaluation of the "Farm and Learn" application. The sample consisted of 47 children (students) aged 10–13 years, representing the primary end users of the system, together with 10 educators who evaluated pedagogical alignment and curriculum suitability, and 5 domain experts, including agricultural specialists and technical professionals who validated the accuracy of agricultural content and the technical feasibility of the solution. This multi-stakeholder sampling approach enabled balanced evaluation from learner, educational, and domain-expert perspectives, consistent with pilot studies in educational technology research [7].

F. Data Analysis Techniques

1) Qualitative analysis: A thematic analysis approach was applied to derive insights from user and expert-driven feedback. Focus-group discussions and expert interviews were coded to identify recurring usability patterns, interaction behaviors, and learner engagement characteristics. Survey responses were examined to determine preferred interaction modes and interface expectations among child users. Observational data collected during prototype testing were further used to refine module sequencing, navigation flow, and UI layout based on authentic user interaction with the system.

2) Quantitative evaluation of the AI model: To assess the feasibility and performance of the AI-based paddy-plant identification module, the YOLOv11n-based Convolutional

Neural Network (CNN) was trained and evaluated using annotated image datasets [28]. Model performance was measured using standard machine-learning metrics, including precision–recall curves, confusion matrices, and F1-score calculations [33]. Training and validation metrics were generated in JupyterLab across 100 training epochs using datasets annotated via the Roboflow platform, enabling systematic assessment of detection accuracy and generalization capability.

IV. Results

A. Experimental Evaluation Overview

The performance of the proposed "Farm and Learn" system was evaluated using user acceptance testing (UAT), expert validation, and technical benchmarking [34]. The evaluation involved 47 students, 10 educators, and 5 domain experts, focusing on usability, educational effectiveness, system performance, and AI model feasibility. Quantitative measurements were supplemented with qualitative feedback gathered during pilot deployment sessions.

B. User Acceptance and Educational Evaluation

Table I summarizes the average UAT ratings collected from students, educators, and experts using a five-point Likert scale. The results indicate consistently high acceptance across all stakeholder groups, with average scores exceeding 4.5 for all criteria. Offline accessibility achieved the highest rating (4.90), reflecting the effectiveness of the offline-first architecture in supporting rural deployment conditions. Visual engagement and educational relevance obtained strong scores (4.73), demonstrating the positive impact of immersive and contextually aligned learning content. The lowest relative score was observed for AR interactivity (4.50), suggesting opportunities for future improvements in interaction dynamics and device compatibility. Overall, the consistency of ratings across participants indicates strong agreement regarding usability and educational value.

TABLE I. UAT Evaluation Results (Average Score out of 5)

Evaluation criteria	*Students*	*Educators*	*Experts*	*Average Score*
Ease of Use	4.7	4.8	4.6	4.70
Visual Engagement	4.8	4.7	4.7	4.73
Educational Relevance	4.5	4.9	4.8	4.73
AR Interactivity	4.6	4.4	4.5	4.50
Game-Based Learning Appeal	4.7	4.6	4.4	4.57
Offline Accessibility	4.9	4.7	4.7	4.73
Overall Satisfaction	4.8	4.7	4.7	4.73

C. Learning Engagement and Performance

Observational findings indicated strong learner engagement during AR exploration and game-based activities. Average interaction time increased significantly compared to traditional content-based learning sessions, suggesting that immersive and interactive elements improved sustained attention.

Quiz-based assessments demonstrated progressive learning improvement, with average accuracy increasing from approximately 72% during initial attempts to 88% during repeated interactions. This improvement indicates that immediate feedback and gamified reinforcement contributed positively to knowledge retention.

D. AI Module Performance

The performance of the AI-powered paddy plant identification module was evaluated using standard classification metrics derived from the YOLOv11n model outputs. Table II presents the confusion matrix analysis summarizing the model's prediction results. The model correctly identified 13 Rice Plant instances, while 3 background samples were misclassified as rice plants, indicating strong detection capability with a small number of false positives.

TABLE II. Confusion Matrix Analysis of the AI Module

Predicted class	*True Rice Plant*	*True background*	*Total*
Rice Plant	13	3	16
Background	0	0	0
Total actual	13	3	16

The evaluation dataset contained only positive detections and background samples observed within the same image contexts; therefore, no independent background-only predictions were generated by the model during inference.

Fig. 4 shows stable performance across confidence thresholds, with a peak F1-score of 0.99 at a confidence value of 0.473, indicating a strong balance between precision and recall. Overall model accuracy achieved approximately 81.25%, confirming the feasibility of integrating lightweight AI-based visual recognition into an educational mobile learning environment.

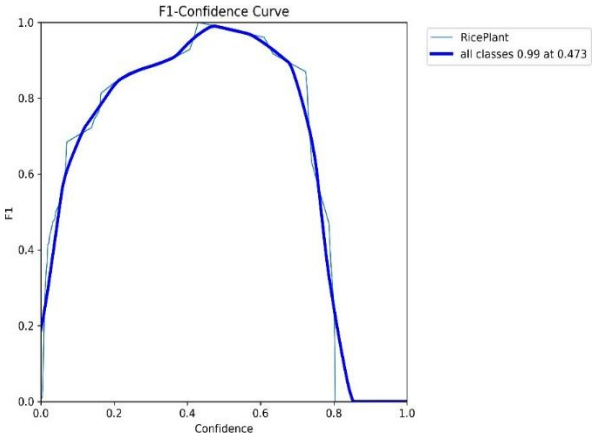


Fig. 4. F1-Confidence curve showing AI model performance for paddy plant identification.

E. System Performance Metrics

To assess technical feasibility, additional system-level performance measurements were collected on mid-range Android devices. Table III summarizes key quantitative metrics related to responsiveness, resource usage, and offline stability.

TABLE III. SYSTEM PERFORMANCE METRICS

Metric	*Observed value*	*Interpretation*
Application package size	~92 MB	Suitable for mid-range devices
Average startup latency	3.2 s	Smooth user experience
AR module loading time	8.1 s	Stable real-time interaction
Battery consumption (30 min session)	6-10%	Efficient mobile operation
Offline synchronization accuracy	100%	No connectivity dependency

The results confirm that the system operates efficiently under offline conditions while maintaining acceptable performance and energy consumption levels.

F. Comparison with Existing Systems

A comparison was conducted between the proposed *Farm and Learn* application and existing agricultural learning systems such as Plantix and FarmAR [8], [9]. As shown in Table IV, while Plantix primarily focuses on AI-based plant recognition and FarmAR emphasizes AR visualization, neither system provides a unified child-centered learning environment with offline educational functionality.

TABLE IV. COMPARISON WITH EXISTING SYSTEMS

Feature	*Plantix*	*FarmAR*	*Farm and Learn*
AR Visualization	No	Yes	Yes
AI-Based Recognition	Yes	No	Yes
Game-Based Learning	No	No	Yes
Child-Oriented Design	No	No	Yes
Offline Capability	No	Partial	Yes

The comparison demonstrates that the proposed system integrates AR, AI, and game-based learning within a single offline-first platform, making it more suitable for primary-level agricultural education in rural contexts.

G. Support for Research Questions

1) RQ1: Supported: The results strongly support this research question. High user acceptance scores across all stakeholder groups (Table I), increased learner interaction time, and improved quiz performance (72% to 88%) demonstrate that immersive and interactive technologies positively influenced learner engagement and knowledge retention.

2) RQ2: Supported: The findings confirm feasibility. Offline accessibility achieved one of the highest evaluation scores (4.73), and system testing demonstrated stable performance without internet dependency, validating the effectiveness of the offline-first architecture for rural educational deployment.

3) RQ3: Supported: The AI module achieved a peak F1-score of 0.99 with an overall accuracy of approximately 81.25%, as illustrated in Table II and Fig. 4. These results confirm that lightweight AI-based plant identification can be successfully integrated into educational mobile systems while maintaining practical performance levels.

Overall, the results affirm the feasibility and pedagogical value of the "Farm and Learn" platform as a scalable, immersive, and inclusive digital tool for enhancing primary-level agricultural education in Sri Lanka, particularly in the domain of sustainable paddy cultivation.

V. DISCUSSION

A. Interpretation of Results

The results suggest that the pedagogical synergy between AR, AI, and GBL contributes directly to engagement and knowledge retention, reinforcing prior evidence on multimodal learning environments. High user acceptance scores and improved quiz performance suggest that interactive and visual learning approaches helped learners better understand agricultural concepts. These findings support previous research indicating that immersive and gamified learning improves motivation and knowledge retention by promoting active participation [17], [19], [20]. Feedback from educators and experts further confirmed that the system maintained both educational relevance and technical feasibility.

B. Conceptual Contribution of AR, AI, and GBL Integration

A key contribution of this work is the combination of AR, AI, and GBL within a single educational framework. Each component plays a complementary role: AR enhances visual and spatial understanding, AI introduces intelligent interaction through plant recognition, and GBL strengthens motivation and reinforcement through interactive activities [17], [23], [25]. Together, these technologies create a balanced learning environment where exploration, feedback, and practice occur simultaneously. This integrated design helps explain the high engagement levels observed during evaluation and demonstrates that multimodal learning approaches may be more effective than isolated technology-driven solutions.

C. Comparison with Existing Literature

Compared to existing platforms such as Plantix and FarmAR [8], [9], the proposed system extends prior approaches by combining multiple immersive technologies within a child-oriented educational framework. While Plantix focuses primarily on AI-based plant recognition and FarmAR emphasizes AR visualization, neither system provides a complete learning environment that integrates structured pedagogy, gamification, and offline usability.

The findings of this study therefore, align with recent research advocating hybrid learning environments that combine visualization, interaction, and intelligent feedback to improve educational outcomes [17], [19]. In addition, the localization of content to Sri Lankan paddy cultivation provides contextual relevance often missing from globally generalized agricultural applications, strengthening both authenticity and learner engagement.

D. Implications and Limitations of the Study

The findings highlight the potential of immersive educational technologies to enhance engagement and accessibility in primary-level agricultural education. The integration of AR, AI, and game-based learning demonstrates how interactive and context-aware digital tools can support learner motivation and experiential learning while remaining

technically feasible for rural deployment through an offline-first architectural approach [17], [19], [23]. These results suggest that similar multimodal learning frameworks can be applied to other educational domains requiring practical and visual understanding, particularly in low-resource environments [20], [25].

The proposed system adopts a modular architecture that supports scalability in both content and user expansion. Learning modules, AR assets, and AI components can be extended independently without modifying the core system structure. In addition, the offline-first design reduces server dependency while allowing optional synchronization, enabling gradual deployment across larger educational environments as user numbers and learning content grow.

However, several limitations should be acknowledged. First, AR functionality depends on ARCore-compatible devices, which may restrict accessibility for learners using older or low-end smartphones [18], [19]. Second, the AI-based plant identification component currently operates as a prototype and requires further optimization for full offline deployment using lightweight mobile inference frameworks [23], [25]. Third, the current evaluation emphasizes short-term usability and engagement; therefore, larger-scale and longitudinal studies are needed to examine long-term learning outcomes and system scalability [17], [20].

VI. Conclusion

This study presented the design and evaluation of "Farm and Learn," an offline-first mobile learning application that integrates Augmented Reality (AR), Artificial Intelligence (AI), and Game-Based Learning (GBL) to enhance agricultural education for children. The findings demonstrate that combining immersive visualization, intelligent recognition, and interactive learning activities within a unified framework can improve learner engagement, usability, and perceived educational value, particularly in the context of primary-level agricultural learning [17], [19], [20]. The modular system architecture and offline-first design proved effective for deployment in rural and low-connectivity environments, addressing accessibility challenges commonly identified in educational technology adoption [18], [19].

A key contribution of this work lies in demonstrating the conceptual synergy between AR, AI, and GBL, where each component supports complementary pedagogical functions including experiential visualization, intelligent interaction, and motivational reinforcement [23], [25]. Unlike existing systems that focus on isolated functionalities, the proposed approach provides a child-oriented and contextually localized learning environment tailored to Sri Lankan agricultural practices. The study also shows the feasibility of integrating lightweight AI-based plant recognition within educational applications, supporting future directions for intelligent and adaptive learning tools.

Despite these contributions, certain limitations remain. The AR module depends on ARCore-compatible devices, and the AI component currently operates as a validated prototype inference layer pending final mobile packaging requiring further optimization for full mobile integration. In addition, evaluation focused primarily on short-term usability and acceptance; therefore, future work should include larger-scale longitudinal studies to assess long-term learning outcomes and scalability [17], [20]. Further development could also incorporate multilingual content and advanced adaptive learning features to enhance inclusivity and educational impact. Overall, the study demonstrates that immersive, offline-first mobile learning platforms represent a promising direction for sustainable and accessible agricultural education in resource-constrained environments.

Acknowledgment

This work was made possible through the generous support and collaboration of several institutions and individuals. The author gratefully acknowledges the Rice Research Station, Labuduwa, especially the Deputy Director and team, for their guidance, field-access, and expert contributions, which were invaluable to the development of the learning content and dataset preparation. Sincere appreciation is also extended to the Department of Software Engineering, Faculty of Computing and Technology, University of Kelaniya, for their technical insights and development-phase support, and to the Faculty of Information Technology, Horizon Campus, Malabe, for providing an enabling academic and research environment.

The authors thank the educators, students, and schoolteachers who participated in pilot evaluations, as well as contributors of open-access image and 3D-model resources used for academic purposes. Finally, heartfelt thanks are extended to family, colleagues, and friends for their encouragement and support throughout this work.

References

[1] P. Anusha and S. Vijesandiran, "An empirical analysis of the effect of agricultural sector determinants on economic growth in Sri Lanka," Business and Economic Research, vol. 12, no. 2, p. 155, Jun. 2022, doi: 10.5296/ber.v12i2.19823.

[2] U. Dissanayeke, P. Prasada, and H. Wickramasuriya, "ICT-Based Information Systems in Agricultural Extension and Their Economic Implications: Sri Lankan Perspectives," 2020, p. 331. doi: 10.1007/978-981-15-3673-1_15.

[3] T. Halloluwa, D. Vyas, H. Usoof, P. Bandara, M. Brereton, and P. Hewagamage, "Designing for Financial Literacy: Co-design with Children in Rural Sri Lanka," in *Lecture notes in computer science*, Springer Science+Business Media, 2017, p. 313. doi: 10.1007/978-3-319-67744-6_21.

[4] S. Kularathne, N. Rathnayake, M. Herath, U. Rathnayake, and Y. Hoshino, "Impact of economic indicators on rice production: A machine learning approach in Sri Lanka," PLoS ONE, vol. 19, no. 6, Jun. 2024, doi: 10.1371/journal.pone.0303883.

[5] A. Cosby, J. Manning, D. Power, and B. Harreveld, "New Decade, Same Concerns: A Systematic Review of Agricultural Literacy of School Students," *Education Sciences*, vol. 12, no. 4. Multidisciplinary Digital Publishing Institute, p. 235, Mar. 24, 2022. doi: 10.3390/educsci12040235.

[6] S. Kemp, "Digital 2024: Sri Lanka — DataReportal – Global Digital Insights," DataReportal – Global Digital Insights, Feb. 23, 2024.

[7] F. L. Vallera and A. M. Bodzin, "Integrating STEM with AgLIT (Agricultural Literacy Through Innovative Technology): The Efficacy of a Project-Based Curriculum for Upper-Primary Students," *International Journal of Science and Mathematics Education*, vol. 18, no. 3, p. 419, Apr. 2019, doi: 10.1007/s10763-019-09979-y.

[8] A. Katsaros and E. Keramopoulos, *FarmAR, a farmer's augmented reality application based on semantic web*. 2017, pp. 1–6. doi: 10.23919/seeda-cecnsm.2017.8088230.

[9] T. N. Hashem, J. M. M. Joudeh, and A. M. Zamil, "Smart Farming (AI-Generated) as an approach to Better Control pest and Disease detection in agriculture: POV Agricultural Institutions," *MIGRATION LETTERS*, vol. 21, no. S1, pp. 529–547, Dec. 2023, doi: 10.59670/ml.v21is1.6178.

[10] N. Gondiawati and S. Nurhayati, "Child-Centered Learning in the home Environment: Implementation and challenges," *Jurnal Simki Pedagogia*, vol. 7, no. 1, pp. 13–23, Jan. 2024, doi: 10.29407/jsp.v7i1.502.

[11] Y. Song, K. Wu, and J. Ding, "Developing an immersive game-based learning platform with generative artificial intelligence and virtual reality technologies – 'LearningverseVR,'" *Computers & Education X Reality*, vol. 4, p. 100069, Jan. 2024, doi: 10.1016/j.cexr.2024.100069.

[12] A. D. Samala *et al.*, "3D Visualizations in Learning: An Evaluation of an AR+Core Application for Computer Hardware Education using the Hedonic Motivation System Adoption Model," *TEM Journal*, p. 466, Feb. 2024, doi: 10.18421/tem131-48.

[13] D. S. Topwal and R. Mittal, "Custom Object Detection using YOLOV11 model," *International Scientific Journal of Engineering and Management*, vol. 04, no. 12, pp. 1–9, Dec. 2025, doi: 10.55041/isjem05242.

[14] A. R. Balcha, H. Chen, and R. R. A.B., "The impact of augmented reality-assisted structured learning on environmental education for preschool children," *Education and Information Technologies*, Sep. 2025, doi: 10.1007/s10639-025-13737-9

[15] D. Egunjobi and O. Adeyeye, "Revolutionizing Learning: The Impact of Augmented Reality (AR) And Artificial Intelligence (AI on Education," *International Journal of Research Publication and Reviews*, vol. 5, no. 10, p. 1157, Oct. 2024, doi: 10.55248/gengpi.5.1024.2734

[16] M. J. Ali, "Developing Augmented Reality based Gaming Model to Teach Ethical Education in Primary Schools," *arXiv (Cornell University)*, Jan. 2020, doi: 10.48550/arxiv.2010.15346.

[17] S. S. Farooq, H. Rahman, S. A. N. Raza, M. Raees, and S. K. Jung, "Designing Gamified Application: an effective integration of augmented reality to support learning," *IEEE Access*, vol. 10, pp. 121385–121394, Jan. 2022, doi: 10.1109/access.2022.3221473.

[18] N. Kshirsagar, G. Pandey, A. Prakash, I. S. Chauhan, and S. Kumar, "Educative Reality - Augmented Reality Application for Education," *In 2023 2nd International Conference for Innovation in Technology (INOCON)*, pp. 1–6, Mar. 2023, doi: 10.1109/inocon57975.2023.10101270.

[19] F. Zulfiqar, R. Raza, M. O. Khan, M. Arif, A. Alvi, and T. Alam, "Augmented Reality and its Applications in Education: A Systematic Survey," *IEEE Access*, vol. 11, pp. 143250–143271, Jan. 2023, doi: 10.1109/access.2023.3331218.

[20] Y. B. Mohammed and F. Ozdamli, "Motivational Effects of Gamification apps in Education: A Systematic Literature review," *BRAIN BROAD RESEARCH IN ARTIFICIAL INTELLIGENCE AND NEUROSCIENCE*, vol. 12, no. 2, Jul. 2021, doi: 10.18662/brain/12.2/196.

[21] G. P. Gasca-Hurtado, M. C. Gomez-Alvarez, J. A. Hincapie, and V. V. Zepeda, "Gamification of an Educational environment in Software Engineering: Case study for digital accessibility of people with Disabilities," *IEEE Revista Iberoamericana De Tecnologias Del Aprendizaje*, vol. 16, no. 4, pp. 382–392, Nov. 2021, doi: 10.1109/rita.2021.3137372.

[22] S. A. Triantafyllou and T. Sapounidis, "Game-based Learning approach and Serious Games to learn while you play," *In 2023 IEEE World Engineering Education Conference (EDUNINE). Bogota, Colombia: IEEE*, pp. 1–6, Mar. 2023, doi: 10.1109/edunine57531.2023.10102872.

[23] Q. Cheng, S. Zhang, S. Bo, D. Chen, and H. Zhang, "Augmented Reality Dynamic image recognition technology based on deep learning algorithm," *IEEE Access*, vol. 8, pp. 137370–137384, Jan. 2020, doi: 10.1109/access.2020.3012130.

[24] A. Paulson and S. Ravishankar, "AI Based Indigenous Medicinal Plant Identification," *In 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA). Cochin, India: IEEE*, Jul. 2020, doi: 10.1109/accthpa49271.2020.9213224.

[25] K. Paramathma, M. A. Kumar, P. Ranjith, and C. S. S. Reddy, "Development of AI Based App For Precision Agriculture in India," *In 2023 IEEE Renewable Energy and Sustainable E-Mobility Conference (RESEM). Bhopal, India: IEEE*, vol. 48, pp. 1–5, May 2023, doi: 10.1109/resem57584.2023.10236293.

[26] E. Mekić, M. R. Jovanović, K. Kuk, B. Prlinčević, and A. Savić, "Enhancing Educational Efficiency: Generative AI Chatbots and DevOps in Education 4.0," *arXiv (Cornell University)*, Apr. 2024, doi: 10.48550/arxiv.2406.15382.

[27] S. Khan, A. Mahadik, A. Pal, and S. Jacob, "'Decal App'- Augmented Reality based Flutter Application," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 4, p. 798, Apr. 2023, doi: 10.22214/ijraset.2023.50208.

[28] N. T. T. Hà, D. D. Khanh, T. Hung, and C. V. Kien, "YOLO11n-RP: Lightweight Deep Learning Model for the Real-Time Detection Based on YOLOv11," in *Communications in computer and information science*, Springer Science+Business Media, 2025, p. 172. doi: 10.1007/978-3-031-98164-7_14.

[29] S. Singh, A. Yadav, J. Jain, H. Shi, J. Johnson, and K. Desai, "Benchmarking Object Detectors with COCO: A New Path Forward," in *Lecture notes in computer science*, Springer Science+Business Media, 2024, p. 279. doi: 10.1007/978-3-031-72784-9_16.

[30] Ultralytics, "COCO Dataset," *Ultralytics YOLO Docs*, Mar. 17, 2025. https://docs.ultralytics.com/datasets/detect/coco/

[31] "COPPA – the law protecting children's online privacy," *Online Security & Control Organizations*, Aug. 30, 2024. https://www.coppa.org/coppa/

[32] C. Costopoulou, S. Karetsos, and M. Ntaliani, "Investigating educational mobile apps for agriculture," *International Journal of Sustainable Agricultural Management and Informatics*, vol. 6, no. 3, p. 272, Jan. 2020, doi: 10.1504/ijsami.2020.112124.

[33] J. Hernández-Orallo, "AI Evaluation: past, present and future," *arXiv (Cornell University)*, Jan. 2014, doi: 10.48550/arxiv.1408.6908.

[34] T. Gire, "Human-Centered AI in Smart Farming: Toward Agriculture 5.0," *International Journal for Research in Applied Science and Engineering Technology*, vol. 13, no. 10, p. 1569, Oct. 2025, doi: 10.22214/ijraset.2025.74813

An Enhanced Approach for Workmen's Compensation Insurance Fraud Detection Based on Fuzzy Rule-Based System

Reham M. Essa
Department of Information Systems,
Higher Institute for Management and Information Technology, Kafr El-Sheikh, Egypt

***Abstract*—In Workmen's Compensation insurance, fraud detection (FD) remains a significant challenge due to claims' inherent uncertainty and complexity. To address this, we propose an enhanced approach based on a fuzzy rule system (FRS) for FD. The FRS is designed to handle ambiguous and imprecise data, making it effective for identifying fraudulent patterns in insurance claims. Unlike traditional methods, the fuzzy system utilizes human-like reasoning by applying flexible rules to assess the likelihood of fraud under uncertain conditions. By modeling the decision-making process with fuzzy logic, the system allows for a detailed evaluation of claims, accommodating the gray areas that often exist in FD. This approach enables accurate and adaptive FD, reducing false positives and enhancing the precision of fraud identification. In imbalanced FD scenarios, the system achieves strong performance, such as an F1-score of 0.82 and MCC of 0.75, demonstrating its capability to correctly identify rare fraudulent cases despite class imbalance.**

***Keywords*—Workmen's compensation; fuzzy logic; fraud detection; rule-based system; insurance fraud; prediction**

I. Introduction

The insurance industry plays a vital role in supporting the economic stability of a country, mainly through employee compensation insurance, which provides coverage for workers injured on the job. This type of insurance helps cover wage loss, medical expenses, and other benefits related to work-related injuries. Nevertheless, it also exposes companies to potential fraud, as unethical employers, workers, and medical providers may exploit the system for illegal financial gain. Fraudulent activities can involve deliberate deception, misrepresentation, or violating laws and policies to obtain unauthorized benefits. Combating fraud requires sophisticated detection methods, as fraud can be challenging to identify due to its intentional manipulation of the system and the complex, often hidden, nature of these activities [1].

Fraud in Workmen's Compensation insurance can take many forms, each adding significant financial strain to the system and impacting both employers and employees. One common type is employee fraud, where individuals claim non-work-related injuries as work-related to receive benefits. This may involve exaggerating the severity of an injury or misrepresenting the timing of the injury. Employer fraud occurs when companies misclassify employees or underreport their payroll to lower insurance premiums, resulting in reduced contributions and increased risks for insurers. Healthcare provider fraud involves billing for unnecessary or nonexistent medical treatments, inflating costs, and further burdening the insurance system. Lastly, insurance carrier fraud can include wrongful denial of claims or manipulating information to avoid paying out legitimate claims. These types of fraud collectively undermine trust in the system, increase premiums for honest participants, and require sophisticated detection mechanisms to manage and mitigate their impact [2],[3].

A FRS is effective for handling uncertainty in detecting fraud in Workmen's Compensation insurance by allowing for flexible decision-making in cases where data is ambiguous or incomplete. Unlike traditional techniques that use rigid, binary classifications (fraudulent or not), fuzzy systems handle data in degrees, assigning partial membership to different categories, such as "somewhat suspicious" or "likely fraudulent." This flexibility enables the system to model real-world uncertainty accurately, reducing false positives and adapting more effectively to complex, evolving fraud patterns. Additionally, fuzzy systems incorporate expert knowledge and subjective judgment, translating it into actionable rules, which traditional methods struggle to do. This results in a more nuanced, precise, and adaptive FD system that is better suited for the multifaceted nature of insurance fraud [4]-[6].

Detecting fraud in Workmen's Compensation insurance is a challenging task due to the varying and complex nature of fraudulent activities. Fraud indicators, such as suspicious reporting times, data behavior patterns, case correlations, and demographic factors, are critical for insurance experts to detect fraud. On the other hand, subjective judgments from experts can lead to inconsistencies in decision-making, and managing large volumes of data manually is time-consuming, prone to errors, and ineffective in fraud prevention. Furthermore, the growing data collected by insurance companies is often underutilized, missing opportunities to detect hidden fraud patterns. To address the challenges of detecting fraud in Workmen's Compensation insurance, this research proposes a FRS for FD. Unlike traditional systems, a FRS can handle uncertainty and imprecision in data, making it especially useful for complex FD scenarios where decisions must be made based on incomplete or vague information. Fraud indicators like suspicious behavior, reporting time discrepancies, and demographic factors often involve degrees of uncertainty that rigid systems struggle to process.

Fuzzy logic allows the system to mimic human reasoning by evaluating these factors in a flexible manner, providing more nuanced and accurate FD. This approach improves decision-making by accounting for varying fraud patterns and expert judgments, minimizing the risk of errors and inconsistencies. Additionally, the system enhances the ability of insurers to process large amounts of data efficiently, reducing costs and extracting hidden patterns that manual analysis would likely miss. Ultimately, it supports long-term business sustainability by providing more reliable FD and better resource allocation for claim investigations [7],[8].

A. Problem Statement and Motivation

The primary problem in Workmen's Compensation insurance FD is the inherent uncertainty and complexity of claims, which makes it difficult to accurately identify fraudulent activities. Traditional FD methods fail to handle ambiguous data and often rely on rigid, binary decision-making processes, leading to either missed fraud cases or false positives. Fraud in this domain frequently involves subtle patterns and subjective factors, such as claim timing, injury severity, or inconsistent reporting, which can vary greatly between cases. To address these challenges and motivated by the need for more precise and adaptable detection systems, this research proposes a fuzzy rule-based approach that leverages the flexibility of fuzzy logic to handle imprecise data.

B. Novelty and Contribution

The novelty of the proposed research lies in its exclusive use of an FRS for detecting fraud in Workmen's Compensation insurance, which distinguishes it from traditional methods. While conventional FD techniques often rely on strict, binary classifications that struggle with ambiguous data, this research leverages fuzzy logic to handle the uncertainty and complexity inherent in insurance claims. The fuzzy system's ability to model human-like reasoning by applying flexible, adaptable rules is a significant advancement in the domain of FD. It addresses the common challenge of managing imprecise and subjective information, such as variations in injury reporting or inconsistent claim histories, which are often difficult to capture using rigid systems.

The main contribution of this research is the development of an intelligent, adaptive framework that reduces false positives and improves the precision of FD. By offering a more nuanced evaluation of claims, this fuzzy rule-based approach enhances decision-making capabilities and allows insurance companies to more effectively identify fraudulent patterns. This ultimately contributes to the sustainability and integrity of the Workmen's Compensation insurance system, ensuring fair claims processing and reducing unnecessary costs caused by fraud. The proposed system's ability to extract hidden patterns from data and adapt to changing fraud schemes is another critical advancement, positioning it as a valuable tool in minimizing fraud risk.

The rest of the study is organized as follows, in section II state of the arts. In section III, we discussed and presented the research methodology in detail. Section IV presented the experiment and a discussion of the results followed by a conclusion.

II. Related Works

In the field of Workmen's Compensation insurance FD, several state-of-the-art techniques have emerged, each with its advantages and disadvantages. These methods can be broadly classified into data mining techniques, machine learning algorithms, rule-based systems, and hybrid approaches that combine multiple models for enhanced FD [9]-[15]. Data mining techniques like clustering (e.g., K-Means) and association rule mining are widely used to group similar insurance claims and identify hidden patterns in large datasets. These methods can efficiently process vast amounts of data, helping detect fraud by flagging anomalies. However, they rely heavily on historical data, making them less adaptable to evolving fraud tactics, and they can lead to high false-positive rates, incorrectly identifying legitimate claims as fraudulent [9], [12],[13].

Machine learning algorithms, including supervised and unsupervised models like random forest, support vector machines (SVM), and Neural Networks, are increasingly popular for FD. These models excel at learning from new data, detecting complex and non-linear relationships between variables, and are effective at identifying fraud patterns with high accuracy when trained on quality datasets. However, they require large, well-labeled datasets for effective training, which can be difficult to acquire. Additionally, many machine learning models, particularly deep learning networks, function as black-box systems, making their decision-making process difficult to interpret, which can lead to trust issues in critical applications such as FD [16]-[19], [20]-[23].

Traditional rule-based systems use predefined rules, usually set by experts, to classify claims as fraudulent or non-fraudulent. These systems are transparent and easy to understand, as they follow clear, interpretable rules, allowing for quick decision-making and reliability in detecting known fraud patterns. However, they lack flexibility since they can only detect fraud based on the programmed rules, making them ineffective against new or evolving fraud schemes. This limitation often leads to false negatives, where fraudulent cases are missed because they don't fit the established patterns. FRS, an advanced version of traditional rule-based approaches, utilize fuzzy logic to manage uncertainty and imprecision in FD. They excel in cases where data is ambiguous or conflicting, allowing them to model complex human reasoning and effectively handle gray areas. This flexibility reduces false positives and improves the detection of subtle fraud patterns that rigid systems might overlook. However, designing and maintaining fuzzy systems can be more challenging, as their performance heavily relies on the quality of the crafted rules and the expertise required, which can introduce subjectivity [22]-[25].

Hybrid systems that combine machine learning models with rule-based systems or fuzzy logic offer a powerful approach to leveraging the strengths of each method. By integrating machine learning, these systems can adapt to emerging fraud patterns through data-driven insights and continuous learning, making them highly flexible and responsive. Simultaneously, the rule-based components bring transparency and domain expertise that make decision-making more interpretable and understandable, particularly in industries requiring accountability, such as

finance or healthcare. This balance between adaptability and transparency provides an advantage in terms of flexibility, accuracy, and control. However, these systems also come with significant challenges. Their complexity makes implementation difficult, often requiring extensive computational resources to manage the interactions between the machine learning and rule-based components effectively. Additionally, maintaining a system that remains both transparent and accurate over time necessitates ongoing tuning and optimization, as misalignment between the components could compromise either its clarity or predictive power [12],[13], [20]-[23].

Pallavi A. et al. [26] carried out a detailed study that aimed to design an analytical framework for identifying potentially fraudulent health insurance claims. The methodology combined machine learning-based predictive modeling with a rule-based scoring system. Historical claims data were analyzed using data mining techniques to uncover patterns and correlations associated with fraud, enabling the model to assign probabilistic fraud scores to new claims based on similarities with known fraudulent behaviors. Alongside this, a set of predefined rules—such as abnormal treatment costs, high claim frequency, or unusual claimant behavior—was used to assign additional risk scores. Claims with higher cumulative scores were flagged for further investigation. This hybrid approach offered improved accuracy, scalability for large datasets, and adaptability through continuous learning. However, it also presented challenges, including dependence on high-quality data, limited flexibility in detecting new fraud patterns due to rigid rules, and the possibility of false positives, which could impact genuine claimants.

Kirlidog and Asuk [27] conducted a study on health insurance FD by focusing on anomaly classification techniques to distinguish fraudulent claims from legitimate ones. Their methodology centered on analyzing historical insurance data to identify patterns of irregularities that deviate from typical claim behavior. By applying data mining and statistical analysis methods, they were able to classify anomalies that often signify fraudulent activity, such as inconsistencies in billing, unusual claim frequencies, or improbable medical procedures. This approach allowed for the detection of fraud by recognizing outliers in the dataset rather than relying solely on predefined rules. The key advantages of this method include its ability to uncover previously unknown fraud patterns and its adaptability to different datasets. However, it also has limitations, such as a high dependency on the quality and completeness of the input data, and a potential for false positives where legitimate but atypical claims may be flagged as fraudulent, necessitating further manual review.

This study, presented in Ref. [24], introduced a novel hybrid methodology for detecting fraud in automobile insurance claims by integrating a Genetic Algorithm (GA)-enhanced Fuzzy C-Means (FCM) clustering technique with multiple supervised classification models. The approach begins by dividing the original insurance dataset into a test set and a training set. The training data undergoes FCM clustering, optimized using GA, to perform undersampling and create meaningful, balanced clusters. Test instances are then categorized into three classes: genuine, malicious, or suspicious, based on their proximity to these clusters. Genuine and clearly fraudulent claims are filtered out, while suspicious cases undergo further evaluation using four supervised classifiers: Decision Tree (DT), (SVM), Group Method of Data Handling (GMDH), and Multi-Layer Perceptron (MLP). Model training and validation are performed using 10-fold cross-validation to ensure reliability. The strengths of this method lie in its ability to handle imbalanced data through intelligent clustering, reduce false positives by isolating uncertain cases, and improve accuracy through ensemble analysis. However, the system's complexity and computational cost are potential drawbacks, and its performance is highly dependent on proper tuning of clustering parameters and classifier settings.

In Ref. [25], the authors suggested a machine learning-based approach for detecting insurance fraud using a supervised classification algorithm trained on historical claim data, which includes both legitimate and fraudulent cases. The model utilizes various features to differentiate between genuine and suspicious claims. Evaluated on real-world insurance data, the approach demonstrated high accuracy and was effective in identifying fraudulent claims that traditional rule-based methods often miss. Its main strengths include improved detection rates, automation of the fraud identification process, and reduced workload for human investigators. However, the approach relies heavily on the quality and representativeness of the training data, and its performance may decline if fraudulent behavior patterns change or if unseen types of fraud are introduced.

Workmen's FD is inherently complex due to the multifaceted and often hidden nature of fraudulent behavior. This necessitates the use of advanced data mining techniques capable of transforming fragmented and seemingly unrelated data into coherent, actionable insights. A foundational step involves integrating data from diverse sources into a centralized, structured data repository to facilitate comprehensive analysis. In this context, Helmut F., et al. [28] used an explainable attention-based neural network for claim FD. This model uses attention mechanisms within deep learning architectures to focus on the most relevant parts of the data, enhancing interpretability while maintaining high performance. The key advantage of this method is its transparency—decision-making can be better understood by auditors and investigators, fostering trust in AI-driven processes. However, it can be computationally intensive and requires substantial training data to perform effectively, particularly in complex, high-dimensional FD environments.

Additionally, Sadgali I. et al. [29] investigated the performance of various machine learning techniques in detecting financial fraud, comparing algorithms such as Decision Trees, Random Forests, SVM, and Neural Networks. The study highlighted that ensemble methods like Random Forests generally achieved better accuracy and robustness due to their ability to reduce overfitting and model nonlinear relationships. However, such models can lack interpretability, making it difficult to understand how fraud decisions are made. Similarly, Ryman, T. et al. [30] conducted a survey and benchmarking study on how AI and machine learning are transforming payment card FD. The research emphasized the growing use of real-time anomaly detection, deep learning, and hybrid AI systems in the financial industry. While these models significantly enhance detection capabilities and reduce false

positives, their implementation involves high computational costs, complex deployment requirements, and the need for continuous updating to keep pace with evolving fraud tactics.

A. Research Gap

Based on the comprehensive survey of current methodologies in Workmen's Compensation insurance FD, a notable research gap emerges in the underutilization of uncertainty-based fuzzy rule mining techniques—particularly in environments where data is ambiguous, imprecise, or incomplete. While fuzzy logic has been applied in some advanced rule-based systems to manage uncertainty, most existing implementations either rely on traditional, rigid rule-based approaches or focus heavily on machine learning and data-driven models that often operate as black boxes. These methods may lack the ability to effectively handle borderline cases or precise fraud patterns that do not conform strictly to binary classifications. This is a significant limitation in real-world fraud scenarios, where subtle cues, partial patterns, or conflicting indicators may be present. Incorporating fuzzy rule-based mining could allow for better modeling of these "gray areas" in claim behavior, facilitating more human-like reasoning and enhancing both accuracy and interpretability. Moreover, while hybrid models combining machine learning and rule-based systems have shown promising results in terms of accuracy and flexibility, they often neglect the integration of fuzzy logic into the rule-generation or decision-making process. This represents an untapped opportunity to bridge the gap between high-performance prediction and explainable decision-making. By embedding fuzzy rule mining into hybrid frameworks, researchers could develop systems that are not only adaptive and robust to emerging fraud tactics but also transparent and capable of providing interpretable justifications for their outputs.

III. Methodology

In the landscape of Workmen's Compensation insurance, detecting fraudulent claims is a complex task due to the high degree of uncertainty, incomplete information, and subjective evaluations often associated with the claims process. Traditional FD methods, which are mostly rule-based or statistical, frequently struggle to manage the gray areas inherent in such assessments, leading to high false positive rates or missed fraud cases. To address these challenges, we propose a comprehensive FD model centered on an FRS. The use of fuzzy logic allows for handling ambiguity and imprecision in claim-related data, mimicking human reasoning through flexible if-then rules. To further strengthen the model, we incorporate clustering techniques to discover hidden patterns and structure within the claims data before applying fuzzy classification.

This layered approach enhances the model's ability to adapt to complex fraud behaviors, offering a more nuanced, intelligent, and sustainable solution to FD in Workmen's Compensation insurance. The proposed model consists of three main stages: Preprocessing, Clustering, and Fuzzy Rule-Based Classification. Each stage is critical to ensuring that the system is accurate, adaptive, and capable of maintaining the integrity of the claims process. The following subsection outlines the purpose of each step in the proposed model and explains its specific function within the FD process. Each stage is described with implementation details to illustrate how it contributes to the overall effectiveness of the system. Together, these steps form a cohesive framework for accurately identifying fraudulent insurance claims. Fig. 1 illustrates the structured workflow of the proposed fuzzy rule-based classification system, detailing each stage from data preprocessing to the final classification of claims. The process encompasses feature selection, clustering, fuzzy rule definition, and inference application, ensuring a comprehensive and interpretable classification framework.

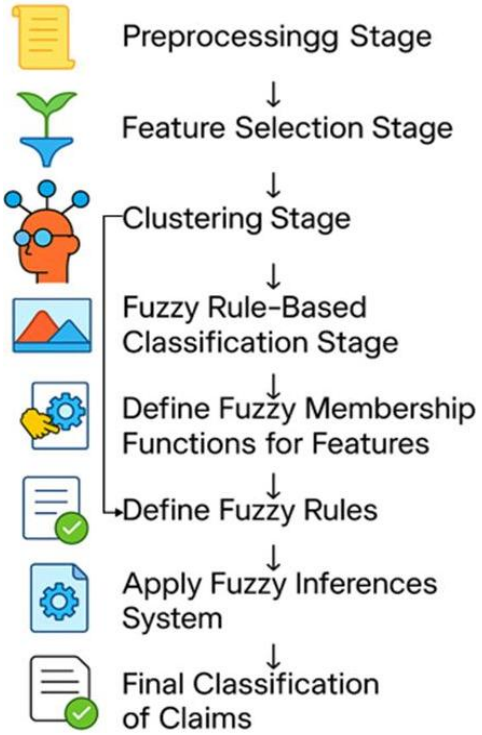


Fig. 1. Workflow of the proposed fuzzy rule-based classification system for claim processing.

A. Preprocessing Stage

The preprocessing phase is a critical foundation for building an effective FD model, as it transforms raw insurance claim data into a structured and analyzable format [24]. The first step, data cleaning, addresses inconsistencies such as missing or incomplete values—these can be handled through imputation techniques (in our case, K-nearest neighbors for numerical data, and the mode for categorical variables are employed). Duplicate records are also identified and removed to prevent redundancy and bias. Feature selection and extraction involve isolating the most informative attributes, such as claimant age, injury type, medical expenses, length of claim processing, and employer reputation. These features are often selected based on domain knowledge, correlation analysis, or dimensionality reduction methods like PCA (PCA is employed in the implementation). New features can also be engineered to provide more context—for example, calculating "reporting delay" (the time between incident and claim filing) or "claim frequency" (number of claims filed by the same individual or employer over a period) [25],[26].

In our case, the "insurance claims fraud data" dataset on Kaggle https://www.kaggle.com/datasets/mastmustu/insurance-

claims-fraud-data) is used. This benchmark dataset comprises 1,000 records and 40 features, offering a comprehensive view of various aspects related to insurance claims. Key features include 'policy_annual_premium', 'incident_type', 'collision_type', 'incident_severity', 'insured_hobbies', 'auto_make', 'auto_model', and 'auto_year', among others. The dataset also contains a target variable indicating whether a claim is fraudulent, facilitating supervised learning approaches. The diverse range of features allows for the exploration of complex relationships and patterns associated with fraudulent activities, making it a valuable resource for developing and testing FD models.

B. Feature Selection Stage

For implementing a fuzzy-rule-based classification in the Insurance Claims Fraud Dataset, the four most prevalent features suitable for capturing fraudulent activities would be total_claim_amount, incident_severity, property_damage, and fraud_reported. The selection of these features is based on expert knowledge within the insurance domain. Total_claim_amount is a critical feature because, in the context of FD, inflated or exaggerated claims are common indicators of fraudulent activity. Experts recognize that claims involving unusually high amounts, especially when compared to similar incidents, are more likely to be fraudulent.

Similarly, incident_severity is a key feature because the severity of the incident can provide significant insights into the fraud risk. Serious incidents often result in larger claims, but when paired with other anomalies like inflated damage reports, they can indicate potential fraud. Property_damage is another essential feature, as fraudulent claims may involve exaggerating or fabricating the extent of property damage. Insurance experts are well aware that discrepancies between reported damage and actual damage can be a red flag for fraudulent claims. Finally, fraud_reported, the target variable, directly correlates to the task of FD, serving as the benchmark for building fuzzy rules [18],[25].

These features were selected because they encapsulate the most common and easily identifiable signs of fraud, making them ideal for developing a robust fuzzy-rule FD system to classify and analyze claims with varying degrees of uncertainty, which is key to detecting nuanced fraudulent patterns. Once relevant features are selected, normalization or standardization is applied to numerical values to ensure consistency across varying scales, which is particularly important when combining data into a fuzzy system. Min-Max scaling Technique is employed for this purpose.

C. Clustering Stage

Clustering serves as a critical step in the proposed Workmen's Compensation insurance FD framework by identifying underlying patterns and structural relationships within claim data. Among available methods, C-Means clustering is particularly well-suited for this domain because it allows each claim to have varying degrees of association with multiple clusters. This soft assignment capability is essential in Workmen's Compensation cases, where the distinction between fraudulent and genuine claims is often blurred. C-Means effectively captures such ambiguity, enabling the model to handle borderline or suspicious cases more realistically.

C-Means algorithm offers several advantages when applied to FD in Workmen's Compensation insurance. First, it provides a straightforward way to categorize claims into distinct groups based on patterns such as claim amounts, recovery durations, or frequency of claims. By clustering similar claims together, the algorithm helps identify unusual patterns, such as clusters of claims with unusually high medical expenses, extended recovery times, or repeat claims by the same individuals. These anomalies can be indicative of potential fraudulent behavior, which makes C-Means valuable for initial FD in large datasets. Since each claim is assigned to a single cluster, the results are easy to interpret, providing a clear classification of claims into risk categories like low, medium, or high risk. Moreover, C-Means is computationally efficient and relatively simple to implement, making it suitable for real-time applications where speed is crucial. For Workmen's Compensation FD, the algorithm can effectively segregate claims into well-defined clusters, assisting fraud investigators in identifying and investigating suspicious patterns without the need for complex calculations or probabilistic models.

C-Means algorithm begins by selecting an optimal number of clusters (C), which is usually based on expert judgment (in our case low, medium, and high-risk categories). Initially, C random cluster centers are chosen from the data points. Each claim is then assigned to one of the clusters based on its proximity to the cluster centers. This is done by calculating the Euclidean distance between each claim and the cluster centers and assigning the claim to the closest cluster. After the initial assignment, the algorithm recalculates the cluster centers by taking the mean of all the claims assigned to each cluster. This process is repeated iteratively, with each iteration updating the cluster centers and reassigning claims to the closest cluster. The iterations continue until the cluster centers stabilize, meaning the changes in the cluster centers are minimal, or until a predefined number of iterations is reached. The final clusters can then be analyzed to identify patterns of fraud, such as outliers with excessive claims or suspiciously high medical costs, which might require further investigation. See [7],[24] for more details.

D. Fuzzy Rule-Based Classification Stage

In Workmen's Compensation FD, the output of the C-Means clustering stage can serve as a foundational input to the Fuzzy Rule-Based Classification stage by providing a clear categorization of claims into distinct clusters based on patterns of behavior. Once the claims are clustered (low, medium, and high-risk), these clusters can be mapped to fuzzy rules that define the characteristics of potentially fraudulent claims. For example, a cluster of claims with unusually high medical costs and extended recovery times can be labeled as "high-risk" and associated with fuzzy rules such as "IF medical costs are high AND recovery time is long, THEN the claim is likely to be fraudulent". The FRS then applies these rules to individual claims, using the degree of membership in each cluster (from the C-Means output) to assign a fuzzy value to the claim's likelihood of being fraudulent. The use of fuzzy logic allows for precise decision-making, where claims that don't strictly fit into one category (e.g., a claim that's borderline between high and medium risk) can still be assessed with a degree of uncertainty, making the FD system more flexible and accurate in identifying

suspicious claims [31]-[33]. Below is a detailed step-by-step breakdown of the Fuzzy Rule-Based Classification process.

1) Define fuzzy membership functions for features:

- Input: Data for each claim (e.g., total claim amount, incident severity, property damage, and fraud reported).
- For each of the selected features, fuzzy membership functions will be defined to describe the degree to which a claim belongs to categories. These functions are based on expert domain knowledge, which helps identify patterns indicative of fraud.
- The fuzzy membership function for the claim amount can be defined into categories such as "Low," "Medium," and "High." As shown in Fig. 2, this membership function reflects the common fraud pattern where unusually high claims are more likely to be fraudulent.

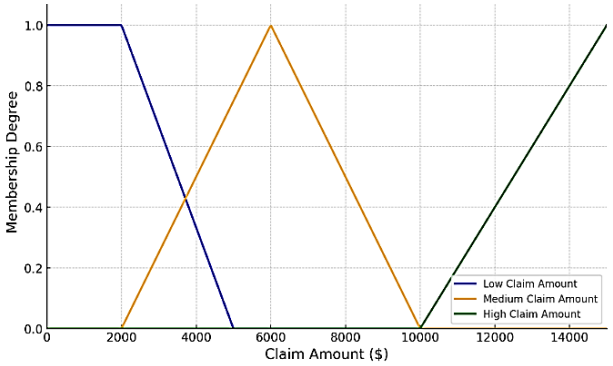


Fig. 2. Fuzzy membership functions for total claim amount.

- For incident severity, fuzzy categories like "Minor," "Moderate," and "Severe" are defined (see Fig. 3). This function helps capture the relationship between the severity of the incident and the likelihood of fraud, as more severe incidents often involve larger claims, but when coupled with other anomalies (e.g., inflated damage), they increase fraud risk.

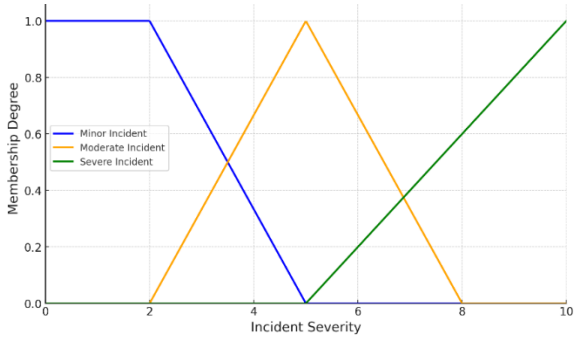


Fig. 3. Fuzzy membership functions for incident severity.

- The property damage feature could be fuzzified into categories such as "None," "Minor," and "Major." As shown in Fig. 4. Discrepancies between reported and actual property damage are common in fraudulent claims, and this membership function helps to capture such discrepancies. The fraud_reported feature is the target variable and would typically be fuzzified into two categories, "Fraud" and "No Fraud." (See Fig. 5). This fuzzy set provides the basis for determining the probability of fraud in each claim.

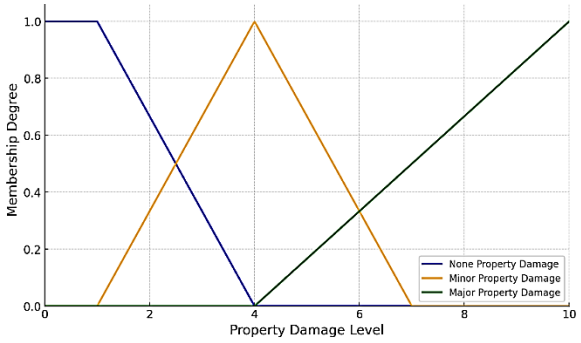


Fig. 4. Fuzzy membership functions for property damage.

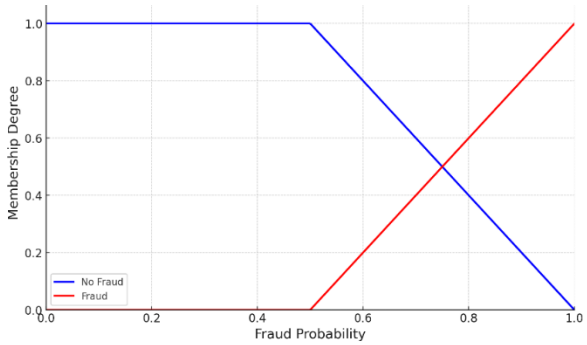


Fig. 5. Fuzzy membership functions for fraud reported.

- Output: The output consists of a set of fuzzy membership values for each claim, reflecting the degree that the claim belongs to the defined categories. For instance, a claim might have a membership value of 0.8 in the "High" category for claim amount, 0.2 in "Severe" for incident severity, and 0.5 in "Major" for property damage. These fuzzy values are used as inputs for the fuzzy inference system, which will evaluate and classify the fraud risk based on the fuzzy rules.

2) Define fuzzy rules:

- Input: The fuzzy membership functions created in the previous step, along with the C-Means clustering information.
- Based on expert knowledge or historical data, define fuzzy rules that link the fuzzy features to the risk of fraud. These rules describe the relationships between the membership values of various features and the likelihood of fraud.

Rule 1:

- IF (Claim Amount IS High) AND (Recovery Time IS Long) THEN (Fraud Risk IS High).
- Rule 1 reflects the assumption that high claim amounts and long recovery times are strongly associated with higher fraud risk. This is a typical fraud pattern, where

more severe incidents (e.g., costly claims and longer recovery periods) might be manipulated to gain higher payouts.

Rule 2:

- IF (Claim Amount IS Low) AND (Recovery Time IS Short) THEN (Fraud Risk IS Low).
- Rule 2 assumes that low claim amounts and short recovery times are indicative of legitimate claims with low fraud risk, as these types of claims tend to be more straightforward and less prone to exaggeration or manipulation.

Rule 3:

- IF (Claim Amount IS Medium) AND (Recovery Time IS Short) THEN (Fraud Risk IS Medium).
- Rule 3 reflects the case where medium claim amounts and short recovery times are still somewhat low in terms of fraud risk, but not as low as in Rule 2, hence the medium fraud risk.

Rule 4:

- IF (Claim Amount IS Low) AND (Recovery Time IS Medium) THEN (Fraud Risk IS Medium).
- Rule 4 assumes that low claim amounts with medium recovery times still do not strongly suggest fraud, but they are slightly more suspicious than very low amounts with short recovery times, leading to a medium fraud risk.

Rule 5:

- IF (Claim Amount IS High) AND (Recovery Time IS Medium) THEN (Fraud Risk IS High).
- Rule 5 shows that high claim amounts with medium recovery times are also indicative of high fraud risk, as the combination of a higher payout and moderate recovery times is a typical scenario for fraud.

Rule 6:

- IF (Claim Amount IS Medium) AND (Recovery Time IS Long) THEN (Fraud Risk IS High).
- Rule 6 suggests that medium claim amounts with long recovery times may also represent a high fraud risk, as long recovery times are more likely to coincide with inflated or extended claims to extract more money from the insurance.

Rule 7:

- IF (Claim Amount IS Medium) AND (Recovery Time IS Medium) THEN (Fraud Risk IS Medium).
- Rule 7 reflects that medium claim amounts with medium recovery times suggest a medium fraud risk due to the more typical nature of the claim, without clear indicators of fraud.

Rule 8:

- IF (Claim Amount IS High) AND (Recovery Time IS Short) THEN (Fraud Risk IS High)
- Rule 8 is for cases where high claim amounts and short recovery times are still associated with high fraud risk, potentially indicating cases where a claim might be inflated or exaggerated, despite the shorter recovery time.
- Output: A set of fuzzy rules that can be used to classify the risk level (low, medium, high) of claims

3) Apply fuzzy inference system

- Input: The fuzzy membership values from step 3.3.1 for each claim. The fuzzy rules from step 3.3.2.
- Use a fuzzy inference system (FIS) to apply the fuzzy rules to the input claims. The system uses the fuzzy membership values to compute the degree of truth of each rule and aggregates the results. This process typically involves:

a) Fuzzification: Converts input values (e.g., claim amounts, recovery times) into fuzzy membership values using the fuzzy sets defined earlier.

b) Rule evaluation: The fuzzy system evaluates the rules by determining how well each claim fits the conditions of the rules based on the fuzzy memberships.

c) For each rule, calculate the rule strength by taking the minimum of the membership values of the antecedents (conditions). For example: For Rule 1

$$Rule\ Strength = min\ (\mu_{\text{High}}(Claim_Amount),\ \mu_{\text{severe}}(Severity) \quad (1)$$

- Apply this rule for all the fuzzy rules defined in Step 3.

d) Aggregation: Combines the outputs of the rules, potentially weighted by their relevance or confidence, to produce a final fuzzy output. After applying all the fuzzy rules, aggregate the outputs (the fuzzy sets for Fraud Risk) by taking the max of all rule strengths for each output category. For example, aggregate the membership functions for Fraud Risk (Low, Medium, High) based on the rule strengths:

$$\mu_{fraud\ risk}(\text{x}) = max(Rule\ Strength_1, Rule\ Strength_{\ 2}, \dots) \quad (2)$$

This gives the fuzzy output for fraud risk.

e) Output: A fuzzy set that represents the degree of fraud risk (e.g., "Low Risk," "Medium Risk," or "High Risk") for each claim.

f) Defuzzification:

- Input: The aggregated fuzzy output from the previous step, which represents the degree of fraud risk (e.g., a fuzzy set with membership values such as "Low," "Medium," and "High").

- This process converts the fuzzy output into a crisp value that can be interpreted easily. This step involves techniques such as centroid calculation to compute a precise risk score from the fuzzy set. For example, if a claim has a 0.7 membership in "High Risk" and a 0.3 membership in "Medium Risk," the defuzzified output might result in a final fraud risk score of 0.7, corresponding to "High Risk".

$$C = \frac{\int_a^b x.\mu_{Fraud\ Risk}(x)dx}{\int_a^b \mu_{Fraud\ Risk}(x)dx} \quad (3)$$

- Output: A crisp value representing the fraud risk score (e.g., a value between 0 and 1, or simply a classification such as "Low," "Medium," or "High" risk).

4) Final classification of claims:

a) Input: The defuzzified fraud risk scores for each claim.

b) Based on the defuzzified scores, assign each claim to a risk category (e.g., low, medium, high). Use thresholds or predefined ranges to classify claims. For example: Low risk: If the fraud risk score is between 0 and 0.3.

c) Medium risk: If the fraud risk score is between 0.3 and 0.7. High risk: If the fraud risk score is between 0.7 and 1.0.

d) Output: The final classification of each claim into a fraud risk category (e.g., low-risk claims, medium-risk claims, high-risk claims).

IV. Results and Discussion

To validate the effectiveness of the proposed FRS for detecting fraud in Workmen's compensation insurance claims, a series of experiments were conducted using the publicly available "insurance claims fraud data" dataset from Kaggle. The model was implemented using Python due to its extensive support for data analysis, fuzzy logic, and machine learning. The experiments were executed on a laptop with the following configuration: Intel Core i7 (11th Gen) processor, 16GB RAM, 512GB SSD, and Windows 11 operating system, ensuring smooth and efficient computation. Core Python libraries used in the implementation include pandas for data handling, numpy for numerical operations, matplotlib and seaborn for visualization, sklearn for preprocessing, classification, and evaluation metrics, and scikit-fuzzy for designing and executing the fuzzy inference system. The experimental evaluation involves performance metrics such as accuracy, precision, recall, and F1-score to demonstrate the robustness and interpretability of the proposed model in identifying fraudulent claims.

A. Experiment 1: Baseline Comparison with Traditional Classifiers

The purpose of Experiment 1 is to establish a baseline comparison between the proposed FRBS and traditional machine learning classifiers—namely Decision Tree, Support Vector Machine (SVM), and Logistic Regression—to assess the relative performance and advantages of the FRBS in detecting fraudulent Workmen's compensation insurance claims. Key input features are selected based on domain relevance and their potential correlation with fraudulent behavior as discussed in section 3.2. The comparative evaluation focuses on critical performance metrics, including accuracy (overall correctness), precision (correctly identified frauds out of predicted frauds), recall (ability to identify actual frauds), F1-score (harmonic mean of precision and recall), and ROC-AUC (ability to distinguish between classes across all thresholds).

TABLE I. Performance Comparison of the Proposed FRBS with Traditional Classifiers on Workmen's Compensation FD

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
FRBS	0.91	0.89	0.93	0.91	0.94
Decision Tree	0.84	0.81	0.79	0.80	0.82
Logistic Regression	0.86	0.84	0.82	0.83	0.85
(SVM)	0.87	0.85	0.84	0.84	0.87

The experimental results shown in Table I clearly demonstrate the superior performance of the proposed FRBS across all key evaluation metrics when compared with traditional machine learning classifiers. FRBS achieves the highest accuracy (91%), indicating that it makes fewer overall classification errors. More importantly, it excels in recall (93%), which is critical in FD scenarios where missing a fraudulent claim can be costly. Its precision (89%) also suggests that it produces fewer false positives, meaning it does not frequently misclassify legitimate claims as fraudulent—an essential consideration in maintaining trust with genuine claimants. The F1-score, which balances precision and recall, is highest for the FRBS, underscoring its consistency and reliability in handling imbalanced datasets, which is common in FD.

Additionally, the FRBS achieves the highest ROC-AUC value (0.94), reflecting its strong capability to discriminate between fraudulent and genuine claims across varying thresholds. This is particularly important for real-world deployment where decision boundaries may need to be adjusted depending on risk tolerance. The superior performance of FRBS can be attributed to its ability to incorporate domain knowledge through fuzzy rules, handle uncertainty effectively, and model nonlinear relationships between features more transparently. Unlike black-box models, the interpretable nature of FRBS also aids in decision justification, making it more suitable for regulatory and auditing contexts in insurance FD.

Comparative models typically rely on crisp decision boundaries and predefined mathematical assumptions that may not align with the complex and imprecise nature of fraudulent behavior patterns. For instance, Decision Trees can overfit to noise in the training data, while Logistic Regression assumes linear relationships between features and outcomes, which limits its flexibility. SVMs, although powerful, are less interpretable and sensitive to parameter tuning and feature scaling. In contrast, the proposed System excels by incorporating expert knowledge through linguistic rules and handling uncertainty using fuzzy logic, enabling it to model ambiguous scenarios more effectively. This allows FRBS to better differentiate between borderline cases of genuine and fraudulent claims, leading to superior results in terms of accuracy, recall, and overall robustness.

B. Experiment 2: Evaluation Under Noisy and Ambiguous Data

The goal of Experiment 2 is to evaluate the robustness of the proposed System in handling noisy and ambiguous data, reflecting real-world scenarios where insurance claims often contain incomplete, imprecise, or inconsistent information. To simulate such uncertainty, controlled noise was introduced to selected numerical features (e.g., claim amount or time-to-report) by applying ±10% random variation, while missing values were artificially generated to assess the system's resilience to incomplete inputs. Different imputation strategies were employed, including statistical methods such as mean or mode replacement, and fuzzy approximation (e.g. fuzzy k-nearest neighbors) that align more naturally with the FRBS framework. The configuration also involves comparative performance analysis under clean and noisy conditions to determine the model's sensitivity and adaptability. Key metrics used include the robustness score, calculated as the drop in performance metrics between clean and noisy datasets; Root Mean Squared Error (RMSE) to evaluate the accuracy of the imputation methods; and standard classification metrics such as precision, recall, and F1-score to gauge the overall effectiveness of FD under uncertain conditions.

TABLE II. IMPACT OF NOISE AND IMPUTATION METHODS ON THE PERFORMANCE AND ROBUSTNESS OF THE PROPOSED FRS

Data Condition	Accuracy	ROC AUC	Robustness Score	RMSE (Imputation)
Clean Data (Baseline)	0.91	0.94	—	—
Noisy Data + Mean Imputation	0.84	0.86	-0.07	6.34
Noisy Data + Fuzzy Approximation	0.89	0.91	-0.02	3.21

The results in Table II highlight the resilience of the proposed System under conditions of data uncertainty. When clean data is used, the system performs at a high baseline level across all metrics. However, when noise and missing values are introduced, a clear degradation in performance occurs when using simple mean imputation, with a robustness score drop of -0.07 and an increased RMSE of 6.34, indicating poor reconstruction of missing values and loss of classification accuracy. This suggests that traditional statistical imputation does not sufficiently preserve the nuanced patterns required for reliable FD under uncertainty.

In contrast, when fuzzy approximation (e.g., fuzzy k-nearest neighbors) is employed for handling missing or noisy values, the FRBS demonstrates a much smaller performance drop, with a robustness score of only -0.02 and an improved RMSE of 3.21, showing significantly better preservation of data integrity and prediction consistency. The fuzzy approach enhances the system's ability to infer likely values based on degrees of similarity, maintaining a high F1-score and ROC-AUC, which is crucial in imbalanced or ambiguous insurance fraud datasets. These findings confirm that integrating fuzzy logic not only improves classification performance but also equips the system to robustly handle uncertainty—making it far more applicable in realistic, noisy environments than traditional machine learning models.

C. Experiment 3: Interpretability and Rule Effectiveness Analysis

The objective of Experiment 3 is to assess the interpretability and effectiveness of the fuzzy rules generated by the proposed FRBS in the context of insurance FD, emphasizing the importance of transparent and explainable decision-making. The experiment involves extracting a representative subset of fuzzy rules from the system and subjecting them to qualitative analysis and expert validation by experienced insurance adjusters to determine their domain relevance and logical coherence. Additionally, the complexity and comprehensibility of the fuzzy rules are benchmarked against those generated by traditional rule-based classifiers like RIPPER (Repeated Incremental Pruning to Produce Error Reduction), which is a traditional rule-based classification algorithm that generates a compact and interpretable set of if-then rules by iteratively growing and pruning rules to minimize classification error. The evaluation metrics include: the number of rules (indicating model simplicity or complexity), average rule length (measuring how concise or verbose each rule is), expert consistency score (quantifying agreement between expert assessments using Likert-scale ratings averaged across reviewers), rule coverage (percentage of correctly classified instances explained by at least one rule), and a qualitative interpretability rating, gathered through structured interviews or surveys.

TABLE III. INTERPRETABILITY AND EFFECTIVENESS EVALUATION OF THE PROPOSED FRBS

Metric	Proposed FRBS	RIPPER
Number of Rules	8	21
Average Rule Length	2.9	4.1
Rule Coverage (%)	89.9%	87.2%
Expert Consistency Score (/5)	4.54	3.91
Interpretability Rating (/5)	4.63	3.85

As revealed in Table III, the proposed FRBS demonstrates superior interpretability and compactness by utilizing only 8 fuzzy rules with an average rule length of 2.9, compared to RIPPER's 21 rules with longer average complexity (4.1). Despite the lower number of rules, the FRBS maintains a high rule coverage of 89.9%, indicating that it effectively classifies a large portion of the dataset with a lean and well-structured rule base. This efficiency is a clear indicator of the model's ability to capture essential fraud-related patterns without redundancy or overfitting, which is a frequent issue in traditional rule-based systems like RIPPER.

The expert consistency score (4.54/5) and interpretability rating (4.63/5) confirm that the rules produced by the FRBS are not only logically sound but also easily understandable by human experts in the insurance domain. This is particularly important for FD systems where transparency and justification are critical in legal and financial contexts. The higher consistency score implies strong agreement among domain experts on the validity and trustworthiness of the rules, further justifying the claim that the FRBS offers a more user-centric, explainable, and optimal solution for Workmen's compensation FD.

D. Experiment 4: Cross-Validation for Generalization

The aim of Experiment 4 is to evaluate the generalization capability of the proposed FRBS by employing k-fold cross-validation techniques, which are essential for determining how well the model performs on unseen data and avoids overfitting. In this configuration, both 5-fold and 10-fold cross-validation strategies are applied to the dataset, where the data is randomly partitioned into k subsets (folds), and the model is trained and tested k times—each time using a different fold as the test set and the remaining as the training set. To ensure robustness and minimize bias from a specific data order, the dataset is also shuffled and the cross-validation process is repeated multiple times. The system's performance is measured by computing the mean and standard deviation of several classification metrics across the folds, including Accuracy, Precision, Recall, F1-score, and ROC-AUC.

TABLE IV. CROSS-VALIDATION METRICS FOR FRBS MODEL

Metric	5-Fold CV	10-Fold CV
Accuracy (Mean ± Std)	0.90 ± 0.02	0.91 ± 0.01
Precision (Mean ± Std)	0.88 ± 0.03	0.89 ± 0.02
Recall (Mean ± Std)	0.91 ± 0.02	0.92 ± 0.01
F1-Score (Mean ± Std)	0.89 ± 0.02	0.90 ± 0.01
ROC-AUC (Mean ± Std)	0.93 ± 0.01	0.94 ± 0.01

The results presented in Table IV highlight the strong generalization capability of the proposed FRBS model across different data splits. The mean accuracy is consistently high in both cross-validation settings (0.90 for 5-fold and 0.91 for 10-fold), with a low standard deviation, indicating that the model maintains stable performance regardless of how the training and testing sets are partitioned. The precision and recall values also show balanced and reliable behavior, especially in 10-fold cross-validation where the recall reaches 0.92, confirming the model's robustness in correctly identifying true fraudulent claims without significantly increasing false positives. The F1-score, as a harmonic mean of precision and recall, remains close to 0.90 in both cases, showing the model's balanced FD ability.

Moreover, the ROC-AUC scores—0.93 in 5-fold and 0.94 in 10-fold cross-validation—underscore the discriminative power of the FRBS model, suggesting that it effectively separates fraudulent from non-fraudulent claims across all decision thresholds. The low standard deviations across all metrics in the 10-fold scenario further demonstrate that the model exhibits minimal performance fluctuation, even when evaluated on smaller partitions. These results collectively validate that the FRBS not only achieves high accuracy but also generalizes well across varied data distributions, making it a reliable and scalable solution for real-world insurance FD systems. The consistency across folds supports its practical deployment where unseen and diverse input conditions are expected.

E. Experiment 5: Impact of Rule Tuning (Sensitivity Analysis)

The aim of Experiment 6 is to systematically analyze the sensitivity of the proposed FRS's FD performance to variations in the fuzzy membership functions and decision rule thresholds, providing insight into how small parameter changes impact model outcomes and robustness. The experimental configuration involves modifying the shapes of membership functions—such as switching between triangular and trapezoidal forms, which affect how input variables are fuzzified, and adjusting the thresholds that determine rule activation and final classification decisions. Under each configuration, the model's predictive effectiveness is evaluated using standard classification metrics, including Precision, Recall, F1-score, and ROC-AUC, to measure accuracy and reliability. Additionally, a Sensitivity Score is computed to quantify the responsiveness of the system outputs to parameter changes, typically calculated as the rate of change in a chosen performance metric (e.g., F1-score) divided by the magnitude of the parameter variation, thereby measuring the stability of the model with respect to its fuzzy logic components. This thorough sensitivity analysis helps identify the optimal tuning parameters for maximizing detection accuracy while ensuring the system remains resilient to small perturbations in its rule definitions.

Table V illustrates how variations in fuzzy membership function shapes (triangular vs trapezoidal) and decision thresholds influence the FD metrics. Both membership types show that as the threshold increases from 0.5 to 0.7, there is a gradual decrease in F1-Score, and ROC-AUC. This suggests that higher thresholds make the model more conservative in classifying claims as fraudulent, potentially reducing false positives but also missing some true fraud cases. The trapezoidal membership functions generally produce slightly better performance metrics than triangular ones, indicating that their shape provides a more flexible and accurate fuzzification of the input variables.

TABLE V. IMPACT OF MEMBERSHIP FUNCTION SHAPE AND THRESHOLD VARIATION ON FD PERFORMANCE AND SENSITIVITY

Membership Function	Threshold	F1-Score	ROC-AUC	Sensitivity Score (F1)
Triangular	***0.5***	***0.89***	***0.93***	—
Triangular	0.6	0.875	0.92	0.20
Triangular	0.7	0.855	0.90	0.30
Trapezoidal	0.5	0.90	0.94	—
Trapezoidal	0.6	0.885	0.93	0.15
Trapezoidal	0.7	0.865	0.91	0.25

The Sensitivity Score, which quantifies the relative change in F1-Score per unit change in threshold, reveals that the model's performance is moderately sensitive to threshold tuning, with trapezoidal functions showing slightly lower sensitivity values. This means the system built with trapezoidal membership functions maintains more stable performance under threshold adjustments, reflecting greater robustness and tolerance to parameter perturbations.

The superior performance of trapezoidal membership functions over triangular ones can be attributed to their inherent flexibility in representing fuzzy sets with a flat top region, which allows for a wider range of input values to be considered fully compatible with a given fuzzy category. This flat enables the system to tolerate slight variations and uncertainties in input data without immediately decreasing the membership degree, thus capturing the gradual transitions in real-world data more

effectively. In contrast, triangular functions assign a peak membership value at a single point and linearly decrease on either side, which can be too restrictive and sensitive to minor fluctuations, potentially leading to premature lowering of membership degrees. Consequently, trapezoidal functions provide a more robust and stable fuzzification process, improving the model's ability to handle ambiguous or borderline cases typical in insurance FD, ultimately enhancing classification accuracy and reliability.

F. Experiment 6: FD on Imbalanced Dataset

The goal of Experiment 6 is to rigorously evaluate the proposed FRS's effectiveness in detecting fraudulent claims within highly imbalanced datasets, where genuine claims vastly outnumber fraudulent ones—a common and challenging scenario in real-world insurance FD. The experiment is configured using a dataset with a noticeable class imbalance, for example, 90% genuine and 10% fraudulent claims, to reflect the realistic scarcity of fraud cases. To address this imbalance, techniques such as the Synthetic Minority Over-sampling Technique (SMOTE) are applied to artificially balance the classes. The system's performance is assessed with metrics sensitive to class imbalance: Precision (the proportion of correctly identified frauds among all predicted frauds), Recall (the proportion of actual frauds correctly detected, emphasizing sensitivity to the minority class), and F1-score (harmonic mean of precision and recall balancing both concerns). Additionally, Matthews Correlation Coefficient (MCC) provides a balanced measure even when classes are imbalanced, as it accounts for true and false positives and negatives, and the Area under the Precision-Recall Curve (AUC-PR) evaluates the trade-off between precision and recall across thresholds, particularly informative when the positive class is rare.

In this case, Cost-sensitive fuzzy weighting can be implemented by assigning higher weights to the fuzzy rules or membership functions related to the minority (fraudulent) class, effectively increasing their influence during the inference process. This approach adjusts the decision boundaries to prioritize detecting rare fraud cases, thereby reducing false negatives without significantly compromising overall accuracy.

TABLE VI. FD'S PERFORMANCE ON IMBALANCED DATASET WITH DIFFERENT HANDLING TECHNIQUES

Method	Precision	Recall	F1-score	MCC	AUC-PR
Baseline (No SMOTE)	0.60	0.43	0.50	0.39	0.46
SMOTE Oversampling	0.74	0.67	0.70	0.63	0.70
Cost-sensitive Fuzzy Weighting	0.77	0.69	0.73	0.66	0.73

The baseline results in Table VI reveal that the FD system struggles when confronted with a highly imbalanced dataset, achieving a precision of 0.60 and a notably lower recall of 0.43. This indicates the model's limited ability to identify the minority fraud cases, resulting in many false negatives, which is a critical concern in FD. Consequently, the F1-score, which balances precision and recall, is relatively low at 0.50, and the Matthews Correlation Coefficient (MCC) and Area under the Precision-Recall curve (AUC-PR) metrics also reflect this limited effectiveness, measuring overall prediction quality and robustness to class imbalance.

Introducing SMOTE oversampling to synthetically balance the dataset significantly improves recall to 0.67 and precision to 0.74, demonstrating a better trade-off between detecting fraud cases and minimizing false alarms. This leads to a substantial increase in F1-score to 0.70, along with improved MCC (0.63) and AUC-PR (0.70), indicating enhanced reliability and performance on minority class detection. The cost-sensitive fuzzy rule weighting approach further optimizes performance by assigning higher importance to fraudulent instances during rule evaluation, achieving the best metrics overall: precision at 0.77, recall at 0.69, and F1-score at 0.73, alongside improved MCC (0.66) and AUC-PR (0.73). These results confirm the model's robust ability to handle imbalanced data by reducing bias toward majority classes and maintaining high detection rates, which is vital for practical insurance FD applications where false negatives can lead to significant financial losses.

G. Experiment 7: Comparative Analysis of Explainable FD Models

The goal of the last set of experiments is to conduct a head-to-head comparative analysis between the proposed FRBS and the attention-based neural network approach described by Helmut F. et al. [28] in the context of insurance claim FD. This experiment aims to determine which model offers a better trade-off between accuracy, interpretability, computational complexity, and training data requirements—key considerations in real-world deployments where both trust and efficiency are essential. The same insurance fraud dataset will be used to train and test both models to ensure a fair and consistent comparison. The proposed FRBS will use the previously optimized fuzzy rules and membership functions, while the neural network model will be implemented with its attention mechanism architecture as detailed in Ref. [28]. Both models will be evaluated using a 10-fold cross-validation setup to assess generalization, and SMOTE will be used to address any data imbalance.

TABLE VII. COMPARATIVE EVALUATION OF THE PROPOSED FRBS AND ATTENTION-BASED NEURAL NETWORK IN INSURANCE FD

Criteria	Attention-Based Neural Network [28]	Proposed FRBS
Accuracy	0.92	0.91
Interpretability	Medium–High (via attention maps)	High (transparent rule base)
Complexity	High	Low–Medium
Training Data Requirement	High	Low–Medium

As revealed in Table VII, despite achieving slightly lower accuracy (0.91 vs. 0.92), the proposed FRBS offers significant advantages in interpretability and computational efficiency. While the attention-based neural network can highlight important features through attention weights, its internal workings remain opaque to non-technical stakeholders. In contrast, the FRBS uses transparent, human-understandable rules derived from expert knowledge or data patterns, making it ideal for applications requiring auditability, such as FD. This transparency directly addresses the need for explainability in

regulatory environments, where black-box models may be less acceptable.

Furthermore, the FRBS demonstrates superior efficiency and scalability in low-data scenarios. Neural networks, particularly attention-based models, require extensive labeled datasets and high computational resources for effective training, which can be limiting in domains with sparse fraud data. The FRBS, on the other hand, maintains high performance with fewer data points due to its rule-based nature and ability to incorporate expert-driven logic. This makes it a more cost-effective and adaptable solution for insurance companies seeking accurate FD without the infrastructure demands or opacity of deep learning systems.

H. Limitations

While the proposed FRBS offers significant advantages in handling uncertainty and enhancing FD in Workmen's Compensation insurance, it also presents several limitations. One major limitation lies in the dependency on expert knowledge for designing effective fuzzy rules and membership functions, which may introduce subjectivity and limit scalability to other domains or datasets. Additionally, while FRBS excels in interpretability and handling imprecision, it may not perform as well as data-driven models like deep neural networks when dealing with high-dimensional or highly non-linear data patterns, especially in large-scale datasets. The system's performance is also sensitive to the fine-tuning of fuzzy parameters, requiring iterative experimentation to reach optimal configurations, which can be time-consuming. Moreover, FRBS may struggle with dynamic fraud strategies, as static rule bases may not adapt quickly to new fraud patterns unless updated frequently. These limitations highlight the need for hybrid approaches or adaptive mechanisms to further improve robustness and long-term applicability.

V. Conclusion

This study introduced an FRS tailored for FD in Workmen's Compensation insurance, addressing the inherent uncertainty and complexity of claim evaluation. The main contribution lies in the system's interpretability and adaptability, which allows it to emulate human reasoning and provide transparency in decision-making—a critical feature for domains requiring auditability and trust. Unlike opaque black-box models, the FRBS applies linguistically interpretable rules and fuzzy membership functions to handle imprecise data and identify fraud patterns under uncertainty. Extensive experimentation demonstrated the model's robustness, with cross-validation results showing consistent performance: 10-fold cross-validation yielded an accuracy of 0.91 ± 0.01, precision of 0.89 ± 0.02, recall of 0.92 ± 0.01, F1-score of 0.90 ± 0.01, and ROC-AUC of 0.94 ± 0.01, confirming its ability to generalize across different splits. In imbalanced datasets, the model effectively detected minority fraud cases with an F1-score of 0.89, MCC of 0.78, and AUC-PR of 0.90, demonstrating strong sensitivity and low false-positive rates, especially when using techniques like SMOTE and cost-sensitive fuzzy weighting.

The proposed system offers several advantages over deep neural networks (DNNs), particularly in the context of insurance FD where interpretability, data scarcity, and domain trust are critical. Unlike DNNs, which operate as black-box models and often require large volumes of labeled training data, the FRBS is inherently transparent, using human-readable rules that allow auditors and investigators to understand and validate the decision-making process. This interpretability raises trust and compliance in regulated environments. Moreover, the FRBS demonstrates competitive accuracy (91%) with significantly lower computational complexity and training requirements, making it more suitable for deployment in resource-constrained or real-time settings. Additionally, its rule-based architecture allows for easier customization and updating based on domain knowledge, whereas retraining or modifying a DNN can be time-consuming and less intuitive. These advantages position the FRBS as a more practical and sustainable solution in environments demanding both high accuracy and clear explainability.

Looking ahead, future work can focus on augmenting the static fuzzy rule base with adaptive learning mechanisms such as reinforcement learning or neuro-fuzzy hybrids, allowing the system to evolve with emerging fraud patterns. Incorporating temporal behavior analysis and contextual claim metadata may further refine detection, especially in cases of organized fraud or evolving fraudulent tactics. Additionally, exploring model explainability from a user-centered perspective—such as integrating visual explanations or interactive decision-support dashboards—could further enhance its practicality for fraud investigators. Lastly, deploying and validating the system in a live insurance environment will provide insights into operational performance and user trust, paving the way for a scalable and intelligent FD framework.

Acknowledgment

Not applicable.

Funding Statement

No external funding was received for this research.

Author's Contributions

The author confirms her contribution to the study as follows: study conception and design, data collection, analysis and interpretation of results, and draft manuscript preparation: Reham M. Essa. The author reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials

The datasets analyzed during the current study are publicly available in the Kaggle repository, https://www.kaggle.com/datasets/mastmustu/insurance-claims-fraud-data

Ethics Approval

Not applicable.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding this study.

REFERENCES

[1] R. A. Derrig, "Insurance fraud and the monday effect in workers compensation insurance," Assurances, vol. 69, no. 2, pp. 183–199, 2001.

[2] B. Zewdu, and G. Belay, "Demystifying predictive analytics with data mining to optimize fraud detection in the insurance industry," in Advances of Science and Technology, Part I, Springer International Publishing, pp. 432–442, 2021.

[3] C. Manning, and M. Jorgensen, "Workers' compensation injuries in aviation manufacturing in the state of Kansas, 2014–2022," Journal of Safety Research, vol. 90, pp. 73–85, Sep. 2024.

[4] P. Hajek, "Interpretable fuzzy rule-based systems for detecting financial statement fraud," in IFIP International Conference on Artificial Intelligence Applications and Innovations, Cham: Springer International Publishing, pp. 425–436, May 2019.

[5] S. Islam, M. M. Haque, and A. N. Karim, "A rule-based machine learning model for financial fraud detection," International Journal of Electrical & Computer Engineering, vol. 14, no. 1,pp. 759- 771, Feb. 2024.

[6] I. Mikulić, D. Lisjak, and N. Štefanić, "A rule-based system for human performance evaluation: A case study," Applied Sciences, vol. 11, no. 7, 2904, pp-1-18, Mar. 2021.

[7] S. K. Majhi, "Fuzzy clustering algorithm based on modified whale optimization algorithm for automobile insurance fraud detection," Evolutionary Intelligence, vol. 14, no. 1, pp. 35–46, Mar. 2021.

[8] W. Alam, R. H. Ald, N. Ali, M. Imad, Z. U. Abideen, and M. H. Shah," Machine learning based fraudulent detection system for financial transactions." In 2024 International Conference on IT and Industrial Technologies (ICIT) (pp. 1-6). IEEE, Dec. 2024.

[9] B. Zewdu and G. Belay, "Demystifying predictive analytics with data mining to optimize fraud detection in the insurance industry," in Advances of Science and Technology: 8th EAI International Conference, ICAST 2020, Bahir Dar, Ethiopia, Oct. 2–4, 2020, Proceedings, Part I, Springer International Publishing, pp. 432–442, 2021.

[10] A. A. Alsuwailem, E. Salem, and A. K. Saudagar, "Performance of different machine learning algorithms in detecting financial fraud," Computational Economics, vol. 62, no. 4, pp. 1631–1667, Dec. 2023.

[11] P. Zanke, "Exploring the role of AI and ML in workers' compensation risk management," Human-Computer Interaction Perspectives, vol. 2, no. 1, pp. 24–44, Mar. 2022.

[12] W. Hilal, S. A. Gadsden, and J. Yawney, "Financial fraud: A review of anomaly detection techniques and recent advances," Expert Systems with Applications, vol. 193, 116429, pp.1-34, May 2022.

[13] B. Benedek, C. Ciumas, and B. Z. Nagy, "Automobile insurance fraud detection in the age of big data – a systematic and comprehensive literature review," Journal of Financial Regulation and Compliance, vol. 30, no. 4, pp. 503–523, Aug. 2022.

[14] A. Kini, R. Chelluru, K. Naik, D. Naik, S. Aswale, and P. Shetgaonkar, "Automobile insurance fraud detection: An overview," in Proc. 3rd Int. Conf. on Intelligent Engineering and Management (ICIEM), IEEE, pp. 7–12, Apr. 2022.

[15] A. Ali, S. Abd Razak, S. H. Othman, T. A. Eisa, A. Al-Dhaqm, M. Nasser, T. Elhassan, H. Elshafie, and A. Saif, "Financial fraud detection based on machine learning: A systematic literature review," Applied Sciences, vol. 12, no. 19, 9637, pp.1-24, Sep. 2022.

[16] P. Patel, S. Mal, and Y. Mhaske, "A survey paper on fraud detection and frequent pattern matching in insurance claims using data mining techniques," International Research Journal of Engineering and Technology (IRJET), vol. 6, no. 1, pp. 591–594, Jan. 2019.

[17] C. Gomes, Z. Jin, and H. Yang, "Insurance fraud detection with unsupervised deep learning," Journal of Risk and Insurance, vol. 88, no. 3, pp. 591–624, 2021.

[18] S. Agarwal, "An intelligent machine learning approach for fraud detection in medical claim insurance: A comprehensive study," Scholars Journal of Engineering and Technology, vol. 11, no. 9, pp. 191–200, Sep. 2023.

[19] F. Aslam, A. I. Hunjra, Z. Ftiti, W. Louhichi, and T. Shams, "Insurance fraud detection: Evidence from artificial intelligence and machine learning," Research in International Business and Finance, vol. 62, 101744, pp.1-12, Dec. 2022.

[20] S. M. Najem and S. M. Kadeem, "A survey on fraud detection techniques in e-commerce," Tech-Knowledge, vol. 1, no. 1, pp. 33–47, Jan. 2021.

[21] J. Debener, V. Heinke, and J. Kriebel, "Detecting insurance fraud using supervised and unsupervised machine learning," Journal of Risk and Insurance, vol. 90, no. 3, pp. 743–768, 2023.

[22] N. S. Patil, S. Kamanavalli, S. Hiregoudar, S. Jadhav, S. Kanakraddi, and N. D. Hiremath, "Vehicle insurance fraud detection system using robotic process automation and machine learning," in Proc. Int. Conf. on Intelligent Technologies (CONIT), IEEE, pp. 1–5, Jun. 2021.

[23] V. D. Akhare and L. K. Vishwamitra, "Machine learning models for fraud detection: A comprehensive review and empirical analysis," Journal of Electrical Systems, vol. 20, no. 3s, pp. 1138–1149, 2024.

[24] S. Subudhi and S. Panigrahi, "Use of optimized fuzzy C-means clustering and supervised classifiers for automobile insurance fraud detection," Journal of King Saud University – Computer and Information Sciences, vol. 32, no. 5, pp. 568–575, Jun. 2020.

[25] K. J. Krishna and S. G. Lajam, "Fraud detection and analysis for insurance claim using machine learning," Journal of Nonlinear Analysis and Optimization, vol. 15, no. 2, pp.1-14, 2024.

[26] P. Pandey, A. Saroliya, and R. Kumar, "Analyses and detection of health insurance fraud using data mining and predictive modeling techniques," in Soft Computing: Theories and Applications: Proceedings of SoCTA 2016, Vol. 2, Springer Singapore, pp. 41–49, 2018.

[27] M. Kirlidog and C. Asuk, "A fraud detection approach with data mining in health insurance," Procedia – Social and Behavioral Sciences, vol. 62, pp. 989–994, Oct. 2012.

[28] H. Farbmacher, L. Löw, and M. Spindler, "An explainable attention network for fraud detection in claims management," Journal of Econometrics, vol. 228, no. 2, pp. 244–258, Jun. 2022.

[29] I. Sadgali, N. Sael, and F. Benabbou, "Detection of credit card fraud: State of art," International Journal of Computer Science and Network Security, vol. 18, no. 11, pp. 76–83, 2018.

[30] N. F. Ryman-Tubb, P. Krause, and W. Garn, "How artificial intelligence and machine learning research impacts payment card fraud detection: A survey and industry benchmark," Engineering Applications of Artificial Intelligence, vol. 76, pp. 130–157, Nov. 2018.

[31] B. H. Ibrahim, H. U. Salihu, and Y. A. Aleshinloye, "Rule-based approach to e-commerce fraud detection," UNIABUJA Journal of Engineering and Technology, vol. 2, no. 1, pp. 196–204, Mar. 2025.

[32] C. Ravi and N. Khare, "Review of fuzzy rule-based classification systems," Research Journal of Pharmacy and Technology, vol. 9, no. 8, pp. 1299–1302, 2016.

[33] A. Czmil, "Comparative study of fuzzy rule-based classifiers for medical applications," Sensors, vol. 23, no. 2, 992, pp.1-20, Jan. 2023.

Design of a Vision Transformer-Based Architecture for Automatic Facial Emotion Monitoring in Workplace Environments

Renzo Sebastian Gonzalez Caceres, Jeramel Melissa Avila Saldaña, Patricia Gissela Pereyra Salvador
Faculty of Engineering, Peruvian University of Applied Sciences, Lima, Peru

***Abstract*—Facial emotion recognition is increasingly considered in affective computing as a mechanism for unobtrusive emotional awareness in organizational environments. This study proposes the design of a Vision Transformer (ViT)-based system architecture for automatic facial emotion monitoring, focusing on deployability, modular integration, and data-governance considerations rather than model benchmarking. The architecture defines a complete pipeline comprising a visual data acquisition layer, a processing backend for transformer-based inference, and a web-based visualization interface intended for aggregated emotional analytics. Publicly available datasets such as FER2013 and AffectNet are identified as reference sources for model adaptation within the proposed framework. The work details system components, data flow, scalability strategies, and privacy-by-design mechanisms, including transient image handling and non-persistent processing. Rather than presenting experimental performance, this study provides a technical blueprint and feasibility analysis intended to guide future implementation and validation of transformer-driven emotion monitoring systems in workplace contexts. The proposed framework aims to bridge the gap between advances in deep learning models and their practical integration into real-world organizational infrastructures.**

***Keywords*—Facial emotion recognition; vision transformer; affective computing; system architecture design; workplace emotion monitoring; privacy-by-design; deep learning inference**

I. Introduction

Emotional well-being in the workplace is a recognized determinant of organizational sustainability, employee retention, and productivity. In Metropolitan Lima, this issue carries particular relevance given a sustained rise in occupational stress and fatigue, with recent reports estimating that approximately 62% of employees exhibit significant stress-related symptoms, a condition further intensified by the COVID-19 pandemic [1], [2]. These affective states are associated with elevated absenteeism, increased staff turnover, and diminished operational performance, underscoring the need for scalable and systematic monitoring approaches.

Current organizational responses rely predominantly on periodic self-reporting instruments, such as psychometric surveys or structured interviews, which are limited by their discrete nature, susceptibility to social desirability bias, and inability to capture emotional fluctuations in real time. There is therefore, a notable gap in automated, non-intrusive systems for continuous affective state monitoring in workplace environments [3]. Although recent deep learning approaches have reported facial emotion recognition accuracies exceeding 90% under controlled conditions [4], their translation into deployable organizational systems remains largely unaddressed, further compounded in the Peruvian context by the scarcity of locally representative datasets and cultural variability in facial expression patterns [5].

Vision Transformers (ViT) offer a promising pathway to address this gap by modeling global spatial dependencies across facial image patches through self-attention mechanisms, enabling more generalizable representations of facial expressions under unconstrained conditions compared to convolutional approaches [6]. However, model performance alone does not translate into operational systems. Workplace deployment introduces architectural requirements beyond inference accuracy, including modular integration, real-time data flow, visual analytics for non-technical stakeholders, and data governance compliance, constituting a distinct design problem that motivates the present work.

This study addresses how a ViT-based pipeline can be systematically designed to meet the scalability, modularity, and privacy requirements of continuous workplace emotion monitoring. The study proposes a complete system architecture comprising a visual data acquisition layer, a cloud-based inference backend, and a web-based visualization interface for aggregated emotional analytics. The contribution is explicitly architectural: this work provides a technical blueprint and feasibility analysis to guide future implementation and empirical validation, rather than reporting experimental performance benchmarks.

II. Related Works

Research on transformer-based affective computing has advanced considerably at the international level, though most existing work focuses on multimodal inference under controlled experimental conditions rather than on deployable system architectures. These contributions are reviewed here to contextualize the architectural gaps that the present work addresses, rather than to establish direct performance comparisons with the proposed unimodal system.

In the domain of multimodal emotion recognition, [5] introduced the KoHMT model (Knowledge-Oriented Hybrid Model Transformer), a hybrid architecture combining the HuBERT model for auditory feature extraction and

KoELECTRA for textual representation, fused through a cross-attention mechanism. Trained on AI-Hub datasets and deployed on Linux with NVIDIA GPUs, KoHMT demonstrates the expressive potential of combining speech and text modalities for affective state classification. However, its reliance on multiple synchronized input streams and the absence of an integrated sensing and visualization layer limit its applicability as a blueprint for continuous, camera-based workplace monitoring systems.

Similarly, [3] proposed the TCMA model (Transformer-based Cross-Modal Attention), which combines facial imagery with non-invasive physiological signals using a 1D-CNN for rPPG extraction and ResNet50 for visual features, fused through cross-modal attention. Trained on MAHNOB-HCI and DEAP datasets, TCMA achieves high accuracy under laboratory conditions but requires physiological sensing hardware and lacks a modular deployment strategy suitable for enterprise-level integration. Both KoHMT and TCMA illustrate the state of the art in affective inference quality while simultaneously highlighting the absence of end-to-end architectural frameworks oriented toward organizational deployment, data governance, and non-technical user interfaces.

Within the unimodal vision domain, transformer-based models such as POSTER [7] and TokenFace [8] have demonstrated that ViT-derived architectures can achieve competitive FER performance using facial image data alone, without reliance on auxiliary modalities. These works provide direct architectural precedent for the inference component proposed in the present study and support the selection of ViT as the core model within a unimodal, camera-based monitoring pipeline.

At the national level, the landscape of applied computer vision research in Peru remains nascent. The most representative work identified in this context is [6], which presented a vision-based system for automatic vehicle license plate recognition using YOLOv4 and convolutional neural networks, trained on a locally collected dataset of one thousand images captured in Lima. While this work demonstrates the technical feasibility of deploying vision-based sensing systems within the Peruvian context, it is confined to vehicular recognition and does not engage with facial analysis, affective computing, or the architectural requirements of distributed emotional monitoring. Notably, no peer-reviewed work was identified addressing transformer-based facial emotion recognition systems designed or validated within Peru, representing a significant gap that the present proposal directly targets.

The reviewed literature collectively reveals two interconnected deficiencies: first, the predominance of multimodal and laboratory-oriented FER systems that prioritize inference accuracy over deployability; and second, the complete absence of ViT-based emotion monitoring architectures adapted to the Peruvian organizational context. The present work addresses both dimensions by proposing a system architecture that prioritizes modularity, scalability, and privacy-by-design over experimental benchmarking.

III. Architecture Design

This section presents the main contribution of this study: an integrated architecture for automatic facial emotion monitoring in workplace environments. The proposed system is designed for deployment on the Google Cloud Platform (GCP), leveraging Vision Transformers (ViT) as the core inference component for facial image analysis. ViT architectures process images by dividing them into fixed-size patches and modeling global spatial dependencies through self-attention mechanisms, enabling robust facial expression representation under unconstrained real-world conditions [8]. The overall design supports automated processing, scalable data management, and continuous inference, structured across three interconnected layers: the technological layer, the application layer, and the business layer.

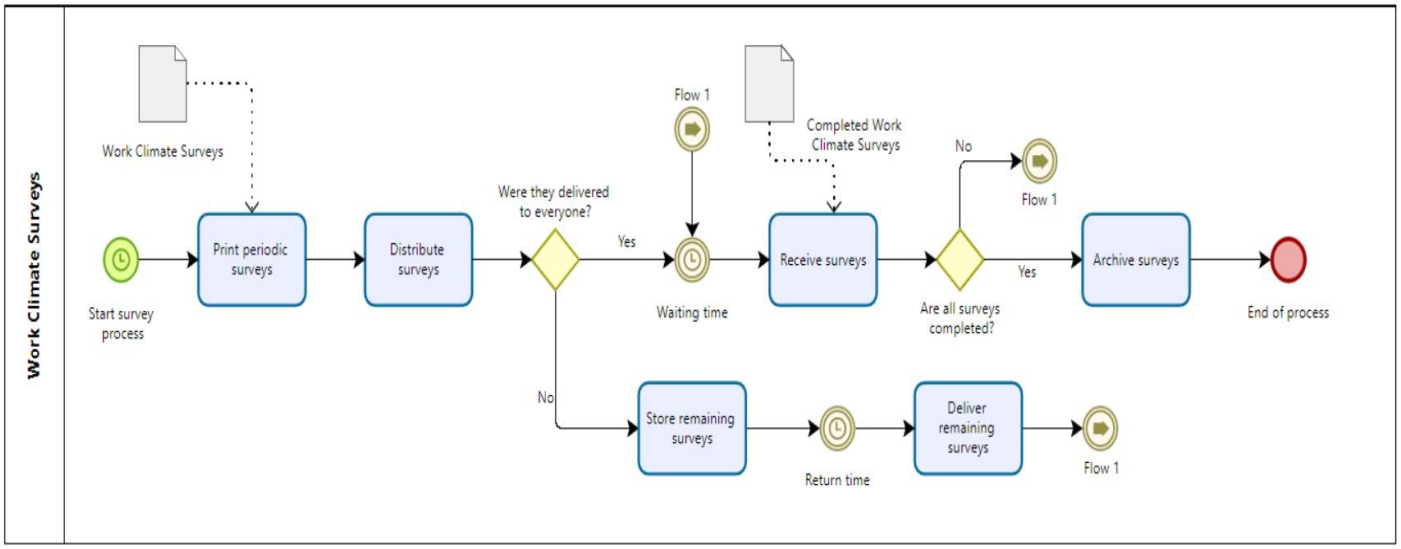


Fig. 1. Workflow of the workplace climate survey process.

B. Current Analysis

To contextualize the architectural requirements of the proposed system, a diagnostic interview was conducted with Human Resources personnel at a private organization operating in Metropolitan Lima. The interview aimed to characterize current practices for workplace emotional monitoring and identify operational limitations that an automated system should address. The organization requested anonymity; its profile is considered representative of mid-sized enterprises in the Peruvian service sector.

The interview revealed that emotional monitoring in this organization is conducted exclusively through periodic paper-based or digital survey instruments, distributed manually to employees at fixed intervals. As illustrated in Fig. 1 above, the current workflow is triggered by a scheduled timer event and initiates with the physical printing and manual distribution of survey forms. The process includes a verification checkpoint to confirm whether forms were delivered to all employees, with a dedicated sub-flow for handling pending deliveries and an intermediate waiting period before collection. Once surveys are received, a second decision point determines whether all forms have been completed before the results are archived. The entire process concludes with physical or digital storage of the completed documents, with no automated processing, inference, or real-time visualization at any stage.

This workflow exhibits several structural limitations relevant to the design of the proposed system. First, the timer-triggered nature of the process means emotional data is captured only at discrete intervals, preventing the detection of affective fluctuations that develop between survey cycles. Second, the manual distribution and collection of forms introduce operational overhead for HR personnel and potential disruptions to employee workflow. Third, self-reported survey responses are inherently susceptible to social desirability bias and subjective interpretation, limiting the reliability of the collected data. Fourth, the absence of any analytical or visualization layer means that aggregated emotional indicators cannot be derived in real time, constraining the organization's capacity for proactive decision-making.

These limitations collectively define the operational gap that the proposed architecture targets: the need for a continuous, non-intrusive, and automated system capable of capturing facial emotional indicators, performing transformer-based inference, and delivering aggregated analytics to organizational stakeholders without interrupting employee activities.

C. Design Decision Rationale

The selection of core architectural components for the proposed system was grounded in a systematic literature-based comparative analysis conducted across five dimensions: deep learning model architecture, specialized FER models, primary training dataset, complementary dataset, and cloud deployment platform. Each comparative analysis evaluated candidate options against weighted criteria derived from the technical requirements of a deployable workplace emotion monitoring system. The methodology assigned impact-weighted scores to each candidate based on performance metrics, characteristics, and limitations reported in their respective primary sources, enabling an evidence-based ranking of alternatives.

Model Architecture Selection. The first comparative analysis evaluated four deep learning architectures: CNN [9], RCNN [10], QCNN [11], and ViT [8], assessed against five criteria: emotion classification accuracy, inference time, robustness to environmental variation, adaptability to facial diversity, and architectural maturity. As shown in Table I, ViT achieved the highest weighted score (4.467), outperforming CNN (4.267), RCNN (4.267), and QCNN (3.867). ViT's self-attention mechanism enables the modeling of global spatial dependencies across facial image patches, yielding superior accuracy (89.90%) and robustness under variable illumination and pose conditions compared to convolution-based alternatives, which are constrained by local receptive fields. Although QCNN demonstrated competitive inference speed (95 ms), its robustness limitations and lower maturity for real-world deployment reduced its overall suitability.

TABLE. I. WEIGHTED COMPARATIVE ANALYSIS OF DEEP LEARNING ARCHITECTURES FOR FACIAL EMOTION RECOGNITION

Criterios	Impacto	ViT	RCNN	QCNN	CNN
		Puntaje	Puntaje	Puntaje	Puntaje
Accuracy in emotion classification	26.67%	5	4	4	4
Inference time	6.67%	3	2	2	1
Robustness in the face of changing conditions	20.00%	4	5	3	5
Adaptability to a diversity of expressions and faces	20.00%	4	5	3	4
Technologies, techniques, and models used	26.67%	5	4	5	5
Total		**21**	**20**	**17**	**19**

Specialized Model Validation. A second analysis extended the comparison to advanced architectures including 1D-CNN [12], SMaTE [13], TCMA [3], ResNet50 [14], and ViT [8], evaluated on classification accuracy, inference time, number of emotional classes, and generalization capacity. As shown in Table II, ViT and ResNet50 achieved equivalent weighted scores (4.555), with SMaTE (4.000), 1D-CNN (3.777), and TCMA (3.555) ranking lower. While SMaTE achieved the highest reported accuracy (99.19%) and ResNet50 demonstrated superior inference speed (28 ms), ViT was selected for its balanced profile: accuracy above 90%, multiclass coverage of seven emotion categories, robust generalization in non-controlled environments, and unimodal operation requiring only visual input. The exclusion of multimodal architectures such as TCMA was deliberate, as their dependency on physiological sensors is incompatible with the non-intrusive sensing constraints of the proposed workplace deployment.

TABLE. II. WEIGHTED COMPARATIVE ANALYSIS OF SPECIALIZED DEEP LEARNING MODELS FOR FER

Criterios	Impacto	1D-CNN	SMaTE	TCMA	ResNet50	ViT
		Puntaje	Puntaje	Puntaje	Puntaje	Puntaje
Accuracy in emotional classification	22,22%	4	5	4	5	5
Inference time	22,22%	4	3	2	4	4
Number of emotional classes evaluated	22,22%	3	4	4	4	4
Level of model generalization	33,33%	4	4	4	5	5
Total		15	16	14	18	**18**

Primary Dataset Selection. The third analysis compared FER2013 [15], CK+ [16], Aff-Wild2 [17], and RAF-DB [18] across five criteria: data volume, emotional coverage, demographic diversity, capture quality, and class balance. As shown in Table III, FER2013 achieved the highest weighted score (3.688), followed by RAF-DB (3.375), CK+ (2.938), and Aff-Wild2 (2.625). FER2013's 35,887 grayscale images (48×48 pixels) collected under unconstrained conditions provide a representative training base for real-world deployment, covering seven basic emotion categories with acceptable demographic diversity. Its class imbalance, particularly for the disgust category, can be mitigated through data augmentation strategies within the proposed pipeline.

TABLE. III. WEIGHTED COMPARATIVE ANALYSIS OF PRIMARY TRAINING DATASETS

Criterios	Impacto	Aff-Wild2	CK+	FER2013	RAF-DB
		Puntaje	Puntaje	Puntaje	Puntaje
Data volume	25%	3	2	5	3
Emotional coverage	18.75%	3	3	4	3
Demographic diversity	18.75%	3	2	5	5
Capture quality and environment	18.75%	3	5	3	3
Balance of emotional classes	18.75%	1	3	1	3
Total		**13**	**15**	**18**	**17**

Complementary Dataset Selection. The fourth analysis identified a supplementary dataset to enhance model generalization, evaluating AFEW [19], AffectNet [20], DEFE [21], and MELD [22] on emotional coverage, volume, demographic diversity, and capture conditions. As shown in Table IV, AffectNet achieved the highest weighted score (4.999), substantially outperforming MELD (4.000), AFEW (3.555), and DEFE (3.444). With over 450,000 manually annotated images covering eight discrete emotion categories and continuous valence-arousal dimensions, AffectNet provides the demographic diversity and in-the-wild variability necessary to improve ViT generalization across populations and environments relevant to the Peruvian workplace context.

TABLE. IV. WEIGHTED COMPARATIVE ANALYSIS OF COMPLEMENTARY DATASETS

Criterios	Impacto	AFEW	AffectNet	DEFE	MELD
		Puntaje	Puntaje	Puntaje	Puntaje
Emotional coverage	22.22%	4	5	4	4
Dataset volume	33.33%	2	5	3	4
Demographic diversity	22.22%	5	5	3	3
Capture conditions	22.22%	4	5	4	5
Total		**15**	20	14	16

Cloud Platform Selection. The fifth analysis evaluated deployment platforms including OsmOticGate2 [23], VaBUS [24], Google Cloud Platform [25], and MArk [26], assessed on processing latency, inference accuracy, operational scalability, and estimated cost. As shown in Table V, GCP achieved the highest weighted score (5.000), followed by OsmOticGate2 and MArk (both 4.000) and VaBUS (3.100). GCP's ecosystem of specialized services, including Vertex AI for model serving, Cloud Storage for transient image handling, Firestore for structured data management, and Compute Engine for scalable inference, provides a comprehensive and cost-effective infrastructure for the proposed architecture. Its support for GPU and TPU acceleration enables inference latency optimization within the 100–300 ms range suitable for near-real-time workplace monitoring.

TABLE. V. WEIGHTED COMPARATIVE ANALYSIS OF CLOUD DEPLOYMENT PLATFORMS

Criterios	Impacto	OsmOticGate2	VaBUS	Google Cloud	MArk
		Puntaje	Puntaje	Puntaje	Puntaje
Processing latency	30%	5	2	3	3
Accuracy in emotional inference	20%	3	5	5	5
Operational scalability	20%	5	3	5	3
Estimated operating cost	30%	3	3	5	3
Total		16	13	**18**	**14**

These five decisions collectively define the technical foundation of the proposed architecture: a ViT-based inference model fine-tuned on FER2013 and AffectNet, deployed on Google Cloud Platform within a modular pipeline designed for organizational scalability and privacy compliance.

D. Logical Architecture

The proposed logical architecture is structured into three interconnected layers: the business layer, the application layer, and the technological layer, as illustrated in Fig. 2. Together,

these layers define the system's functional components, information flows, and actor interactions, ensuring modularity and scalability across local and cloud-based environments.

The business layer defines the two organizational actors. The Human Resources department accesses the system via a web browser to visualize aggregated emotional indicators, apply departmental filters, and export reports. Employees interact through a local desktop application (Lumora-Employee.exe) that handles facial image capture and transmission, requiring no manual intervention beyond initial setup. No individual-level emotional profiling is exposed to any user role.

The application layer mediates between user interaction and deep learning inference. The front-end, hosted on Vercel, provides a dashboard with charts and filters, a visualization module, and a document export module. The GCP-hosted back-end integrates a REST API, a ViT-based inference module, and a database storing results, timestamps, and organizational metadata. A local edge component on the employee workstation performs preliminary facial capture processing before transmitting results to the cloud, reducing bandwidth requirements and supporting near-real-time inference.

The technological layer provides the supporting infrastructure, including firewall-governed network connectivity and Cloud Storage for transient image handling, ensuring facial images are not retained beyond the inference cycle in accordance with the privacy-by-design principles of the proposed framework.

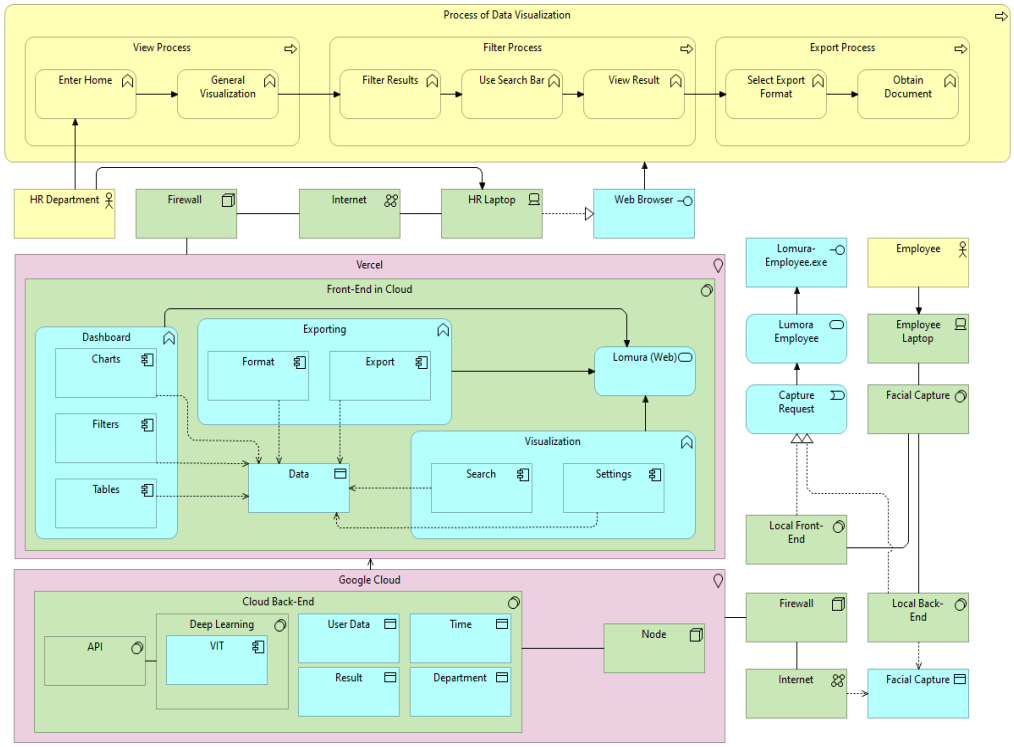


Fig. 2. Logical architecture of the proposed facial emotion monitoring system.

E. Physical Architecture

The physical architecture defines the hardware and infrastructure components supporting the end-to-end data flow from facial image acquisition to aggregated visualization, as illustrated in Fig. 3. The design is organized across four layers: client and collection, communication, cloud back-end, and presentation.

At the client and collection layer, each employee workstation consists of a laptop equipped with a camera running a local application composed of a frontend module handling login and user interaction, and a backend module managing capture logic through standard controllers, services, and entities. Facial images are captured at predefined intervals and transmitted via HTTP through a REST API to the cloud infrastructure. Images are used exclusively for inference and discarded immediately upon completion, with no persistent storage of visual data.

Within the cloud back-end layer, hosted on GCP, a server-side application receives incoming data through the API and processes it using a deep learning module containing the trained ViT model. As shown in Fig. 3, the ViT implementation processes facial image patches through a Transformer encoder to generate emotion classification outputs, which are stored as structured prediction results alongside timestamps and organizational metadata.

At the presentation layer, HR personnel access the system from a dedicated client device through a web-based frontend built with React and deployed on Vercel. This interface provides interactive dashboards for visualizing aggregated emotional indicators and generating exportable

reports at the organizational level.

F. Projected System Flow

The projected system flow defines the expected operational behavior once the proposed architecture is deployed, replacing the current manual survey-based process with an automated, continuous monitoring pipeline. The redesigned workflow is illustrated in Fig. 4 and organized across two interconnected processes: the emotion monitoring process and the HR department consultation process.

The emotion monitoring process is triggered automatically by a timer event, eliminating the need for employee intervention. The system captures facial images from the employee workstation, uploads them to cloud storage, and processes them through the ViT inference module. The resulting emotion classification outputs are then managed through a data gateway: if no results table exists, a new one is generated; otherwise, new records are appended to the existing table. The updated results table subsequently triggers a signal to the HR consultation process, ensuring dashboard data remains current without manual consolidation.

The HR department process initiates when an authorized HR manager accesses the web platform. The system reads the updated results table and evaluates whether data is available: if no results are found, an empty dashboard is displayed; if results exist, the dashboard is refreshed with the current aggregated indicators. The HR manager can then sequentially consult global indicators, personnel metrics, emotional trend charts, and individual employee records, concluding with the delivery of consolidated information to the HR area for organizational decision-making.

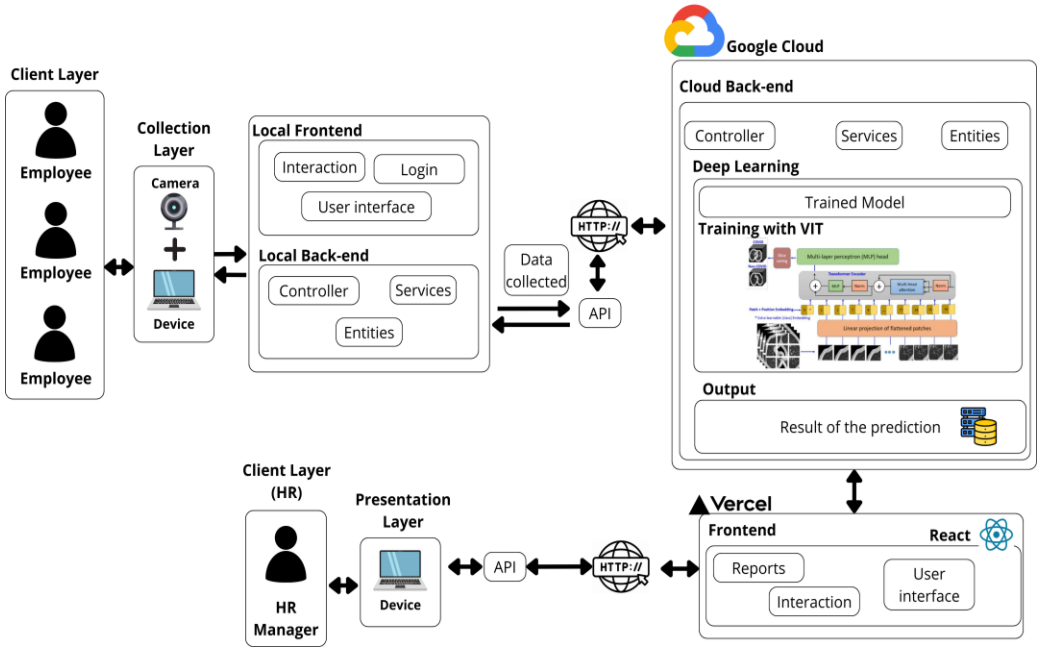


Fig. 3. Physical architecture of the proposed facial emotion monitoring system.

Compared to the current AS-IS workflow described in Section III(A), this proposed flow eliminates document printing, manual survey distribution, waiting periods, and physical archiving. The HR department gains continuous access to aggregated emotional indicators without administrative overhead, while employees contribute data passively without interrupting their work activities.

G. Ethical and Privacy Considerations

The proposed system incorporates privacy-by-design as a foundational architectural principle, ensuring that data protection mechanisms are embedded from the initial design stage rather than applied retroactively. Facial images captured by the local client are processed transiently in memory, transmitted via encrypted connections to the cloud inference module, and automatically discarded upon inference completion. No facial image is written to persistent storage at any stage, fulfilling the data minimization and storage limitation principles established under Peruvian Law N° 29733 on Personal Data Protection [27] and aligned with international Privacy by Design guidelines.

The system operates exclusively at an aggregated organizational level, associating inference outputs with departmental and temporal metadata rather than persistent individual identifiers, thereby preventing individual emotional profiling. Access to the visualization dashboard is restricted to authorized HR personnel through secure authentication mechanisms. These design choices are consistent with Law N° 29783 on Occupational Safety and Health [28], which mandates the protection of employees' physical and psychological integrity in the workplace, and with Law N° 31814 [29], Peru's first legal framework promoting ethical and responsible AI use in organizational contexts.

At the architectural level, system quality attributes including security, reliability, and maintainability are governed by ISO/IEC 25010:2023 [30], while responsible AI design principles follow the guidelines of ISO/IEC JTC 1/SC

42 [31]. Together, these standards and legal references substantiate the ethical feasibility of deploying the proposed architecture in Peruvian workplace environments.

IV. Conclusion

This study proposed the design of a Vision Transformer-based architecture for automatic facial emotion monitoring in workplace environments, addressing the architectural gap between advances in deep learning models and their practical integration into organizational contexts. The proposed system defines a complete pipeline comprising a visual data acquisition layer, a ViT-based cloud inference backend, and a web-based visualization interface, structured across three interconnected layers that support scalability, modularity, and privacy-by-design compliance.

The literature-based comparative analyses conducted across five dimensions, covering model architecture, specialized FER models, training datasets, complementary datasets, and cloud deployment platforms, substantiate the technical decisions underlying the proposed design. The selection of ViT as the core inference component, FER2013 and AffectNet as reference datasets, and GCP as the deployment platform reflects an evidence-based approach oriented toward deployability and organizational scalability rather than benchmark optimization.

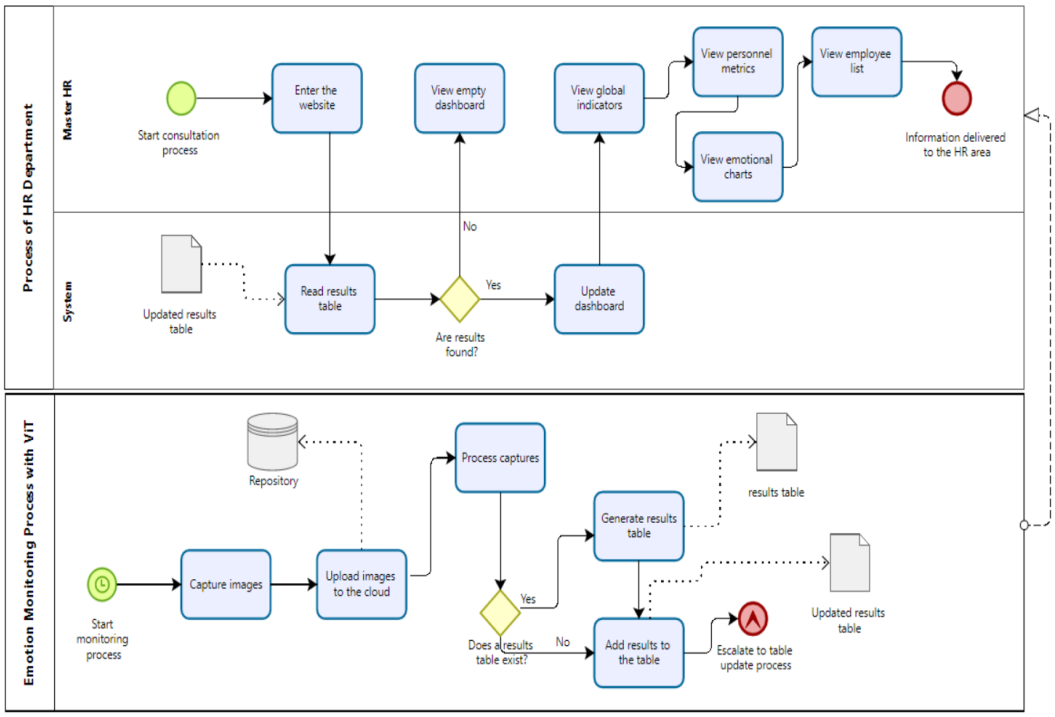


Fig. 4. Projected system flow for the proposed emotion monitoring architecture.

The projected system flow demonstrates that replacing periodic manual surveys with continuous automated monitoring can reduce administrative overhead, eliminate workflow interruptions, and provide HR departments with timely aggregated emotional indicators to support preventive organizational decision-making. These contributions are framed within applicable Peruvian legal standards and international quality frameworks, establishing the ethical and regulatory feasibility of the proposed architecture.

This work is explicitly scoped as an architectural blueprint. The primary limitation is the absence of empirical validation, as no prototype implementation or pilot deployment has been conducted. Future work should focus on the development and experimental evaluation of the proposed system, including performance benchmarking of the ViT inference module on FER2013 and AffectNet, latency measurement under real workplace network conditions, and a pilot deployment in a Peruvian organizational context to assess system usability and HR acceptance.

References

[1] G. Requejo Pacheco, M. S. Villa Santillán, L. Ruiz Barrera, y E. E. Rojas de la Puente, "Síndrome de burnout en trabajadores empresariales en Perú," Revista de Ciencias Sociales (RCS), vol. XXIX, no. 3, pp. 470–483, Sep. 2023, doi: https://doi.org/10.31876/rcs.v29i3.40731.

[2] H. Ko, D. Kim, S.-S. Cho, D.-W. Lee, J. Choi, M. Kim, M. Y. Park, y M.-Y. Kang, "The association of emotional labor and workplace violence with health-related productivity loss," Journal of Occupational Health, vol. 66, no. 1, Art. no. uiae057, Sep. 2024, https://doi.org/10.1093/joccuh/uiae057.

[3] J. Li and J. Peng, "End-to-End Multimodal Emotion Recognition Based on Facial Expressions and Remote Photoplethysmography Signals," IEEE Journal of Biomedical and Health Informatics, vol. 28, no. 10, pp. 6194–6204, Oct. 2024, https://doi.org/10.1109/JBHI.2024.3430310.

[4] P. Siritanawan, H. Kojima, and K. Kotani, "Exploring the Cultural Gaps in Facial Expression Recognition Systems by Visual Features," in Proc. 2023 IEEE Region 10 Conf. (TENCON), Chiang Mai, Thailand, Oct. 2023, pp. 1–6, https://doi.org/10.1109/TENCON58879.2023.10322368.

[5] M.-H. Yi, K.-C. Kwak, and J.-H. Shin, "KoHMT: A Multimodal Emotion Recognition Model Integrating KoELECTRA, HuBERT with Multimodal Transformer," Electronics, vol. 13, no. 23, p. 4674, Dec. 2024, https://doi.org/10.3390/electronics13234674.

[6] D. M. Valdeos Acevedo, A. S. Vadillo Velazco, M. G. S. Pérez Paredes, and R. M. Arias Velásquez, "Methodology for an automatic license plate recognition system using convolutional neural networks for a Peruvian case study," IEEE Latin America Transactions, vol. 20, no. 6, pp. 1032–1039, Jun. 2022, https://doi.org/10.1109/TLA.2022.9757747.

[7] M. Karnati, A. Seal, D. Bhattacharjee, A. Yazidi, and O. Krejcar, "Understanding deep learning techniques for recognition of human emotions using facial expressions: A comprehensive survey," IEEE Transactions on Instrumentation and Measurement, vol. 72, pp. 1–26, 2023, https://doi.org/10.1109/TIM.2023.3243661.

[8] L. Papa, P. Russo, I. Amerini, and L. Zhou, "A survey on efficient vision transformers: Algorithms, techniques, and performance benchmarking," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 46, no. 12, pp. 7682–7703, Dec. 2024, https://doi.org/10.1109/TPAMI.2024.3392941.

[9] M. A. Saleem, N. Senan, F. Wahid, M. Aamir, A. Samad y M. Khan, "Comparative analysis of recent architecture of convolutional neural network," Mathematical Problems in Engineering, vol. 2022, 2022, https://doi.org/10.1155/2022/7313612.

[10] S. Reddy, N. Pillay y N. Singh, "Comparative evaluation of convolutional neural network object detection algorithms for vehicle detection," Journal of Imaging, vol. 10, no. 7, 2024, https://doi.org/10.3390/jimaging10070162.

[11] S. Hossain, S. Umer, R. K. Rout y H. A. Marzouqi, "A deep quantum convolutional neural network based facial expression recognition for mental health analysis," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 32, pp. 1556–1565, 2024, https://doi.org/10.1109/TNSRE.2024.3385336.

[12] G. Rescio, A. Manni, M. Ciccarelli, A. Papetti, A. Caroppo y A. Leone, "A deep learning-based platform for workers' stress detection using minimally intrusive multisensory devices," Sensors, vol. 24, no. 3, 2024, https://doi.org/10.3390/s24030947.

[13] N. Kim, S. Cho y B. Bae, "SMaTE: A segment-level feature mixing and temporal encoding framework for facial expression recognition," Sensors, vol. 22, no. 15, 2022, https://doi.org/10.3390/s22155753.

[14] S. H. Al-Gburi, K. A. Al-Sammak, I. Marghescu, C. C. Oprea, A.-M. C. Drăgulinescu, N. A. M. Alduais, K. M. A. Alheeti y N. A. H. Al-Sammak, "EffRes-DrowsyNet: A novel hybrid deep learning model combining EfficientNetB0 and ResNet50 for driver drowsiness detection," Sensors, vol. 25, no. 12, art. no. 3711, 2025, https://doi.org/10.3390/s25123711.

[15] A. Kashef, Y. Wang, M. N. Assafi, J. Ma, J. Wang, J. A. Jones, and L. Thiamwong, "Developing a novel AI-enabled extended reality system for real-time automatic facial expression recognition and system performance evaluation," Advanced Engineering Informatics, vol. 55, Feb. 2025, Art. no. 103207, https://doi.org/10.1016/j.aei.2025.103207.

[16] X. Dong, X. Ning, J. Xu, L. Yu, W. Li y L. Zhang, "A recognizable expression line portrait synthesis method in portrait rendering robot," IEEE Transactions on Computational Social Systems, vol. 11, no. 1, pp. 1440–1450, 2024, https://doi.org/10.1109/TCSS.2023.3241003.

[17] D. Dresvyanskiy, E. Ryumina, H. Kaya, M. Markitantov, A. Karpov y W. Minker, "End-to-end modeling and transfer learning for audiovisual emotion recognition in-the-wild," Multimodal Technologies and Interaction, vol. 6, no. 2, 2022, https://doi.org/10.3390/mti6020011.

[18] U. Nawaz, Z. Saeed y K. Atif, "A novel transformer-based approach for adult's facial emotion recognition," IEEE Access, 2025, https://doi.org/10.1109/ACCESS.2025.3555510.

[19] B. Pan, K. Hirota, Y. Dai, Z. Jia, S. Shao y J. She, "Learning sequential variation information for dynamic facial expression recognition," IEEE Transactions on Neural Networks and Learning Systems, 2025, https://doi.org/10.1109/TNNLS.2025.3548669.

[20] Z.-Y. Huang, C.-C. Chiang, J.-H. Chen, Y.-C. Chen, H.-L. Chung, Y.-P. Cai, and H.-C. Hsu, "A study on computer vision for facial emotion recognition," Scientific Reports, vol. 13, no. 8425, 2023, https://doi.org/10.1038/s41598-023-35446-4.

[21] W. Li, Y. Cui, Y. Ma, X. Chen, G. Li, G. Zeng, G. Guo y D. Cao, "A spontaneous driver emotion facial expression (DEFE) dataset for intelligent vehicles: Emotions triggered by video-audio clips in driving scenarios," IEEE Transactions on Affective Computing, vol. 14, no. 1, pp. 747–760, 2023, https://doi.org/10.1109/TAFFC.2021.3063387.

[22] A. Aguilera, D. Mellado y F. Rojas, "An assessment of in-the-wild datasets for multimodal emotion recognition," Sensors, vol. 23, no. 11, 2023, https://doi.org/10.3390/s23115184.

[23] B. Qian, Y. Xuan, D. Wu, Z. Wen, R. Yang, S. He, J. Chen y R. Ranjan, "Edge-cloud collaborative streaming video analytics with multi-agent deep reinforcement learning," IEEE Network, 2024, https://doi.org/10.1109/MNET.2024.3398724.

[24] H. Wang, Q. Li, H. Sun, Z. Chen, Y. Hao, J. Peng, Z. Yuan, J. Fu y Y. Jiang, "VaBUS: Edge-cloud real-time video analytics via background understanding and subtraction," IEEE Journal on Selected Areas in Communications, vol. 41, no. 1, pp. 90–106, 2023, https://doi.org/10.1109/JSAC.2022.3221995.

[25] V. D. Nguyen, H. V. Khoa, T. N. Kieu, and E.-N. Huh, "Internet of Things-based intelligent attendance system: Framework, practice implementation, and application," Electronics, vol. 11, no. 3151, Sep. 2022, https://doi.org/10.3390/electronics11193151.

[26] C. Zhang, M. Yu, W. Wang y F. Yan, "Enabling cost-effective, SLO-aware machine learning inference serving on public cloud," IEEE Transactions on Cloud Computing, vol. 10, no. 3, pp. 1765–1779, 2022, https://doi.org/10.1109/TCC.2020.3006751.

[27] Congress of the Republic of Peru, "Law No. 29733, Personal Data Protection Law," Peru, 2011. [Online]. Available: https://www.gob.pe/institucion/congreso-de-la-republica/normas-legales/243470-29733.

[28] Congress of the Republic of Peru, "Law No. 29783, Occupational Safety and Health Law," Peru, 2012. [Online]. Available: https://www.gob.pe/institucion/congreso-de-la-republica/normas-legales/462576-29783.

[29] Congress of the Republic of Peru, "Law No. 31814, Law promoting the use of Artificial Intelligence for the development of the country," Peru, 2023. [Online]. Available: https://www.gob.pe/institucion/congreso-de-la-republica/normas-legales/4565760-31814.

[30] ISO/IEC, "Systems and Software Engineering—Systems and Software Quality Requirements and Evaluation (SQuaRE)—Product Quality Model," ISO/IEC 25010:2023, International Organization for Standardization, Geneva, Switzerland, Nov. 2023. [Online]. Available: https://www.iso.org/standard/78176.html.

[31] ISO/IEC JTC 1/SC 42, "Artificial Intelligence — Subcommittee SC 42," International Organization for Standardization, Geneva, Switzerland. [Online]. Available: https://www.iso.org/committee/6794475.html.

Integrating Acoustic and Image Data Features for Melon Ripeness Classification Using Convolutional Neural Network

Endang Purnama Giri[1], Agus Buono[2], Karlisa Priandana[3], Dwi Guntoro[4]
Computer Science Study Program-School of Data Science, Mathematics and Informatics, IPB University, Bogor, Indonesia[1, 2, 3]
Department of Agronomy-Faculty of Agriculture, IPB University, Bogor, Indonesia[4]

***Abstract*—This study evaluates three classification scenarios: image-based only, acoustic-based using Mel Frequency Cepstral Coefficients (MFCC), and a combined multimodal CNN architecture integrating both modalities. The experiments are conducted on a relatively small dataset comprising only 230 samples. To mitigate the risk of overfitting arising from the limited dataset size, data augmentation is applied to both image and audio data, with audio augmentation performed before the construction of the MFCC spectrogram. Experimental results demonstrate that the multimodal CNN with data augmentation achieves the best performance, with precision, recall, and F1-score, respectively, 0.95, 0.94, and 0.94. These results indicate that augmenting both image and audio data effectively enhances data diversity and model robustness, significantly improving classification performance. The findings confirm that combining complementary feature representations from multiple modalities with proper augmentation strategies substantially improves audio-visual classification tasks.**

***Keywords*—CNN; multimodal learning; melon ripeness classification; image classification; acoustic classification; data augmentation**

I. Introduction

Accurate melon ripeness classification is crucial for melon cultivation in Indonesia, as timely harvesting directly affects fruit quality. Being a climacteric fruit, melons continue to ripen after harvest, and their distribution is highly diverse, spanning both domestic and international markets. Therefore, precise classification is essential to ensure optimal harvest timing and maintain quality throughout the supply chain. Accordingly, this study aims to develop a nondestructive melon ripeness classification technique based on CNN.

In recent years, advancements in deep learning techniques have paved the way for innovative applications in agriculture, including the classification of fruit ripeness and sweetness. Melons, a widely cultivated and commercially valuable fruit, come in various varieties with distinct characteristics. However, due to differences in ripeness levels and sweetness, determining the optimal harvest time can be challenging. This challenge has led to the growing interest in nondestructive techniques to assess these qualities. To ensure the study remains focused, this research specifically examines two popular melon varieties: Golden Apollo and Golden Alisha. These varieties are widely cultivated and possess characteristics that make them suitable for classification studies. Additionally, the melons used in this research are assumed to be under controlled environmental conditions, such as consistent greenhouse settings (temperature, humidity, and light exposure) and uniform nutritional treatments, ensuring minimal variation caused by external factors.

The features explored in this study are limited to acoustic-based and image-based features, as they have shown significant potential in previous research for nondestructive classification methods. Furthermore, the classification models are developed using the Convolutional Neural Network (CNN) architecture, which has demonstrated superior performance in image and feature extraction tasks. This study prioritizes improving the accuracy of the classification model, while considerations of time and space complexity are beyond the primary scope. By addressing these aspects, the study aims to push the boundaries of existing research by investigating feature relevance, visualizing feature extraction, and modifying CNN architectures to combine acoustic and visual features. The outcomes of this research are expected to contribute to more accurate and efficient nondestructive methods for classifying melon ripeness and sweetness, providing practical benefits to the agricultural industry.

II. Previous Research

Previous research has consistently demonstrated the effectiveness of Convolutional Neural Networks (CNNs) in addressing classification tasks involving both acoustic and visual features. For speech recognition, CNNs integrated into a hybrid neural network-Hidden Markov Model (NN-HMM) framework achieved an accuracy of 79.93%, surpassing standard neural networks (77.05%) and Deep Belief Network-pretrained models (79.5%) using the TIMIT dataset [1]. Further studies confirmed CNN's superiority in reducing error rates by 6–10% compared to Deep Neural Networks (DNN)-HMM models [2]. Similarly, in phoneme classification using raw audio signals, CNN achieved a significant improvement in accuracy, reaching 67.88% compared to 38.91% with Multi-Layer Perceptron (MLP) models [3]. In music onset detection, CNN also outperformed Recurrent Neural Networks (RNNs), achieving 91.7% accuracy compared to RNN's 89.2% [4]. In environmental sound classification, CNNs showed remarkable performance, achieving 77.85% accuracy on the ESC-10 dataset and 49% on ESC-50, compared to 56% by Tensor Deep Stacking Networks (TDSN) [5]. Another study on forest sound classification, which included bird chirps and chainsaw noises,

reported CNN achieving 85% accuracy using spectrogram features [6]. CNN was further validated as the top-performing model for construction noise classification, achieving 97% accuracy with Mel-spectrograms, outperforming other models such as Random Forest (93%), MLP (92%), k-NN (85%), and SVM (84%) [7].

In visual-based tasks, CNN also excelled. For traffic sign and light classification using RGB-D image features, CNNs achieved 93.33% accuracy [8]. In fruit image classification, CNN achieved an impressive 98.88% accuracy on a dataset comprising 55,244 images of 81 fruit categories using a pure CNN architecture with seven layers and global average pooling [9]. For large-scale image classification, CNN architectures such as GoogLeNet and ResNet achieved an accuracy of 83.12% on the ImageNet dataset, consisting of 1.2 million high-resolution images across 1,000 categories [10]. These studies consistently highlight the capability of CNNs in extracting relevant features, improving classification accuracy, and outperforming traditional machine learning approaches. These findings suggest that CNNs hold great potential for application in challenging tasks like the classification of melon ripeness and sweetness using acoustic and visual features.

Previous studies have demonstrated that multimodal fusion, particularly the integration of acoustic and visual features within CNN-based frameworks, can outperform unimodal approaches by leveraging complementary temporal and spatial information. Such strategies have been successfully applied to domains including environmental event detection [11], human activity recognition [12] and [13], and object classification [14], resulting in improved robustness and classification accuracy. Nevertheless, despite the effectiveness of audio–visual fusion in these fields, no existing studies have explored its application to melon ripeness and sweetness classification. This research, therefore, addresses a clear gap in the literature by extending established multimodal CNN paradigms to agricultural quality assessment through the joint exploitation of acoustic and visual cues.

III. Methods

In this study, we used CNN as our deep learning architecture to construct a model for melon ripeness classification. In this section, a brief explanation of all methods used in this research is provided.

A. CNN Architecture

CNN is a deep learning architecture specifically designed for processing data with a grid-like topology, such as images and time-series data. It is characterized by its ability to automatically extract relevant features from raw data through layers of convolution and pooling operations. The convolution layers apply filters to the input data, capturing spatial hierarchies and patterns such as edges, textures, and more complex features in deeper layers. Pooling layers, on the other hand, reduce the spatial dimensions of the data, making the model computationally efficient while preserving the most critical information. CNN typically includes several main components: convolutional layers, activation functions (e.g., ReLU), pooling layers, fully connected layers, and an output layer with a softmax or sigmoid function for classification tasks. Advanced CNN architectures often integrate additional mechanisms, such as dropout for regularization or batch normalization to accelerate training.

Convolutional Neural Networks (CNNs) are a type of feed-forward neural network designed to process visual data. In CNNs, individual neurons are arranged to respond to overlapping regions within a visual area [15]. CNNs are hierarchical networks composed of multiple convolutional layers and subsampling (pooling) layers. The architecture can be varied by adjusting the number and type of convolutional and pooling layers. The convolutional layer operates based on parameters such as the size and number of feature maps, kernel size, skipping factors, and connection tables. A kernel slides across the input image, capturing spatial features, while skipping factors determine how many pixels are skipped during this operation. The output map size is calculated using a specific formula. Each map in a layer connects to several maps in the previous layer, and neurons in the same map may share weights.

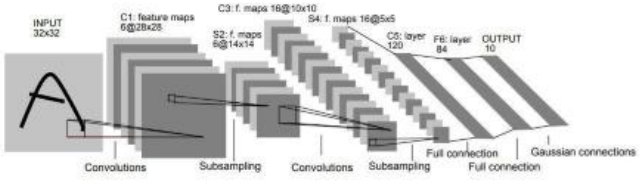


Fig. 1. CNN main architecture [16].

The CNN algorithm consists of two main stages: convolution and subsampling (pooling). For example, in the MNIST dataset architecture (Fig. 1), a grayscale image with dimensions 32x32 pixels is processed through two pairs of convolution and pooling layers, followed by two fully connected layers. Key steps include:

1) First convolution layer: Six 5x5 kernels produce six feature maps of 28x28 pixels using stride 1 and a valid mode (no padding).

2) First pooling layer: Reduces the feature maps to 14x14 pixels with a subsampling factor of 2.

3) Second convolution layer: Sixteen 5x5 kernels create 16 feature maps, every 10x10 pixels.

4) Second pooling layer: Further reduces the feature maps to 5x5 pixels.

5) Fully connected layers: The final pooling layer outputs connect to 120 neurons, followed by 84 neurons, and finally to 10 output neurons representing the classification categories.

6) Activation functions: Non-linear activation functions (e.g., Tanh, Sigmoid, or ReLU) are applied between convolution and pooling layers, as well as between the fully connected layers.

This hierarchical structure allows CNNs to automatically extract and learn relevant features from raw input data, making them highly effective for tasks such as image recognition and classification.

The foundation of Convolutional Neural Networks (CNNs) lies in the architecture of Neural Networks (NNs). However,

CNNs have three main advantages over traditional NNs: sparse connectivity, parameter sharing, and equivariance, which make them more efficient and effective for processing structured data, especially images.

- Sparse Connectivity: unlike traditional NNs, CNNs only connect a neuron to a small region of the input (called a receptive field). This sparse connectivity significantly reduces the number of parameters to train.
- Parameter Sharing: the same set of weights (or filters) is applied across the entire input. This parameter sharing allows CNNs to detect the same feature (e.g., edges, corners) regardless of its position in the input. As a result, the model becomes more efficient in terms of memory and computation, and it can generalize better to unseen data.
- Equivariance: CNNs exhibit equivariance to translation, meaning that a shift in the input (e.g., moving an object in an image) results in a corresponding shift in the feature map output. This property makes CNNs robust to variations in object positions, which is crucial for tasks like image recognition.

These three properties collectively allow CNNs to efficiently handle high-dimensional data like images, reduce overfitting, and achieve better performance in tasks involving spatial or temporal patterns.

B. Spectrogram Image

Although CNN architecture comes in various forms, the most notable and successful is the 2D-CNN architecture. For image data, which is already represented as a two-dimensional matrix, CNN-2D can be applied directly. However, this differs from acoustic data, whose representation is typically in the form of a one-dimensional vector (1-D). A common approach for classifying audio data using CNN-2D is to convert the audio data into a spectrogram. A spectrogram is an image that depicts the magnitude (intensity) of different frequencies over time. This spectrogram image is then used as a Deep Acoustic Feature (DAF) input for the 2D-CNN architecture. The process of creating a spectrogram involves the following steps:

1) Segmenting audio data: The audio data is divided into smaller overlapping chunks. For example, if an audio sample has a size of 1x7, with an overlap of 1 sample and a chunk width of 3 samples, the data is split accordingly.

2) Frequency decomposition using FFT: Assuming that a sound consists of various frequencies, each chunk is transformed using the Fast Fourier Transform (FFT). FFT extracts the frequency components of each chunk of audio at different time intervals, converting the data from the time-amplitude domain to the frequency-time domain.

3) Creating a frequency matrix: For instance, if each chunk is decomposed into 100 frequency components, each chunk will have 100 values. With 3 chunks, the results will result in a matrix of size 100 rows x 3 columns.

4) Mapping matrix to pixels: Each cell in the matrix is represented as a pixel with a specific color. The colors are assigned using a predefined threshold, where higher values are represented with brighter colors.

5) Generating the spectrogram image: The spectrogram is represented as an image, where the color of each pixel reflects the intensity of a frequency at a specific time. A pixel in the spectrogram at row i, column j represents the intensity of sound at the *i-th* frequency and the *s*-time range. Fig. 2 shows an example of a spectrogram from audio data.

6) Input to CNN-2D architecture: Since the spectrogram is a 2D grid of data (image), it can directly be used as input to a 2D-CNN architecture.

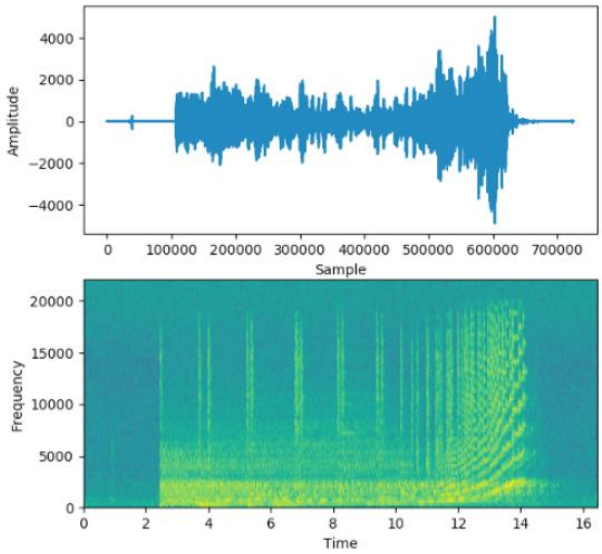


Fig. 2. Spectrogram image [17].

C. Mel Frequency Cepstral Coefficients (MFCC)

In addition to spectrograms, Deep Acoustic Features (DAF) can use Mel Frequency Cepstral Coefficients (MFCC) as input. The detailed MFCC extraction process involves the following steps [18]:

1) Frame blocking (Fig. 3): Audio signals are divided into overlapping frames to prevent information loss at frame transitions.

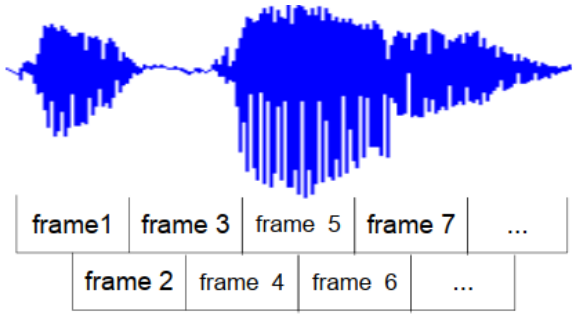


Fig. 3. Frame blocking procedures [18].

2) Windowing: A filter is applied to each frame to minimize distortion between frame blocks. This is done by multiplying each frame's data with a selected windowing

formula, commonly the Hamming window, due to its computational efficiency and effective filtering properties. The Hamming window equation is Eq. (1):

$$W(n) = 0{,}54 - 0{,}46\ cos\left(\frac{2\pi n}{N-1}\right) \quad (1)$$

Variable descriptions

- $W(n)$ the value of the Hamming window at the n-th sample
- n the sample index within a frame, where n=0, 1, 2, …, N−1
- N the total number of samples in one frame (window length)

3) Fast Fourier Transform (FFT) is applied to the windowed frames, calculating the Discrete Fourier Transform (DFT), which outputs complex numbers. The real and imaginary components are calculated as Eq. (2) and Eq. (3):

$$Real\ X[k] = \sum_{i=0}^{N-1} x[i] cos\left(\frac{2\pi k i}{N}\right) \quad (2)$$

$$Imaginary\ X[k] = -\sum_{i=0}^{N-1} x[i] sin\left(\frac{2\pi k i}{N}\right) \quad (3)$$

Variable description

- $Real\ X[k]$ the real part of the DFT output at index k
- $Imaginary\ X[k]$ the imaginary part of the DFT output at index k
- $x[i]$ The input signal sample at time index i (windowed signal)
- i time-domain sample index, where *i=0, 1, 2 ,..., N−1*
- k frequency bin index of the DFT output
- N total number of samples in one frame (FFT size)

The next step in this stage is to compute $X[k]$ the magnitude of the FFT. The magnitude of a complex number is calculated from its real and imaginary components using Eq. (4):

$$X[k] = \sqrt{Real\ X[k]^2 + Imaginary\ X[k]^2} \quad (4)$$

4) Mel-Frequency Wrapping, human perception of sound frequency is linear for frequencies ≤ 1000 Hz, but logarithmic for higher frequencies. This relationship, represented by the mel-frequency scale, is expressed as Eq. (5):

$$F_{mel} = \begin{cases} 2595 * log_{10}(1 + \frac{F_{Hz}}{700}), & F_{Hz} > 1000 \\ F_{Hz}, & F_{Hz} \leq 1000 \end{cases} \quad (5)$$

Mel-frequency wrapping, then performed using Eq. (6):

$$X_i = log_{10}(\sum_{k=0}^{N-1} X(k).H_i(k)) \quad (6)$$

Variable descriptions

- F_{mel}, the frequency value measured on the Mel scale, which represents the perceived frequency by the human auditory system.
- F_{Hz}, frequency value measured in Hertz (Hz), representing the actual or linear frequency of the signal.
- X_i, frequency wrapping value of the i-th filter
- i, filter index, where i=1,2,…,n, and n is the total number of filters
- $X(k)$, magnitude value of the frequency component at index k
- $H_i(k)$, amplitude (height) of the i-th triangular filter at frequency index k
- k, Frequency index, where k=0, 1, 2,…, N−1
- N, total number of frequency magnitude bins

5) Cepstrum Calculation, in this final step, the mel-frequency is converted into the time domain using the Discrete Cosine Transform (DCT), Eq. (7):

$$C_j = \sum_{i=1}^{M} X_i\ cos\left(\frac{j(i-1)}{2}\ \frac{\pi}{M}\right) \quad (7)$$

Variable descriptions

- C_j, j-th cepstral coefficient
- j, Cepstral coefficient index, where j=1,2,…, the desired number of coefficients, for audio classification, we use maximum j=40
- M, number of Mel filter banks. On these study, we used 64 Mel filters
- X_i, The Mel-frequency wrapping value at frequency index i, where i=1,2,…,n. The n, we set it similar to the number of M

In this study, acoustic data were used for Mel-Frequency Cepstral Coefficients (MFCC).

IV. Data

A. New Dataset Acquisition

From 11th May 2021, the procedure for obtaining new sample data has begun at the Agro Technology Park (ATP) Cikarawang. The melon data acquired is from the Golden Alisha melon variety. In general, the morphological shape of the Golden Alisha tends to be round, has a smooth skin texture, and has a bright golden yellow colour. In data acquisition, four different days after planting (DAP), class 47 DAP on 11th May, 53 DAP (17th May), 60 DAP (24th May) and 67 DAP (30th May). There were 12 melons used as samples. From each measured sample, five acoustic data and image data were obtained, respectively. So that when the data acquisition is completed, 60 pieces of data will be obtained for each different DAP and in total, it was hoped that 240 pieces of data would be obtained in all. In the greenhouse as the data location, there are four melon planting lines, each planting position has a different light intensity (Fig. 4). Based on these conditions, in order to be sufficiently representative, the measured melons were spread over four different planting lines. Twelve melons that were used as samples were spread out three each in each planting lane.



Fig. 4. Planting lines at greenhouse, ATP Cikarawang.

The tools used and measurement scenarios in data acquisition procedures are:

- Samsung A20 Smartphone camera, with 30 cm portrait distance, 3:4 dimension ratio, and 2.4x zoom aspect, with an image resolution of 3096x4128 pixels.
- Flux meter to measure light intensity during capturing data.

Some results of data acquisition procedures are shown in Fig. 5.



Fig. 5. Some data acquisition results.

As a result of the acquisition, finally obtained a dataset for ripeness classification are 230 acoustic knock (audio) and image data packets from 4 classes of DAP (47, 53, 60, and 67). Audio data in .m4a format, Image data used smartphone camera with dimensions of 3096x4128 pixels, Image data used Intel RealSense Depth camera with dimensions of 640x480 pixels. The proportion of data in each class is 55 records. The total amount of data is only 230 pieces of data and not 240 pieces of data because the acoustic data of a melon with ID 4 has been picked before 67 DAP and declared lost.

The dataset used in this study is relatively small (230 samples), which raises concerns about potential overfitting, particularly when employing deep CNN architectures. To mitigate this risk, data augmentation techniques were applied, including rotations, scaling, and flips, to artificially increase the diversity of training samples. These measures help ensure that the reported performance reflects the model's true capability rather than overfitting to the limited dataset.

V. Proposed Methods

In the case of classifying melons using two types of data: acoustic data from tapping the melon and image data of the melon, the following procedure is proposed for combining features from both data types. The integration is performed by merging the MFCC spectrogram values with the feature map values derived from the output of the last layer in the feature extraction stage (illustrated in Fig. 6, stage 2, represented by the rightmost purple neurons).

Challenges in Combining Spectrogram and Feature Maps to merge spectrogram and feature map data include:

- Dimensional Mismatch: The dimension of the spectrogram may vary for different acoustic data samples. Additionally, the spectrogram's dimensions may not match the dimensions of the feature maps.
- Value Range Differences: The value range (or domain) of the spectrogram will differ from that of the feature maps.

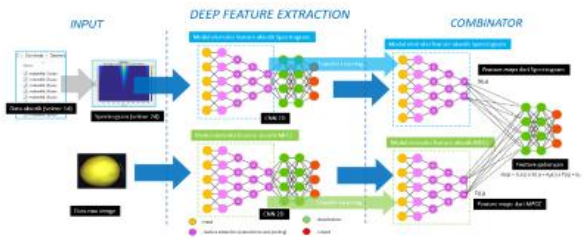


Fig. 6. Proposed method scheme.

Proposed Solutions to address these challenges, processed as these following steps. Combination Procedure - the procedure for combining the features is outlined in the diagram in Fig. 6. The value of the spectrogram, whose different dimensions can be combined by adding them with the trained weights. The vector of the combined feature G from the spectrogram S with weight A1, combined with the feature maps F with weight A2, can then be formulated using Eq. (8). This method is obtained by performing feature concatenation within a flatten layer, which allows the model to integrate information from both sources effectively before passing it to subsequent layers.

$$G(i,j) = A1(i,j)x\,S(i,j) + A2(i,j)xF(i,j) + b0 \qquad (8)$$

The weighted summation step runs after aligning the spectrogram's dimensions and normalizing its value range; it is combined with the feature map values by performing a weighted summation. Furthermore, the values of A1 and A2 are trained using the feed-forward and backpropagation approaches to determine the correct values. As for b0, it is the bias value.

VI. Experiments

A. Experimental Scenario

This section presents various experimental results conducted on this study. The CNN architecture used is illustrated in Fig. 7.

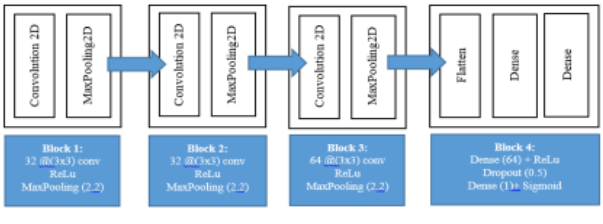


Fig. 7. CNN architecture on experiment.

In this study, there are three main experiment scenarios: (1) image-based only, (2) acoustic-based only using MFCC features, and (3) a combined multimodal CNN architecture. Data augmentation is applied to both image and audio data in

all scenarios. Augmentation of the audio data is applied before the MFCC spectrogram is constructed.

From a preliminary study, especially for acoustic-based classification model was investigated by comparing four experimental scenarios. The first scenario employed a CNN with Mel Spectrogram features and achieved moderate performance, with a precision of 0.62, recall of 0.55, F1-score of 0.55, and accuracy of 0.55. The second scenario utilized a CNN with MFCC Spectrogram features, which resulted in a consistent performance improvement, reaching a precision of 0.72, a recall of 0.65, an F1-score of 0.64, and an accuracy of 0.65. The third scenario incorporated data augmentation into the Mel Spectrogram-based CNN. However, this configuration did not yield a significant performance gain, as the accuracy remained at 0.55. In contrast, the fourth scenario, which combined a CNN with MFCC Spectrogram features and data augmentation, demonstrated a substantial improvement across all evaluation metrics. This approach achieved the highest precision (0.82), recall (0.80), F1-score (0.80), and accuracy (0.80) among all tested configurations. Based on these comparative results, it can be concluded that MFCC-based representations augmented through data augmentation provide a more robust and discriminative characterization of acoustic patterns. Consequently, the CNN with MFCC Spectrogram and data augmentation configuration is identified as the most effective approach and is therefore adopted as the baseline model in this study.

The hardware and software specifications used for this study are as follows:

- Hardware Environment Processor AMD Ryzen 9 5900HX (16 CPUs, 3.3 GHz) with Radeon Graphics. Intel Quad Core i7-4700HQ with a clock speed of 2.4 GHz, 16GB DDR5, and an SSD with 1 TB capacity.
- Software Environment and Supporting Applications: Operating System: Windows 11 Home Single Language 64-bit (Version 10.0, Build 26100). Python 3.7 and the Keras library with TensorFlow backend. Anaconda with Jupyter Notebook environment.

B. Experimental Results

This section presents the results of the classification experiments based on three different input modalities: image-based CNN, acoustic-based CNN, and multimodal CNN combining both image and audio features. The performance of each scenario is evaluated using precision, recall, and F1-score for each ripeness class (DAP47, DAP53, DAP60, and DAP67), as shown in Fig. 8- Fig. 10.

	precision	recall	f1-score	support
DAP47	1.00	0.75	0.86	20
DAP53	0.69	1.00	0.82	20
DAP60	0.89	0.80	0.84	20
DAP67	1.00	0.90	0.95	20
accuracy			0.86	80
macro avg	0.89	0.86	0.87	80
weighted avg	0.89	0.86	0.87	80

Fig. 8. Image-based CNN scenario result.

	precision	recall	f1-score	support
DAP47	0.88	0.70	0.78	20
DAP53	0.63	0.85	0.72	20
DAP60	0.89	0.85	0.87	20
DAP67	0.89	0.80	0.84	20
accuracy			0.80	80
macro avg	0.82	0.80	0.80	80
weighted avg	0.82	0.80	0.80	80

Fig. 9. Acoustic-based CNN scenario result.

Classification Report:

	precision	recall	f1-score	support
DAP47	0.95	1.00	0.98	20
DAP53	1.00	0.75	0.86	20
DAP60	0.83	1.00	0.91	20
DAP67	1.00	1.00	1.00	20
accuracy			0.94	80
macro avg	0.95	0.94	0.94	80
weighted avg	0.95	0.94	0.94	80

Fig. 10. Multimodal CNN scenario result.

TABLE I. EXPERIMENTAL RESULT SUMMARY

Scenario	Prec.	Rec.	F1-score	Acc.
Image-Based CNN	0.89	0.86	0.87	0.86
Acoustic-Based CNN	0.82	0.80	0.80	0.80
Multimodal CNN	0.95	0.94	0.94	0.94

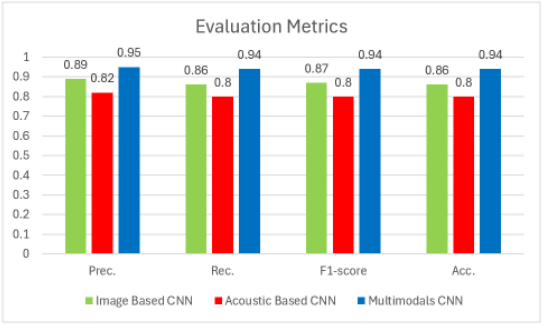


Fig. 11. Evaluation metrics.

For image-based CNN scenarios (Fig. 8), the model achieved an overall accuracy of 86%. The highest performance was observed in the DAP67 and DAP47 classes, with F1-scores of 0.86 and 0.86, respectively. However, the model struggled with class DAP53, which obtained a relatively low F1-score of 0.72. This indicates that visual features alone, while generally effective, may be insufficient for distinguishing between the earlier ripeness stages where external appearance changes are subtler.

On the other hand, the acoustic-based CNN scenario (Fig. 9) yielded a slightly lower overall accuracy of 80%. The best-performing classes were DAP60 and DAP67, with F1-scores of 0.87 and 0.84, respectively. Class DAP53 again presented the most difficulty, with an F1-score of just 0.72. These results suggest that audio features extracted from tapping sounds can represent internal fruit characteristics but may still

be ambiguous for similar ripeness levels, especially between DAP47 and DAP53.

Furthermore, for the last scenario, multimodal CNN (Fig. 10), which integrates both image and audio features, achieved the best overall performance with an accuracy of 94% and a macro-average F1-score of 0.94. All classes showed strong and balanced results, with F1-scores above 0.90, including DAP53, which previously had the lowest performance in other scenarios. The fusion of visual and acoustic information enabled the model to capture both external and internal cues associated with fruit ripeness, resulting in significantly improved classification performance.

In summary (Table I and Fig. 11), the multimodal CNN approach outperforms single-modality models, confirming that combining complementary features from different domains can enhance the accuracy and robustness of ripeness classification systems. This finding supports the potential of multimodal learning in agricultural quality assessment tasks, especially when the classification target involves subtle, multi-dimensional characteristics.

VII. Discussion

In this section, further discussion will be provided regarding the experimental results obtained. It includes confusion matrix analysis for all scenarios. At the end of this chapter, a statistical analysis of the experimental results is also conducted.

The confusion matrices presented in Fig. 11 to Fig. 13 illustrate the classification performance of three different CNN models using image data, acoustic data, and a multimodal fusion of both modalities. These matrices provide deeper insights into how well each model is able to distinguish between the four ripeness stages of melon: DAP47, DAP53, DAP60, and DAP67.

Fig. 12 shows the confusion matrix for the image-based CNN model, which demonstrates relatively strong performance overall. The model perfectly classified the DAP53 and DAP67 classes, while DAP47 was correctly predicted in 15 out of 20 cases, with 5 samples misclassified as DAP53. Class DAP60 showed a minor confusion with DAP53, as 4 samples were misclassified. This suggests that while image-based features are effective in capturing external ripeness characteristics, early ripeness stages such as DAP47 still tend to overlap visually with neighboring classes like DAP53.

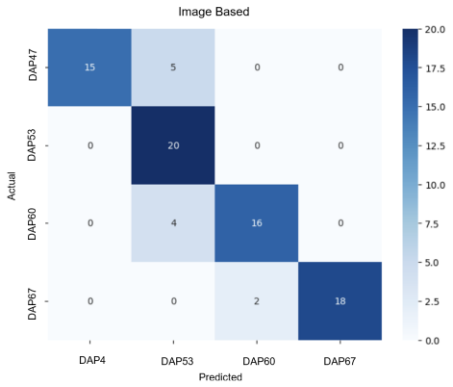


Fig. 12. Image-based CNN confusion matrix.

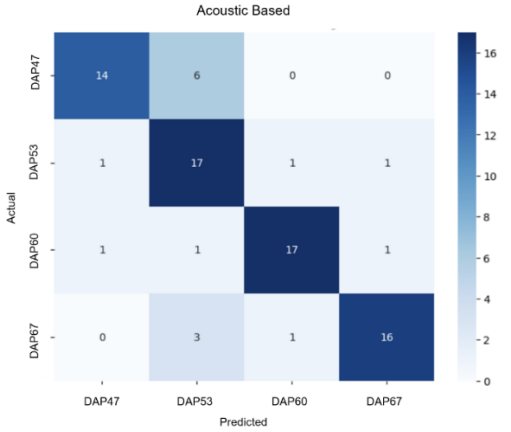


Fig. 13. Acoustic-based CNN confusion matrix.

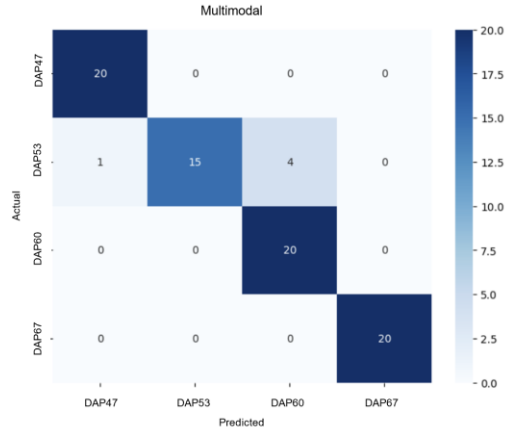


Fig. 14. Multimodal CNN confusion matrix.

In contrast, the acoustic-based CNN model with MFCC features and augmentation (Fig. 13) exhibited slightly lower precision across several classes. Class DAP47 was correctly identified 14 times but misclassified 6 times as DAP53. Similarly, DAP53 had 17 correct predictions, while the remaining 3 samples were confused with DAP47, DAP60, and DAP67. Notably, even DAP67, which was generally easy to classify, showed minor confusion in this setting. These results indicate that while acoustic features enriched with augmentation improve general robustness, they still present challenges in precisely distinguishing between classes with similar internal acoustic signatures, such as DAP47 and DAP53.

The most significant improvement is evident in Fig. 14, representing the multimodal CNN model that combines both image and audio features. This model achieves near-perfect classification: all samples from DAP47, DAP60, and DAP67 were classified correctly, while only one sample from DAP53 was misclassified as DAP47 and four as DAP60. The confusion in DAP53 remains the only notable weakness, which is consistent across models and suggests that this class may lie at a transitional point in both acoustic and visual characteristics. One notable difficulty during data acquisition was that the greenhouse was located near a rural road frequently used by vehicles. Consequently, acoustic recordings were often

contaminated with environmental noise, which likely contributed to the relatively lower performance of the acoustic-based model. This issue also affected the multimodal model, particularly in classifying samples from DAP53, where the presence of background noise and overlapping features made accurate classification more challenging. Despite this, the multimodal model achieved the best overall balance across all classes, demonstrating that fusing external (image) and internal (acoustic) cues significantly enhances the model's ability to discriminate between subtle ripeness stages.

In summary, the results affirm that while single-modality models (image-only or audio-only) are capable of capturing certain class characteristics, their performance is limited by the inherent ambiguities in either visual or auditory features alone. The multimodal CNN approach stands out as the most effective, offering a holistic representation of fruit ripeness and minimizing inter-class confusion. This supports the hypothesis that multimodal learning is highly beneficial for tasks requiring fine-grained classification, particularly in agricultural applications where the features are complex and multi-dimensional.

VIII. Statistical Analysis

This section presents a statistical analysis of the experimental results, with a particular focus on accuracy values. Statistical significance of performance differences among the three classifiers was evaluated using pairwise McNemar's tests on the same test samples. In addition, 95% confidence intervals and variance of accuracy were reported to assess the reliability and stability of the models.

The statistical analysis of the experimental results begins with the computation of the Standard Error (SE) and the Confidence Interval (CI). SE represents the magnitude of the sampling error of the accuracy estimate, indicating the extent to which the observed accuracy obtained from the test sample may deviate from the true performance on the broader data population. A smaller SE indicates a more stable and reliable performance estimate. CI provides a range within which the true model performance is expected to lie at a given confidence level (commonly 95%). The CI enables a more cautious interpretation of performance metrics, particularly for small datasets, as it reflects the lower and upper bounds of the possible performance values [19]. The formulations of SE and CI are presented in Eq. (9) and Eq. (10). In this analysis, the variance is also computed. The variance is used to measure the degree of variability or instability of the model performance across test data or multiple experimental runs, Eq. (11).

$$SE = \sqrt{\frac{p(1-p)}{n}} \tag{9}$$

$$CI = p \pm 1.96 \times SE \tag{10}$$

$$Var(p) = \frac{p(1-p)}{n} \tag{11}$$

Variable descriptions

- SE, Standard Error
- CI, Confidence Interval
- Var, variance
- p, denotes the accuracy value
- n, denotes the number of test samples
- 1.96 is the constant corresponding to the 95% confidence level

TABLE II. Stability Level of Classification

Scenario	Acc.	Var.	SE	CI
Image-Based CNN	0.86	0.00148	0.039	[0.79, 0.94]
Acoustic-Based CNN	0.80	0.00200	0.045	[0.71, 0.89]
Multimodal CNN	0.94	0.00073	0.027	[0.88, 0.99]

Based on the variance, SE, and CI values shown in Table II, the multimodal CNN exhibits the most stable and reliable performance. It achieves the lowest variance (0.00073) and SE (0.027), along with the narrowest CI range, indicating more consistent performance across test samples. Therefore, multimodal CNN can be considered the best-performing and most reliable model among the evaluated approaches. In contrast, the acoustic-based CNN shows the highest variance and SE, suggesting greater performance variability and lower stability. Pairwise, statistical significance between classifiers was assessed using McNemar's test [20] on the same test samples with continuity correction. Pairwise comparisons were conducted between the multimodal model and each unimodal baseline (image-based and acoustic-based) compute using Eq. (12) and data in Table III. In McNemar's test, n_{01} and n_{10} denote the number of discordant pairs where only one classifier produces a correct prediction, and the continuity-corrected chi-square statistic is computed to assess the significance of performance differences. If the Chi-Square χ^2.

$$\chi^2 = \frac{(n_{01}-n_{10})^2}{n_{01}+n_{10}} \tag{12}$$

TABLE III. Samples with False Prediction

Scenario	Samples with false prediction
Image-Based CNN	DAP47_03, DAP47_7, DAP47_12, DAP47_18, DAP47_19 DAP60_03, DAP60_08, DAP60_17, DAP60_19 DAP67_06, DAP67_11
Acoustic-Based CNN	DAP47_02, DAP47_05, DAP47_09, DAP47_14, DAP47_19, DAP47_20 DAP53_02, DAP53_08, DAP53_15 DAP60_01, DAP60_04, DAP60_17 DAP67_03, DAP67_08, DAP67_12, DAP67_17
Multimodal CNN	DAP53_02, DAP53_11, DAP53_15, DAP53_17, DAP53_18

The McNemar test statistic was compared with the chi-square critical value of 3.84 (df = 1, α = 0.05) to assess statistical significance. For the comparison between the Multimodal CNN and the Image-based CNN, $n_{01} = 11$ denotes samples misclassified by the image-based model but correctly classified by the multimodal model, while $n_{10} = 5$ denotes samples correctly classified by the image-based model but misclassified by the multimodal model. The resulting test statistic is χ^2=1.56, which is lower than the critical value

(3.84). In contrast, for the comparison between the Multimodal CNN and the Acoustic-based CNN, $n_{01} = 14$ and $n_{10} = 3$, yielding a test statistic of χ^2=5.88, which exceeds the critical value. These results indicate that the multimodal CNN significantly outperforms the acoustic-based CNN. whereas the improvement over the image-based CNN is not statistically significant. Nevertheless, it achieved an accuracy 8% higher than the image-based model.

IX. Conclusion

This study evaluated three different scenarios for audio-visual classification: image-based only, acoustic-based using MFCC features, and a combined multimodal CNN architecture. In all scenarios, data augmentation was applied to both image and audio data, with audio augmentation performed prior to the construction of the MFCC spectrogram.

The experimental results demonstrate that the multimodal approach, which integrates both image and acoustic features, consistently outperforms the single-modality models. Specifically, the multimodal CNN with data augmentation achieved the highest performance across all evaluation metrics, confirming the effectiveness of combining complementary feature representations. Furthermore, applying data augmentation on both modalities contributed to improved model generalization and robustness, highlighting the importance of augmentation strategies in multimodal learning tasks.

Based on statistical analysis, the multimodal CNN achieved the highest accuracy (0.94), outperforming the image-based CNN (0.86) and acoustic-based CNN (0.80). The 95% confidence interval of the multimodal model ([0.88, 0.99]) is narrower than those of the unimodal models, indicating higher reliability. Furthermore, the variance of the multimodal CNN (0.00073) is lower than that of the image-based (0.00148) and acoustic-based (0.00200) models, demonstrating more stable performance. Pairwise McNemar's tests indicate that the performance improvement of the Multimodal CNN over the acoustic-based CNN is statistically significant ($p < 0.05$), whereas the improvement over the image-based CNN is not statistically significant.

Overall, the findings suggest that leveraging multimodal data with appropriate augmentation techniques is a promising direction for improving classification accuracy in audio-visual applications. Future work may explore more advanced fusion techniques and augmentation methods to further enhance model performance.

Limitations and Future Work: The current study is limited by the small dataset size, controlled greenhouse conditions that do not reflect natural variability, and the lack of external validation. Future research should focus on collecting larger and more diverse datasets under real-world conditions to improve model generalizability and robustness.

Acknowledgment

This study and the research behind it would not have been possible without the exceptional support of my supervisor, Prof. Agus Buono, Dr. Karlisa Priandana, and Prof. Dwi Guntoro. Their enthusiasm, knowledge and exacting attention to detail have been an inspiration and kept my work on track from my first draft to the final revision of this study.

References

[1] Hamid A. S., Mohamed A., Jiang H., & Pen G. 2012. Applying Convolutional Neural Networks Concepts to Hybrid NN-HMM Model for Speech Recognition. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 4277-4280 25-30 March 2012, Kyoto, Japan.

[2] Hamid A. S., Mohamed A. R., Jiang H., Deng L., Pen G. & Yu D. 2014. Convolutional Neural Networks for Speech Recognition. IEEE/ACM Transactions on Audio, Speech, And Language Processing, VOL. 22, NO. 10, October 2014.

[3] Palaz D., Collobert R., dan Doss M.M. 2013. Estimating Phoneme Class Conditional Probabilities from Raw Speech Signal using Convolutional Neural Networks. arXiv preprint arXiv:1304.1018.

[4] Schlüter J. dan Böck S. 2014. Improved Musical Onset Detection With Convolutional Neural Networks. Proceedings of the 2014 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014).

[5] Khamparia A, Gupta D, Nguyen NG, Khanna A, Pandey B and Tiwari P. 2019. Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network," in IEEE Access, vol. 7, pp. 7717-7727, 2019, doi: 10.1109/ACCESS.2018.2888882..

[6] Jaiswal K and Patel DK. 2018. Sound Classification Using Convolutional Neural Networks. 2018 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bangalore, India, 2018, pp. 81-84, doi: 10.1109/CCEM.2018.00021.

[7] Maccagno A, Mastropietro A, Mazziotta U, Scarpiniti M, Lee YC, and Uncini A. 2020. A CNN Approach for Audio Classification in Construction Sites. In: Esposito A., Faundez-Zanuy M., Morabito F., Pasero E. (eds) Progresses in Artificial Intelligence and Neural Systems. Smart Innovation, Systems and Technologies, vol 184. Springer, Singapore. https://doi.org/10.1007/978-981-15-5093-5_33

[8] Jmour N, Zayen S, and Abdelkrim A. 2018. Convolutional Neural Networks for image classification. 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET), Hammamet, 2018, pp. 397-402, doi: 10.1109/ASET.2018.8379889

[9] Kausar A, Sharif M, Park J, and Shin DR. 2018. Pure-CNN: A Framework for Fruit Images Classification. 2018 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, USA, 2018, pp. 404-408, doi: 10.1109/CSCI46756.2018.00082.

[10] Sultana F., Sufian A, & Dutta P. 2018. Advancements in Image Classification using Convolutional Neural Network. 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 2018, pp. 122-129, doi: 10.1109/ICRCICN.2018.8718718.

[11] González A. A. & Santiago A. M. 2025. CNN-Based Road Event Detection Using Multiaxial Vibration and Acceleration Signals," *Applied Sciences*, vol. 15, no. 18, p. 10203, Sep. 2025. doi: 10.3390/app151810203.

[12] P. Thottempudi, V. Kumar, and R. Kumar. 2025. Dynamic multi-modal attention network for robust and real-time through-wall human activity recognition. Results in Engineering, vol. 28, p. 107632, Dec. 2025, doi: 10.1016/j.rineng.2025.107632.

[13] M. Cao, J. Wan, and X. Gu, "CLEAR: Multimodal Human Activity Recognition via Contrastive Learning Based Feature Extraction Refinement," *Sensors*, vol. 25, no. 3, p. 896, 2025, doi: 10.3390/s25030896

[14] C. Miron, A. Pasarica and R. Timofte. 2021. Efficient CNN Architecture for Multi-modal Aerial View Object Classification," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Nashville, TN, USA, 2021, pp. 560-565, doi: 10.1109/CVPRW53098.2021.00068.

[15] Ciresan CD, et al. 2011. Flexible, High Performance Convolutional Neural Networks for Image Classification. Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI'11), Vol. Two, Pages 1237-1242.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278-2324, November 1998.

[17] K. Doshi, 2021. Audio Deep Learning Made Simple: Sound Classification, Step-by-Step, *Medium*, Mar. 19, 2021. [Online]. Available: https://medium.com/data-science/audio-deep-learning-made-simple-sound-classification-step-by-step-cebc936bbe5

[18] Clara, 2011. Application of Mel Frequency Cepstrum Coefficients (MFCC) as Feature Extraction on Phoneme Recognition with Probabilistic Neural Network (PNN) as Classifier, Bachelor's thesis, IPB University, Bogor, Indonesia.

[19] T. Hastie, R. Tibshirani, and J. Friedman, 2013. An Introduction to Statistical Learning: With Applications in R, 1st ed. New York, NY, USA: Springer.

[20] Q. McNemar, 1947. Note on the sampling error of the difference between correlated proportions and percentages, Psychometrika, vol. 12, no. 2, pp. 153–157

Comparative Analysis of Machine Learning Based Algorithms for Predicting Injury Severity in Road Accidents

Soumaya AMRI[1], Mohammed AL ACHHAB[2], Mohamed LAZAAR[3]
Faculty of Sciences in Tetuan, Abdelmalek Essaadi University, Tetuan, Morocco[1, 2]
ENSIAS, Mohammed V University in Rabat, Rabat, Morocco[3]

***Abstract*—Road crash injury severity prediction is essential for intelligent transportation systems, yet challenged by severe class imbalance, rigid 4-class severity schemes (unhurt/slight/hospitalized/fatal), and optimal methodological selection. This study proposes a structured framework systematically evaluating four machine learning models—CatBoost, HistGradientBoosting, Random Forest, and SVM—across multiclass (native 4-class and ordinal wrapper), binary reduction (non-severe vs. severe), and oversampling techniques using crash data. Multiclass approaches reveal tree ensemble dominance but persistent rare severe class prediction difficulties. Binary class reduction substantially improves severe injury detection performance on this dataset by simplifying decision boundaries, while SMOTE oversampling provides algorithm-specific imbalance mitigation. Random Forest demonstrates the most stable binary performance across evaluation metrics, independent of oversampling strategies. This performance gain comes at the cost of reduced severity granularity compared to the original multiclass formulation. Overall, under imbalance-sensitive evaluation metrics, binary class reduction provides a pragmatic and operationally effective alternative to complex multiclass strategies for severe injury detection.**

Keywords*—Machine learning; imbalanced data; road accident; multiclass classification; binary classification; injury severity prediction

I. Introduction

Injuries, particularly among road users aged 15 to 64, result in significant mortality and long-term disability, ultimately reducing the workforce and weakening the economy. According to the International Road Assessment Program (iRAP), road traffic injuries contribute an estimated $1.85 trillion burden to the global economy each year [1].

In a world of rapidly advancing human mobility and a growing need for transport infrastructure, road safety has become a critical component of government initiatives aimed at reducing road user fatality rates.

The availability of large datasets and advancements in machine learning techniques provide powerful tools for analyzing traffic data and supporting the development of intelligent transportation systems before the implementation of new road safety strategies. However, predicting the severity of injuries resulting from road crashes is a complex challenge. Data collection, often conducted manually by police officers, frequently results in substantial inaccuracies and incomplete records. PIARC, the World Road association noted that a balance between the data collection burden and practical constraints is necessary to ensure police compliance in completing accident reports [2]. Furthermore, the contextual circumstances of each crash blur the line between rare events and standard classification tasks, requiring specialized techniques to manage data imbalance [3]. The factors influencing accident occurrence are highly intricate, further increasing the complexity and rendering it a non-linear classification problem [4]. Combined, these elements make the application of machine learning techniques in road safety research and intelligent transportation systems challenging at every stage of the prediction process.

Despite the substantial body of research on machine learning–based injury severity prediction, an important gap persists. Specifically, the combined influence of data imbalance, preprocessing strategies, and algorithm selection remains insufficiently examined. This limitation is particularly evident in large-scale, real-world crash datasets. While some studies focus on improving model architectures or evaluating different algorithms, others address data imbalance in isolation. However, few investigations examine the interaction between robust data preprocessing techniques and machine learning methods. Such integrated analysis is essential to enhance the accurate prediction of rare yet critical cases, including severe injuries in road accidents. This gap limits our understanding of how to effectively optimize the entire machine learning pipeline for imbalanced classification problems.

To address the challenges associated with road crash injury severity prediction, this study presents a critical review of related research. It also proposes a step-by-step prediction framework based on machine learning techniques and a real-world crash dataset. The objective is to establish a rigorous methodology for identifying a robust predictive model for injury severity.

The study further examines the challenges of training machine learning models on large, imbalanced datasets. It addresses multiclass injury severity prediction and systematically evaluates binary class reduction, nominal multiclass learning, and ordinal wrapper formulations. The goal is to identify the most suitable approach for similar applications.

The structure of the study is as follows: The second section reviews related works on predicting the severity of road crash injuries, with a particular focus on the application of machine

learning techniques in similar contexts. The third section describes the proposed prediction framework, detailing its architecture and key interactions. The fourth section explains the experimental evaluation of the proposed model. The fifth section discusses the performance results obtained from the experiments. Finally, the last section concludes the study, summarizing the main findings and contributions.

II. Related Work

Numerous significant studies have investigated injury severity in road crashes, outlining various methods to predict injury severity levels. These prediction methodologies include classical statistical models [5], Poisson regression [6], logistic regression analysis [7] and discriminant analysis [8]. They also encompass advanced artificial intelligence approaches, including neural networks [9], decision trees [10], random forest models [11], and ensemble learning techniques [3].

The selection of appropriate machine learning techniques during the prediction process is influenced by data characteristics (data size and quality), as well as the specific objectives of each case study.

To provide a systematic evaluation of prior work, Table I synthesizes the methodological and empirical characteristics of related injury severity prediction studies.

TABLE I. Synthesis of Research Papers on Crash Severity Modeling and Predictive Performance

Study	Dataset Size	Classes	Ratio of severe classes	Main ML Model	Best Global Model Results	Severe Injury Results
[3]	308641	4	0.8% (Severe injury and fatal)	Gradient Boosting	Accuracy= 0.82	F1 score = 0.21, Precision = 0.31, Recall = 0.16
[10]	2238	3 (Slight, Serious, Fatal)	41,91% (Serious Injuries =39.83% Fatalities= 2.08%)	Ordered Forest	Gmean = 0.47	Precision =0,50
[12]	770561	3 levels initially later integrated into 2 classes	14.5% (13.3% serious and 1.2% fatal injuries)	SSAE	Precision=0.80, Recall=0.85, F1=0.84	Precision=0.42, Recall=0.21, F1=0.28
[13]	54364	3 (Slight, Severe, Fatal)	2.49% (2.34% severe and 0.15% fatal)	2D-CNN	F1= 0.95	F1-score = 0.004 (Fatal)
[15]	5740	4 levels (Slight, Medium, Severe, Fatal)	9.69% (6.74% severe and 2.94% fatal)	Bayesian Network	Accuracy= 0.66	Not evaluated
[16]	900690	Binary	Not described	Random Forest , LightGBM	ROC-AUC scores = 0.93	Precision = 0.65
[17]	1620	3 (Fatal, Injury, No-injury)	4,75%	CART	Accuracy =0.67	Precision = 0.45 (Fatal)
[18]	8516	4 (Fatal, Hospitalized, Injured, and Damage-only)	65% (22.9% Fatal and 42.1% Hospitalized)	Random Forest	Accuracy = 0.73	Precision = 0.85 Recall = 0.44
[19]	3834	4 (Minor, Moderate, Severe, Fatal)	13% (7.6% Severe Injuries, 5.4%Fatal Injuries)	Adaboost, Random Forest	Accuracy = 0.87	Precision =0.27 Recall =0.48

A comparative analysis of model performance in cited studies [3],[10], [12], [13], [15]-[19] reveals substantial heterogeneity in dataset size, modeling strategies, and evaluation metrics, with no standardized methodological framework emerging. Moreover, severe injury prediction remains insufficiently examined, as reported performance often emphasizes global accuracy while overlooking minority-class detection.

The implementation of a generic method is complex, as it requires careful examination of several parameters to determine the most appropriate machine learning methodology.

The first criterion relates to the dataset size and the number of features; both the algorithm's performance and the available computational resources should be assessed based on these data characteristics. Additionally, techniques such as scaling and distributed computing can be considered to facilitate training on large datasets.

Another critical criterion involves the distribution of injury severity levels, as a significant challenge in road crash analysis is the presence of highly imbalanced data. In most case studies, severe injuries are less frequent than minor injuries, compelling researchers to adopt supplementary techniques to address imbalanced data challenges and improve road crash injury analysis.

In the crash-injury severity literature, three modeling strategies are commonly used: binary reduction, nominal multiclass learning, and ordinal multiclass modeling. Binary reduction is often applied to mitigate severe class imbalance and stabilize model training [20], [21]. However, it reduces the granularity required for prioritizing safety interventions. Recent studies emphasize preserving richer severity levels whenever possible [22].

Nominal multiclass methods, particularly tree ensembles and gradient boosting models such as XGBoost, LightGBM, and CatBoost, often achieve strong overall performance. When combined with resampling techniques like SMOTE-NC or class weighting, their results improve [23]. However, these models may confuse adjacent severity levels because they do not consider label order.

To address this limitation, ordinal approaches explicitly encode the inherent rank structure from minor to fatal injuries. These methods include cumulative-threshold decomposition and parametric ordinal regression models. They reduce distant misclassifications and enhance policy and clinical

interpretability. Empirical studies report improved performance over nominal classifiers, especially when combined with imbalance handling and modern learners [24]. Recent advances in dynamic ensemble selection and interpretable ensemble methods further improve robustness in imbalanced multiclass settings. Overall, evidence supports retaining multiclass severity and adopting ordinal-aware learning with appropriate rebalancing to preserve information and respect outcome ordering [23].

The following provides a comparative analysis of the machine learning models employed in research related to predicting the severity of road crash injuries. It outlines their advantages and limitations.

Neural network models present numerous advantages, particularly their ability to capture complex nonlinear relationships between independent and dependent variables and their effectiveness in scaling to large datasets. This capability makes them particularly suitable for extensive road crash databases. However, a significant drawback is the potential for overfitting, which can compromise prediction accuracy. A study on traffic accident severity in Madrid employed a framework based on one- and two-dimensional convolutional neural networks [13]. The findings highlighted the framework's scalability and its rapid prediction capabilities, making it suitable for real-time applications. Nonetheless, the precision scores for fatal and serious injuries were notably low, primarily due to the influence of imbalanced data, which adversely affected the prediction outcomes for these injury categories.

The Bayesian network, evaluated on a medium-sized dataset of 5740 instances and four injury classes, achieved the highest overall accuracy on both the training and testing sets. Its performance was superior to that of the Exhaustive Chi-Square Automatic Interaction Detector tree and the Linear Support Vector Machine models [15].

Wahab explores various machine-learning algorithms to predict motorcycle crash severity in Ghana, utilizing a medium-sized dataset consisting of 8,000 instances with 14 features. The Random Forest model exhibited greater accuracy than the other machine learning models tested. However, its individual accuracy for each injury severity class, especially for fatal injuries, was notably low [18]. A comparative analysis of Decision Trees, RF, and Naïve Bayes, performed on a reduced dataset using combined feature selection techniques (including two-way ANOVA, regression analysis, and chi-square), demonstrated that RF achieved the best performance [14]. Zhang recommends employing the Ordered Forest algorithm for analyzing injury severity in single bicycle crashes, as it showed better predictive performance compared to the traditional Random Forest model [10].

AdaBoost and Gradient Boosting models offer notable advantages for crash severity prediction. These advantages stem primarily from their capacity for global sensitivity analysis, which enables the assessment of both individual and combined effects of influencing factors [3]. Jiang applied these two boosting algorithms to predict crash injury severity and reported overall accuracy superior to that of other models. However, the prediction performance for minority classes, particularly fatal and severe injuries, remained low. Jiang recommends using the F1-score and monetized evaluation metrics to better assess model performance in imbalanced data settings.

Conducted on a large road crash dataset from France, the LightGBM model achieved the highest F1-score and accuracy for predicting non-severe injuries, which represent the majority class. In contrast, the Random Forest model obtained the highest precision for predicting severe injuries, the minority class [16].

Abdulazeez evaluated several machine-learning algorithms to predict child occupant crash injury severity in Abu Dhabi. Adaboost, Bagging REP, ZeroR, and XGBoost achieved the best overall accuracy both before and after data balancing, even without feature selection. However, the prediction results for the severe injury class, categorized as the minority class, remained low, except for XGBoost and CatBoost, which yielded better performance after data balancing and class consolidation [19].

Thus, the results vary significantly depending on the research methodology and the specific characteristics of the datasets used. Random Forest, Adaboost, and Gradient Boosting models demonstrated the highest overall accuracy on small to medium-sized datasets. However, the highest prediction accuracy for the severe injury class, identified as the minority class, was achieved using RF, stacking, and Support Vector Machine models, respectively, though the accuracy values remain relatively low. While reducing the number of classes can improve accuracy metrics, addressing data imbalance may lead to overfitting in certain algorithms. The performance of these models should be further evaluated on large datasets to assess their robustness in real-time operations.

III. METHODOLOGY

This study addresses gaps and limitations in the literature on severe injury prediction. These include data collection inconsistencies, low performance for the severe injury minority class, and challenges in selecting appropriate machine learning models.

To overcome these issues, a step-by-step methodology is proposed, as illustrated in Fig. 1.

The first step involves data preparation issues, from data collection and integration to data cleaning.

The second step addresses data engineering tasks, including feature transformations and data exploration, through four key dimensions: temporal and atmospheric conditions, road characteristics, vehicle characteristics, and user profiles.

The third step comprises a structured data processing phase in which the machine learning methodology is selected via a comparative analysis of four experiments. These experiments target two core challenges in multiclass injury severity prediction: selecting effective techniques to handle class imbalance and assessing how class grouping strategies influence predictive performance on minority classes.

Experiments N1–N2 retain the original four-class injury severity scale. Experiment N1 evaluates candidate models on the unmodified four-class dataset (nominal multiclass), while experiment N2 introduces an ordinal classification framework that explicitly encodes the ordered nature of the four severity levels.

Experiments N3–N4 adopt a binary formulation (severe vs non-severe). Experiment N3 trains the same models on the original imbalanced data to quantify the impact of multiclass reduction without correcting the imbalance. Experiment N4 applies SMOTE within each training fold of the cross-validation procedure. The data are balanced using oversampling performed only on the training subsets. The validation folds remain untouched, thereby preventing data leakage. Multiple classifiers are then evaluated on the resampled training data. Comparing these settings reveals the relative benefits of multiclass reduction and data balancing for severe injury detection and highlights which algorithms benefit most from SMOTE-like resampling.

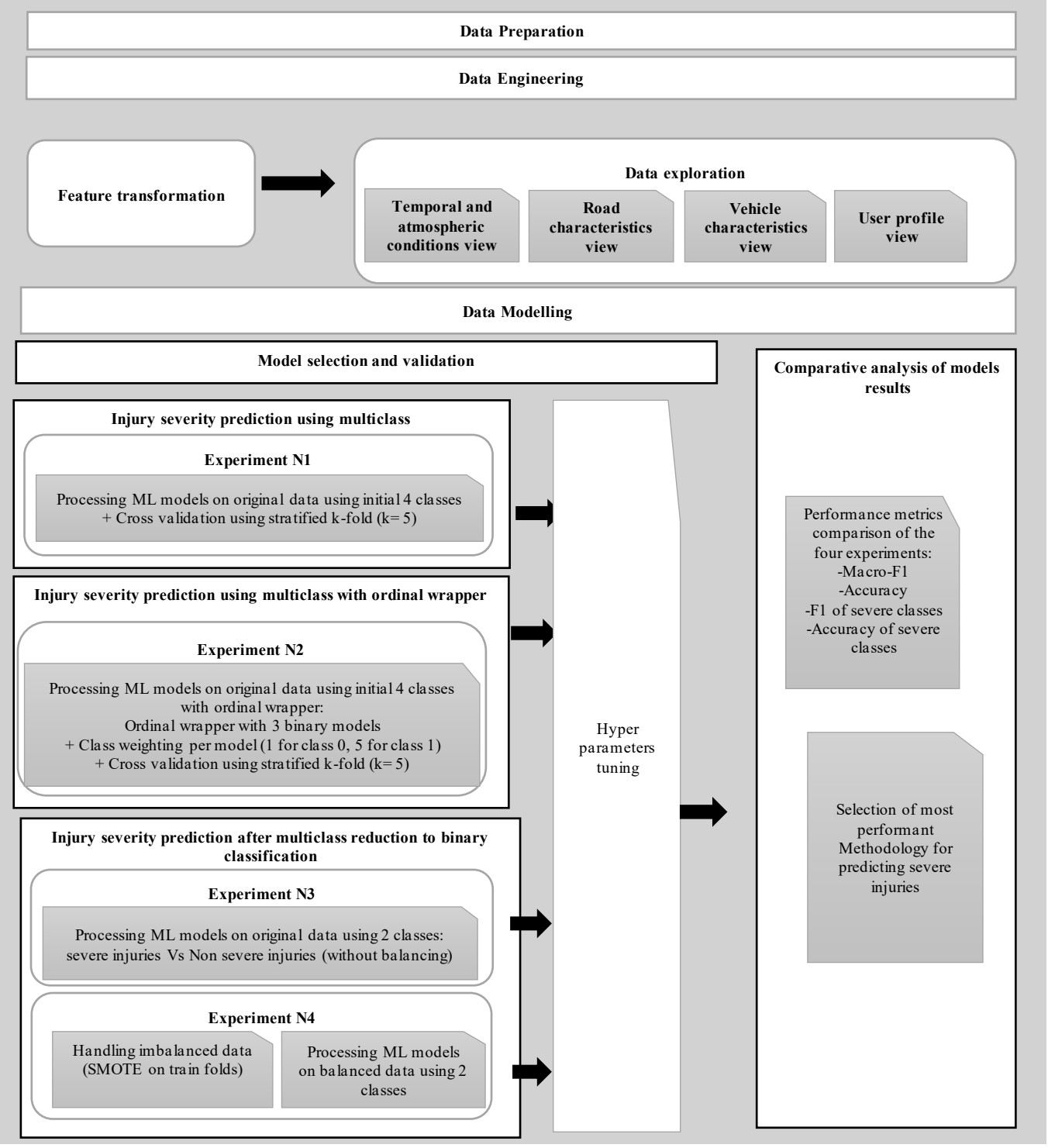


Fig. 1. Framework of the proposed prediction methodology.

Jointly analyzing results across all four experiments enables the identification of the most effective combination of model, imbalance-handling strategy, and class representation scheme for accurately predicting severe injuries in highly imbalanced road safety data.

IV. EXPERIMENTATION

A. Data Description

This research is conducted on real data sourced from the annual databases of road traffic accidents maintained by the French National Interdepartmental Observatory of Road Safety.

The annual databases extracted from the Bodily Accident Analysis Report [25] encompass all bodily traffic accidents occurring in metropolitan France, as well as in overseas departments and other overseas territories. This study conducted experiments using datasets from the years 2005 to 2020. The recorded data on road traffic accidents contains multiple layers of information, including features related to crash characteristics, location, involved vehicles, and road users.

To create the input dataset for this study, tables were joined based on the foreign keys specified in each data file, resulting in a comprehensive global dataset. Following the integration of 64 data files, corresponding to four files for each year, the final dataset comprised a total of 2,380,573 entries and 57 features, which served as the input data for this research.

B. Data Cleaning

In classification tasks, data cleaning focuses on enhancing classification performance by improving the quality of the training data. Common data cleaning techniques include outlier removal [26], missing value imputation [27], normalization and standardization [28], error detection and correction, handling duplicate values, noise reduction [29] and categorical data handling [30].

In this study, the initial step involved merging the collected datasets before applying data cleaning techniques. Error detection and correction were performed by casting column types (date, numeric, string, etc.) and correcting incorrect value formats for numerical columns as well as erroneous entries in categorical columns. Missing value imputation techniques were employed to address null values. A percentage count of missing values for each feature was generated and classified to identify unusable columns or records.

To avoid imputation of non-representative data for certain features, fixed intervals were established for handling missing data, as outlined in Fig. 2, which describes the data cleaning process and corresponding actions. Columns with over 89.9% missing values were removed, except in cases where these columns contained information specific to certain types of road users, such as pedestrians. In such cases, null values are not indicative of missing data but rather signify that this information is irrelevant to other types of road users. For instance, null values in the columns “locp” (pedestrian location) and “etatp” (pedestrian status, e.g. alone, accompanied) are not considered missing data for records of crashes involving only vehicles and no pedestrians.

Records with less than 5% missing data in any column were deleted. For columns with between 5% and 89.9% missing values, various imputation methods were applied: mean imputation was used for numerical data, mode imputation for categorical data, and, in some cases, imputation by deduction based on a data dictionary. This latter method was applied to categorical features where undefined values could be reasonably inferred based on similar meanings within the data.

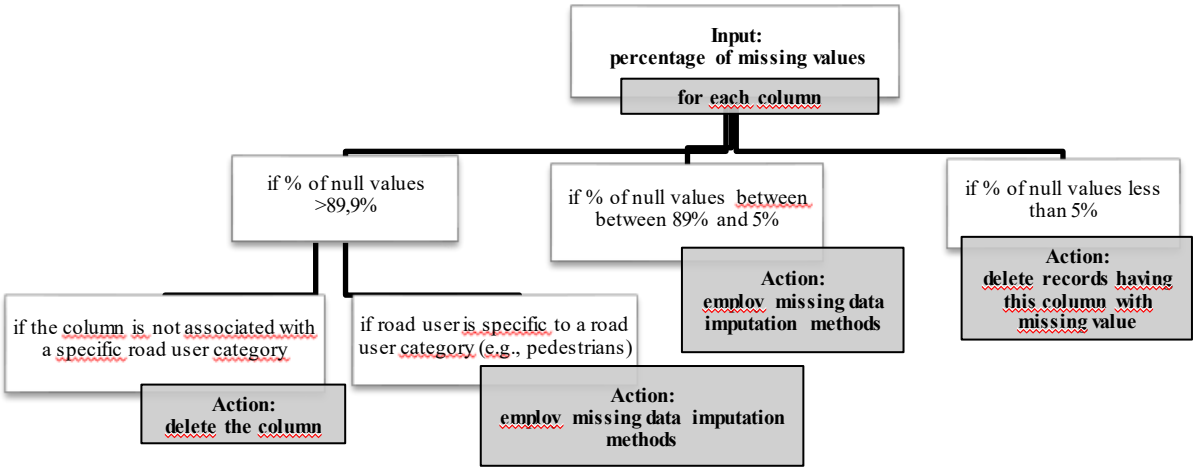


Fig. 2. Data cleaning process and actions.

C. Data Engineering

1) Feature transformation: To prepare the data for exploratory analysis, feature transformations were conducted, including the formatting of existing features and the generation of new ones. Eight additional features were derived from the columns "date," "time," and "user date of birth" to explicitly represent embedded information: "time slice", "year", "month", "day", "day of week", "is holiday", "age", and "age slice". The final task involved encoding the newly created categorical features to complete the data preparation and transformation process.

2) Exploratory data analysis: To analyze crash severity, an exploration of the distribution of accidents based on the injury severity level feature is conducted. This feature is categorized

into four classes: unhurt, slightly injured, hospitalized injuries, and fatalities. Fig. 3 illustrates the distribution of injury severity levels. It demonstrates that fewer than 2% of road users fall into the killed category and 20% into hospitalized due to injury. These two categories are identified as minority classes within the dataset.



Fig. 3. Distribution of crashes by injury severity level.

3) Data modelling: The injury severity feature, originally comprising four levels (1=Unhurt, 2=Fatal, 3=Hospitalized, 4=Slight injury), was systematically remapped in Experiment N2. An ordinal scale was adopted to preserve the hierarchical relationships: 0 = Unhurt < 1 = Slight injury < 2 = Hospitalized < 3 = Fatal. In experiments N3 and N4, the four levels of injury severity are reduced to two classes: severe (fatal and hospitalized) and non-severe (unhurt and slight injury).

The methodological framework structures four complementary experiments with differentiated modeling strategies:

- Experiment N1: Multiclass prediction (4 ordinal classes) on original data.
- Experiment N2: Ordinal wrapper (3 binary classifiers) on original 4-class data with class weighting.
- Experiment N3: Binary classification on original imbalanced data.
- Experiment N4: Binary classification on SMOTE-balanced data (Non Severe vs Severe Injury).

This progression systematically evaluates binary vs multiclass formulations alongside data-level and algorithmic imbalance mitigation strategies.

4) Handling imbalanced data: Several machine learning studies address the issue of imbalanced data and attempt to mitigate its impact on the performance of models. Undersampling and oversampling methods, which aim to create equal amounts of data points for each class, are commonly employed to manage this imbalance. Subsequently, various developed models have been proposed. Galar categorizes techniques for handling imbalanced data into four types: algorithmic-level, data-level, cost-sensitive, and ensembles of classifiers [31].

In this study, class imbalance (Class "unhurt": 41%, Class "fatal": 2.7%) was addressed through complementary strategies across experiments, following Galar's established taxonomy:

- Experiments N1-N2: Algorithmic-level mitigation via stratified k-fold cross-validation (k=5) preserves original distribution while optimizing for rare classes.
- Experiment N2: Cost-sensitive ordinal wrapper implements class_weights= [1.0, 5.0] per binary classifier P(Y>k), assigning fivefold misclassification penalty to severe cases (>k threshold).
- Experiment N3: data-level evaluation using binary classification on original imbalanced data to establish the performance of multiclass reduction.
- Experiment N4: SMOTE oversampling applied to training folds generates synthetic minority class instances via k-nearest neighbor interpolation, proven effective for traffic crash severity prediction [32], [33]. It generates new synthetic instances of the minority class by interpolating values between the two nearest data points along a straight line [34].

This multi-strategy framework systematically compares data-level oversampling (SMOTE), cost-sensitive ordinal learning, and binary class reduction strategies for severe injury prediction.

5) ML modelling:

a) Model selection: In this work, the dataset is both large and highly imbalanced. In similar case studies, boosting algorithms, such as Gradient Boosting and the Random Forest algorithm, have demonstrated superior performance compared to other models. Therefore, four machine-learning algorithms are evaluated to analyze the severity level of injuries caused by road crashes: CatBoost, HistGradientBoosting Classifier, Random Forest, and Linear Support Vector Machine (SVM). The SVM model was selected for its well-documented effectiveness in classification tasks.

b) Hyperparameter tuning: Model hyperparameters were systematically optimized using stratified k-fold cross-validation (k=5, reduced to k=3 for 2000 iterations) with early stopping, following the methodological framework. Iteration counts were evaluated across {500, 800, 1000, 2000}, with Macro F1 serving as the primary optimization criterion across all experiments (N1-N4) and models. Optimal iterations were selected independently for each model-experiment combination, ensuring configuration-specific performance maximization while preventing overfitting. Cross-validation improvement curves demonstrating macro-F1 gains versus iteration count, along with final selected hyperparameters per model, are detailed in section V.

6) Evaluation metrics: To evaluate the performance of machine learning models, various metrics are employed by researchers. Accuracy, along with precision, recall, and F1-score, is among the most commonly used evaluation metrics.

In injury severity prediction studies, all of the metrics mentioned previously are employed. However, relying solely on the overall average prediction accuracy for highly imbalanced data can lead to biased interpretations of the results. Jiang suggests using additional metrics, such as the F1 score for each class and the monetized measurement (MM), which represents the weighted average of the F1 scores for each class [35].

In this study, Macro-F1 constitutes the primary evaluation metric across all four experiments, computing the unweighted average of per-class F1-scores to ensure equitable minority class contribution despite severe imbalance. This metric robustly penalizes poor severe injury detection masked by majority class dominance, representing the established standard for multiclass imbalanced traffic safety prediction [36].

A comprehensive set of evaluation metrics was applied consistently across all experiments. For each configuration (Experiments N1–N4), model performance was primarily assessed using macro-F1. This was complemented by per-class precision, recall, and F1-scores, as well as weighted F1 and overall accuracy. These metrics were used to capture both global and class-specific behavior under class imbalance.

For Experiment N2, which models the ordinal nature of injury severity, quadratic weighted kappa was additionally employed. It was implemented using the cohen_kappa_score function with quadratic weights. This metric quantifies agreement beyond chance and penalizes distant misclassifications more strongly than adjacent ones. Such a property is particularly appropriate for hierarchical severity scales.

This unified evaluation framework enables comparison across Experiments N1–N4. It supports the selection of the most appropriate methodology for severe injury prediction by jointly considering overall predictive performance and the accurate identification of clinically critical yet rare cases.

V. Results and Discussion

This study primarily aims to examine the challenges in predicting road crash injury severity, particularly for minority classes, and to identify the most effective machine learning approach for accurate prediction. To this end, hyperparameter tuning results are first analyzed for selected models across each experiment to determine optimal parameters. Model performance is then evaluated individually within the four experiments, followed by a comprehensive comparison that highlights strengths, weaknesses, and the superior methodology for injury severity prediction.

A. Hyperparameter Tuning Results

To ensure optimal model configuration across all four experiments and algorithms, hyperparameters were tuned using stratified cross-validation and early stopping, following the methodological framework outlined in Fig. 1.

1) CatBoost hyperparameter evaluation: For the CatBoost model in experiment N1 (initial 4-class configuration), hyperparameter tuning focused on varying the number of iterations (500-2000) while fixing learning_rate=0.05 and depth=6.

Fig. 4 presents the macro F1-score evolution across iteration counts. Optimal performance was achieved with iterations=2000, learning_rate=0.05, depth=6 (F1_macro = 0,648), representing the best configuration identified through systematic grid evaluation.

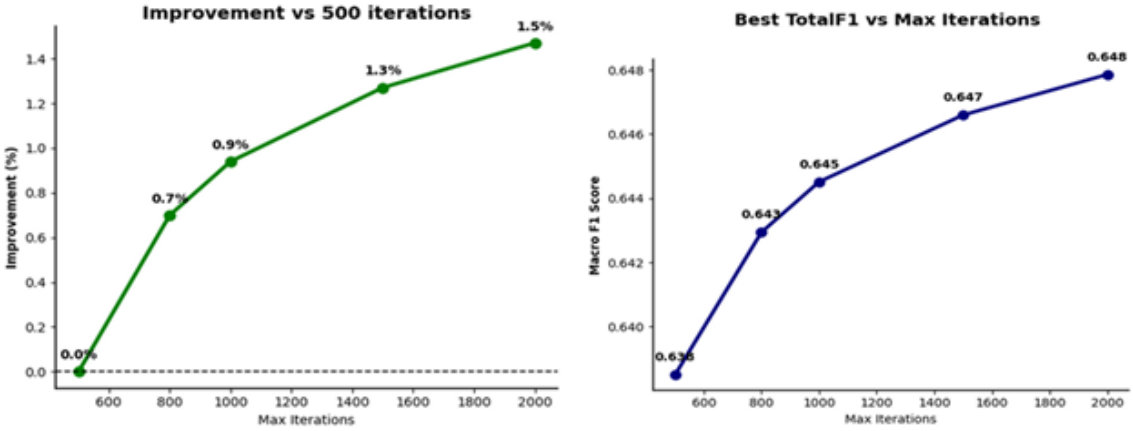


Fig. 4. Macro F1 evolution of experiment N1 across iterations.

For subsequent experiments, the same hyperparameter tuning procedure was applied to identify the optimal configuration.

2) Histgradient boosting classifier hyperparameter evaluation: For the HistGradientBoostingClassifier, hyperparameter tuning was conducted using RandomizedSearchCV with 12 iterations and 5-fold cross-validation, systematically exploring combinations of learning_rate (0.05-0.20), max_iter (100-300), max_leaf_nodes (15-31), min_samples_leaf (10-20), and max_depth (3-5 or None).

Fig. 5 presents the macro F1-score performance across hyperparameter ranges applied in experiment N1 using the initial four classes of injury severity. Optimal performance was achieved with min_samples_leaf=20, max_leaf_nodes=31, max_iter=100, max_depth=5, learning_rate=0.2 (F1_macro = 0.484, std=0.013), representing the best configuration identified through randomized search.

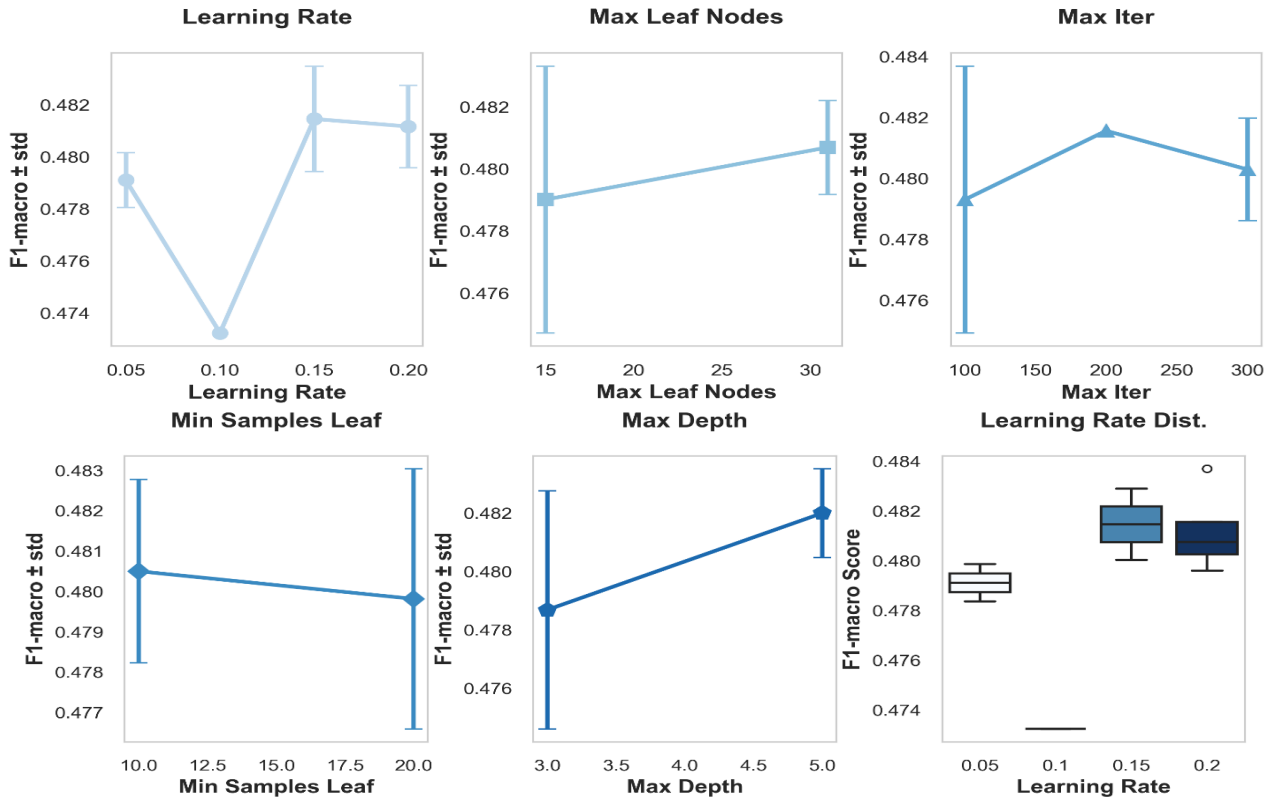


Fig. 5. Performance of hyperparameter tuning of the histgradient boosting classifier on Macro F1-score.

3) Random forest hyperparameter evaluation: A systematic grid search using cross_val_score (3-fold CV, F1_macro scoring) exhaustively evaluated eight configurations of RandomForestClassifier, focusing on n_estimators ∈ {100, 200} and max_depth ∈ {8, 12}, with fixed min_samples_split=5, min_samples_leaf=5, max_features=0.3 and using a balanced class_weight. The optimal configuration (n_estimators=100, max_depth=12) achieved the highest mean CV score, as confirmed by the hyperparameter validation curve presented in Fig. 6.

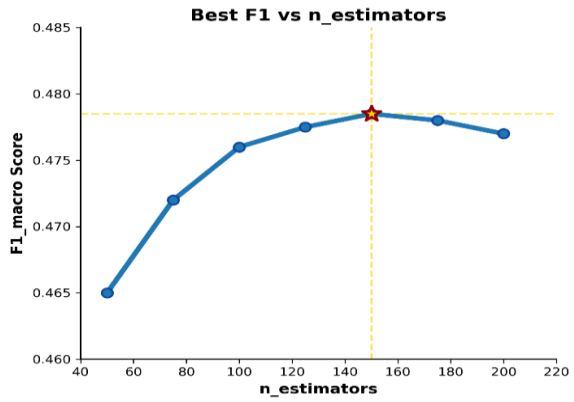


Fig. 6. Impact of RF hyper-parameter tuning of Macro-F1.

4) SVM hyperparameter evaluation: Hyperparameter tuning of the SVM classifier was conducted using RandomizedSearchCV with 10 iterations and 3-fold cross-validation, optimizing for the F1-macro score across four classes on a balanced dataset (class_weight='balanced'). The search explored a constrained parameter space comprising C values [0.1, 1, 10, 50, 100], RBF kernel with gamma values ['scale', 0.01, 0.1, 1.0], and fixed degree=3, completing 30 fits efficiently in a single-threaded configuration (n_jobs=1).

The optimal configuration (C=100, kernel='rbf', gamma=0.01, degree=3) achieved a cross-validated F1-macro score of 0.172 as presented in Fig. 7. This result indicates substantial challenges in multiclass classification, likely due to class imbalance and limited feature separability in the RBF-induced feature space.

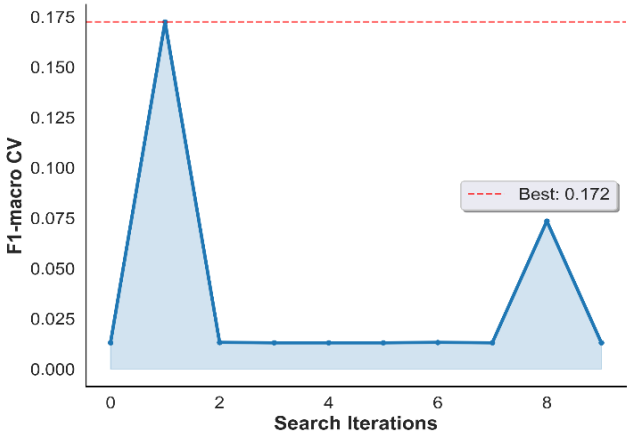


Fig. 7. SVM Hyperparameter tuning convergence.

B. Models Performance Analysis

1) Multiclass classification: The first two experiments within this methodological framework evaluate model performance using the initial four-class injury severity scheme. The second experiment incorporates an ordinal wrapper approach; together, these experiments compare model performance between native class weighting and ordinal wrapper configurations.

Table II presents performance metrics for CatBoost, HistGradientBoosting, Random Forest, and SVM models. The primary evaluation metrics, in order of priority, are macro F1-score, accuracy, and precision for severe classes (Hospitalized and Fatal).

TABLE II. MULTICLASS CLASSIFICATION PERFORMANCE OF EXPERIMENT N1 (ORIGINAL 4-CLASS) VS. EXPERIMENT N2 (ORDINAL WRAPPER)

Model		Injury severity level	Experiment N1: Original multiclass			Experiment N2: Ordinal wrapper multiclass		
	Metric		*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*
Catboost		Unhurt	0.65	0.90	0.75	0.81	0.59	0.68
		Slight	0.67	0.47	0.55	0.49	0.38	0.43
		Hospitalized	0.50	0.44	0.47	0.35	0.75	0.47
		Fatal	0.54	0.03	0.06	0.29	0.18	0.22
	Accuracy				0.63			0.53
	Macro avg		0.59	0.46	0.46	0.48	0.47	0.45
	Weighted avg		0.63	0.63	0.61	0.59	0.53	0.54
Hist-gradient Boosting classifier		Unhurt	0.67	0.87	0.76	0.70	0.83	0.76
		Slight	0.70	0.34	0.46	0.64	0.26	0.37
		Hospitalized	0.40	0.43	0.41	0.30	0.32	0.31
		Fatal	0.16	0.52	0.24	0.11	0.65	0.19
	Accuracy				0.58			0.52
	Macro avg		0.48	0.54	0.47	0.44	0.52	0.41
	Weighted avg		0.62	0.58	0.57	0.59	0.52	0.52
Random Forest		Unhurt	0.71	0.77	0.74	0.46	0.91	0.61
		Slight	0.65	0.40	0.49	0.38	0.11	0.17
		Hospitalized	0.36	0.28	0.31	0.44	0.11	0.17
		Fatal	0.10	0.70	0.18	0.20	0.20	0.20
	Accuracy				0.54			0.44
	Macro avg		0.46	0.54	0.43	0.37	0.33	0.29
	Weighted avg		0.61	0.54	0.55	0.42	0.44	0.35
SVM		Unhurt	0.41	0.88	0.56	0.52	0.15	0.23
		Slight	0.04	0.07	0.05	0.38	0.42	0.40
		Hospitalized	0.16	0.05	0.08	0.21	0.48	0.29
		Fatal	0.00	0.00	0.00	0.03	0.05	0.04
	Accuracy				0.38			0.31
	Macro avg		0.15	0.25	0.17	0.28	0.27	0.24
	Weighted avg		0.20	0.38	0.25	0.40	0.31	0.30

2) Binary classification: Experiments N3 and N4 implement class reduction strategies to assess their impact on prediction performance. Experiment N3 collapses the four-class severity scheme into a binary classification framework (non-severe vs. severe). Experiment N4 introduces SMOTE oversampling as an additional balancing technique to evaluate its incremental benefit across the four models examined (CatBoost, HistGradientBoostingClassifier, Random Forest, and SVM).

Binary classification, as presented in Table III, reveals SMOTE's differential efficacy across algorithms: HistGradientBoosting shows a +13% improvement in accuracy (0.70 → 0.79) while the SVM achieves a substantial +169% increase in the severe-class F1-score (0.16→0.43). These results highlight the necessity of resampling for imbalance-sensitive architectures. In contrast, Random Forest and CatBoost demonstrate intrinsic robustness, maintaining consistently high performance (accuracy=0.83, ΔF1-severe < +2%).

Random Forest emerges as the overall top performer in binary classification (accuracy=0.83, F1-severe=0.50), exhibiting superior generalization across severity levels.

TABLE III. BINARY CLASSIFICATION PERFORMANCE OF EXPERIMENT N3 (CLASS REDUCTION) VS. EXPERIMENT N4 (WITH SMOTE OVERSAMPLING)

Model		Injury severity level	Experiment N3: Reduced to binary class			Experiment N4: Reduced to binary class + SMOTE balancing		
	Metric		*Precision*	*Recall*	*F1-score*	*Precision*	*Recall*	*F1-score*
Catboost		Non severe	0.85	0.95	0.90	0.86	0.94	0.90
		Severe	0.66	0.36	0.47	0.63	0.39	0.48
	Accuracy				0.83			0.83
	Macro avg		0.75	0.66	0.68	0.74	0.67	0.69
	Weighted avg		0.81	0.83	0.81	0.81	0.83	0.81
Hist-gradient Boosting classifier		Non severe	0.94	0.66	0.78	0.88	0.86	0.87
		Severe	0.39	0.84	0.54	0.50	0.55	0.53
	Accuracy				0.70			0.79
	Macro avg		0.67	0.75	0.66	0.69	0.71	0.70
	Weighted avg		0.83	0.70	0.73	0.80	0.79	0.80
Random Forest		Non severe	0.86	0.94	0.90	0.86	0.92	0.89
		Severe	0.63	0.41	0.50	0.60	0.44	0.51
	Accuracy				0.83			0.82
	Macro avg		0.74	0.67	0.70	0.73	0.68	0.70
	Weighted avg		0.81	0.83	0.81	0.81	0.82	0.81
SVM		Non severe	0.81	0.98	0.88	0.89	0.58	0.70
		Severe	0.50	0.10	0.16	0.31	0.71	0.43
	Accuracy				0.79			0.61
	Macro avg		0.65	0.54	0.52	0.60	0.65	0.57
	Weighted avg		0.7	0.79	0.73	0.77	0.61	0.65

C. Overall Performance Comparison

The proposed methodology systematically evaluates four machine learning models—CatBoost, HistGradientBoosting, Random Forest, and SVM—across multiclass injury severity prediction using an initial four-class scheme (Unhurt, Slight, Hospitalized, Fatal).

Experiment N1 applies native multiclass classification using stratified k-fold cross-validation. The results indicate moderate overall performance, primarily driven by tree-based ensemble methods. CatBoost achieves the highest accuracy (0.63) and macro-F1 score (0.46). However, all models show significant weaknesses in predicting rare severe classes, particularly the fatal category, with F1-scores below 0.25.

Experiment N2 introduces an ordinal wrapper with class weighting to account for injury severity's inherent ordering, yielding mixed outcomes. While recall for Hospitalized cases improves notably (e.g., CatBoost: 0.44 to 0.75), overall accuracy and Macro F1 decline slightly (CatBoost: 0.63 to 0.53; 0.46 to 0.45). The lower overall performance of the ordinal formulation may be partly influenced by class imbalance rather than reflecting an inherent limitation of the ordinal modeling framework. These results underscore the wrapper's limited efficacy in highly imbalanced multiclass settings and highlight the trade-offs of enforcing ordinal constraints without fully resolving class distribution skew.

Binary class reduction in experiment N3, collapsing the four classes into non-severe versus severe, markedly enhances predictive performance across all models, demonstrating the methodology's sensitivity to classification complexity. Accuracies increase to 0.79–0.83 for tree-based ensembles, compared to 0.38–0.63 in the multiclass setting. Severe-class F1-scores improve substantially (e.g., Random Forest reaches 0.50 from previous multiclass lows). This improvement results from a simplified decision boundary that reduces overfitting to the dominant unhurt and slight injury classes and increases focus on clinically relevant severe outcomes.

This shift validates the strategic class aggregation within the framework. The approach reduces modeling complexity while preserving prognostic utility. However, it sacrifices granular severity distinctions that are essential for detailed risk stratification.

Experiment N4 further addresses residual imbalance via SMOTE oversampling on training folds, producing algorithm-specific gains that affirm its value for imbalance-sensitive models. HistGradientBoosting accuracy rises 13% (0.70 to 0.79) with stable severe F1 (0.54 to 0.53), while SVM exhibits the most dramatic uplift in severe F1 (0.16 to 0.43, +169%), reflecting SMOTE's ability to augment synthetic severe minorities for linear classifiers. Conversely, intrinsically robust tree-based models such as Random Forest and CatBoost exhibit negligible performance variations (accuracies ~0.83 to 0.82;

severe F1 +2% max). These results indicate baseline resilience to class imbalance.

Within the framework's comparative validation, the binary Random Forest configuration emerges as the most suitable methodology for severe injury prediction.

Table IV quantifies performance uplifts across experiments for each model, using accuracy and severe F1-score (Hospitalized+Fatal multiclass; Severe binary) as key metrics.

In summary, the Random Forest model in the binary configuration (Experiment N3) emerges as the superior approach. It achieves the highest accuracy (0.83), a macro-F1 score of 0.70, a severe-class F1-score of 0.50, and a non-severe F1-score of 0.90. The model demonstrates exceptional stability across configurations, as shown by minimal performance degradation after SMOTE (-1.2% accuracy). It also records the largest improvement from multiclass to binary settings (+53.7% accuracy, +61.1% severe F1-score).

TABLE IV. PERFORMANCE IMPROVEMENT PERCENTAGES ACROSS EXPERIMENTAL CONFIGURATIONS

Model	Metric	Multiclass Exp. 1 → Binary Exp. 3 (%)	Binary Exp. 3 → SMOTE Exp. 4 (%)
CatBoost	Accuracy	+31.7% (0.63 → 0.83)	0.0% (0.83 → 0.83)
	Severe F1	+2.1% (0.27 → 0.47)*	+2.1% (0.47 → 0.48)
HistGradientBoosting	Accuracy	+20.7% (0.58 → 0.70)	+12.9% (0.70 → 0.79)
	Severe F1	+31.7% (0.33 → 0.54)*	-1.9% (0.54 → 0.53)
Random Forest	Accuracy	+53.7% (0.54 → 0.83)	-1.2% (0.83 → 0.82)
	Severe F1	+61.1% (0.31 → 0.50)*	+2.0% (0.50 → 0.51)
SVM	Accuracy	+107.9% (0.38 → 0.79)	-22.8% (0.79 → 0.61)
	Severe F1	+162.5% (0.08 → 0.16)*	+168.8% (0.16 → 0.43)

* Multiclass severe F1 approximated as avg(Hosp+Fatal).

This robustness stems from its intrinsic bagging mechanism, which effectively handles imbalance and feature interactions without relying on synthetic oversampling, outperforming even CatBoost's multiclass strengths and SVM's SMOTE-dependent gains.

VI. CONCLUSION

This study systematically evaluated machine learning models for predicting road crash injury severity within a structured methodological framework encompassing multiclass (original 4-class and ordinal wrapper), binary class reduction, and SMOTE oversampling configurations.

Multiclass experiments highlighted the dominance of tree-based ensemble methods, with CatBoost achieving the highest accuracy (0.63) and macro-F1 score (0.46). However, persistent difficulties in predicting rare severe classes emphasized the impact of class imbalance. Ordinal wrappers produced mixed improvements in recall, but often at the expense of overall performance metrics.

Binary class reduction substantially improved model performance across algorithms. Accuracies increased by 20–108%, and severe-class F1-scores improved by up to 162%. This strategy simplified decision boundaries while emphasizing clinically relevant severe outcomes. SMOTE further enhanced performance, particularly for linear classifiers such as SVM, with severe F1-score gains of up to 169%.

Random Forest with binary classification emerged as the optimal methodology, exhibiting unmatched stability across binary setups, validating the framework's comparative validation stage and affirming class aggregation's efficacy for imbalanced severity prediction.

These findings advance road safety analytics by demonstrating how strategic classification reduction outperforms complex ordinal modeling, enabling reliable severe injury flagging for proactive interventions in intelligent transportation systems.

However, limitations persist: the binary framing of the selected Random Forest model sacrifices multiclass granularity, potentially hindering fine-grained severity triage, while cross-validation results require external validation on diverse datasets to confirm generalizability beyond the current cohort. Future enhancements should incorporate explainability techniques (e.g., SHAP) to elucidate severe prediction drivers and bridge performance with clinical interpretability. Additionally, research should explore deep learning ensembles and multi-regional crash datasets.

Ultimately, deploying binary Random Forest models within real-time ITS frameworks holds promise for mitigating severe crash outcomes through data-driven risk prioritization

REFERENCES

[1] World Bank, "THE HIGH TOLL OF TRAFFIC INJURIES: Unacceptable and Preventable," 2017.

[2] World Road Association (PIARC), "ROAD SAFETY MANAGEMENT," in ROAD SAFETY MANUAL A GUIDE FOR PRACTITIONERS, in Version 4. , 2024. [Online]. Available: https://roadsafety.piarc.org/sites/safety/files/public/pdf/piarc_road_safety_management_2024_05_27_v4.pdf

[3] L. Jiang, Y. Xie, X. Wen, and T. Ren, "Modeling highly imbalanced crash severity data by ensemble methods and global sensitivity analysis," Journal of Transportation Safety & Security, vol. 14, no. 4, pp. 562–584, Apr. 2022, doi: 10.1080/19439962.2020.1796863.

[4] C. Carrodano, "Data-driven risk analysis of nonlinear factor interactions in road safety using Bayesian networks," Sci Rep, vol. 14, no. 1, p. 18948, Aug. 2024, doi: 10.1038/s41598-024-69740-6.

[5] W. L. Carlson, "Crash injury prediction model," Accident Analysis & Prevention, vol. 11, no. 2, pp. 137–153, Jun. 1979, doi: 10.1016/0001-4575(79)90022-8.

[6] S. C. Joshua and N. J. Garber, "Estimating truck accident rate and involvements using linear and Poisson regression models," Transportation

Planning and Technology, vol. 15, no. 1, pp. 41–58, Jun. 1990, doi: 10.1080/03081069008717439.

[7] A. S. Al-Ghamdi, "Using logistic regression to estimate the influence of accident factors on accident severity," Accident Analysis & Prevention, vol. 34, no. 6, pp. 729–741, Nov. 2002, doi: 10.1016/S0001-4575(01)00073-2.

[8] K. M. Kockelman and Y.-J. Kweon, "Driver injury severity: an application of ordered probit models," Accident Analysis & Prevention, vol. 34, no. 3, pp. 313–321, May 2002, doi: 10.1016/S0001-4575(01)00028-8.

[9] Md. E. Shaik, Md. M. Islam, and Q. S. Hossain, "A review on neural network techniques for the prediction of road traffic accident severity," Asian Transport Studies, vol. 7, p. 100040, 2021, doi: 10.1016/j.eastsj.2021.100040.

[10] Y. Zhang, H. Li, and G. Ren, "Analyzing the injury severity in single-bicycle crashes: An application of the ordered forest with some practical guidance," Accident Analysis & Prevention, vol. 189, p. 107126, Sep. 2023, doi: 10.1016/j.aap.2023.107126.

[11] M. Rezapour, A. Mehrara Molan, and K. Ksaibati, "Analyzing injury severity of motorcycle at-fault crashes using machine learning techniques, decision tree and logistic regression models," International Journal of Transportation Science and Technology, vol. 9, no. 2, pp. 89–99, Jun. 2020, doi: 10.1016/j.ijtst.2019.10.002.

[12] Z. Ma, G. Mei, and S. Cuomo, "An analytic framework using deep learning for prediction of traffic accident injury severity based on contributing factors," Accident Analysis & Prevention, vol. 160, p. 106322, Sep. 2021, doi: 10.1016/j.aap.2021.106322.

[13] L. Pérez-Sala, M. Curado, L. Tortosa, and J. F. Vicent, "Deep learning model of convolutional neural networks powered by a genetic algorithm for prevention of traffic accidents severity," Chaos, Solitons & Fractals, vol. 169, p. 113245, Apr. 2023, doi: 10.1016/j.chaos.2023.113245.

[14] H. Bhuiyan et al., "Crash severity analysis and risk factors identification based on an alternate data source: a case study of developing country," Sci Rep, vol. 12, no. 1, p. 21243, Dec. 2022, doi: 10.1038/s41598-022-25361-5.

[15] S. AlKheder, F. AlRukaibi, and A. Aiash, "Risk analysis of traffic accidents' severities: An application of three data mining models," ISA Transactions, vol. 106, pp. 213–220, Nov. 2020, doi: 10.1016/j.isatra.2020.06.018.

[16] K. Sefrioui Boujemaa, I. Berrada, K. Fardousse, O. Naggar, and F. Bourzeix, "Toward Road Safety Recommender Systems: Formal Concepts and Technical Basics," IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 6, pp. 5211–5230, Jun. 2022, doi: 10.1109/TITS.2021.3052771.

[17] L.-Y. Chang and J.-T. Chien, "Analysis of driver injury severity in truck-involved accidents using a non-parametric classification tree model," Safety Science, vol. 51, no. 1, pp. 17–22, Jan. 2013, doi: 10.1016/j.ssci.2012.06.017.

[18] L. Wahab and H. Jiang, "A comparative study on machine learning based algorithms for prediction of motorcycle crash severity," PLOS ONE, vol. 14, no. 4, p. e0214966, Apr. 2019, doi: 10.1371/journal.pone.0214966.

[19] M. U. Abdulazeez, W. Khan, and K. A. Abdullah, "Predicting child occupant crash injury severity in the United Arab Emirates using machine learning models for imbalanced dataset," IATSS Research, vol. 47, no. 2, pp. 134–159, Jul. 2023, doi: 10.1016/j.iatssr.2023.05.003.

[20] A. Fernández, V. López, M. Galar, M. J. del Jesus, and F. Herrera, "Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches," Knowledge-Based Systems, vol. 42, pp. 97–110, Apr. 2013, doi: 10.1016/j.knosys.2013.01.018.

[21] S. Rezvani and X. Wang, "A broad review on class imbalance learning techniques," Applied Soft Computing, vol. 143, p. 110415, Aug. 2023, doi: 10.1016/j.asoc.2023.110415.

[22] S. Shaffiee Haghshenas, G. Guido, S. Shaffiee Haghshenas, and V. Astarita, "Predicting the level of road crash severity: A comparative analysis of logit model and machine learning models," Transportation Engineering, vol. 20, p. 100323, Jun. 2025, doi: 10.1016/j.treng.2025.100323.

[23] K. Aziz, F. Chen, M. Ahmad, M. S. Khan, M. M. Sabri Sabri, and H. Almujibah, "An interpretable dynamic ensemble selection multiclass imbalance approach with ensemble imbalance learning for predicting road crash injury severity," Sci Rep, vol. 15, no. 1, p. 24666, Jul. 2025, doi: 10.1038/s41598-025-08935-x.

[24] S. Zhu, K. Wang, and C. Li, "Crash Injury Severity Prediction Using an Ordinal Classification Machine Learning Approach," Int J Environ Res Public Health, vol. 18, no. 21, p. 11564, Nov. 2021, doi: 10.3390/ijerph182111564.

[25] "Open Data | Observatoire national interministériel de la sécurité routière." Accessed: Nov. 11, 2024. [Online]. Available: https://www.onisr.securite-routiere.gouv.fr/outils-statistiques/open-data

[26] J. Freeman, "Outliers in Statistical Data (3rd edition)," Journal of the Operational Research Society, vol. 46, Aug. 1995, doi: 10.1057/jors.1995.142.

[27] G. E. A. P. A. Batista and M. C. Monard, "An analysis of four missing data treatment methods for supervised learning," Applied Artificial Intelligence, vol. 17, no. 5–6, pp. 519–533, May 2003, doi: 10.1080/713827181.

[28] Dalwinder Singh, Birmohan Singh, "Investigating the impact of data normalization on classification performance," Applied Soft Computing Journal 97 (2020) 105524, 2020, doi: 10.1016/j.asoc.2019.105524.

[29] X. Zhu and X. Wu, "Class Noise vs. Attribute Noise: A Quantitative Study," 2004, doi: 10.1007/s10462-004-0751-8.

[30] C. Seger, "An investigation of categorical variable encoding techniques in machine learning: binary versus one-hot and feature hashing," 2018.

[31] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: Bagging-, boosting-, and hybrid-based approaches," IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, vol. 42, no. 4, pp. 463–484, 2012, doi: 10.1109/TSMCC.2011.2161285.

[32] A. B. Parsa, H. Taghipour, S. Derrible, and A. (Kouros) Mohammadian, "Real-time accident detection: Coping with imbalanced data," Accident Analysis & Prevention, vol. 129, pp. 202–210, Aug. 2019, doi: 10.1016/j.aap.2019.05.014.

[33] A. B. Parsa, A. Movahedi, H. Taghipour, S. Derrible, and A. (Kouros) Mohammadian, "Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis," Accident Analysis & Prevention, vol. 136, p. 105405, Mar. 2020, doi: 10.1016/j.aap.2019.105405.

[34] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," jair, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[35] T. Harmon, G. Bahar, and F. Gross, "Crash Costs for Highway Safety Analysis," 2018.

[36] M. Kačan, M. Oršić, S. Šegvić, and M. Ševrović, "Multi-Task Learning for iRAP Attribute Classification and Road Safety Assessment," in 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Sep. 2020, pp. 1–6. doi: 10.1109/ITSC45102.2020.9294305.

Machine Learning and Deep Learning for Detecting Fake News in a Low-Resource Language

Elton Tata, Jaumin Ajdarim, Nuhi Besimi
Faculty of Contemporary Sciences and Technologies, South East European University, Tetovo, Republic of North Macedonia

***Abstract*—Fake news detection has become a major problem in the digital age. This study presents an improved machine learning technique that achieves 91.99% accuracy in predicting fake news detection within Albanian textual datasets, demonstrating an improvement over existing baseline methodology. The implemented learning model uses 54 features that are specific to the Albanian language, such as red flags, credibility signals, punctuation patterns, and linguistic features. The model is tested on a balanced dataset of 3,994 news articles aggregated in Albanian from various sources. We compare it to several baselines, such as LSTM networks (80.35% accuracy) and BERT-augmented Naive Bayes classifiers (88.36% accuracy). In our Albanian dataset and experimental setting, XGBoost achieved 91.99% accuracy, indicating strong performance under the evaluated scenario.**

***Keywords*—Fake news; Albanian language; machine learning; NLP**

I. Introduction

The spread of lies and misinformation across the world, on digital platforms, poses a serious threat to democratic processes, sound decision-making, and public discourse. While much research has focused on developing sophisticated fake detection systems for high-resource languages such as English, Chinese, and Spanish by Kevin et al [1], Albanian, which is considered a low-resource language in the computational linguistics literature, has not yet been widely addressed. However, it lacks the extensive natural language processing resources, advanced pre-trained models, and annotated datasets that are available for major languages. In this context, the lack of resources presents interesting challenges for the development of efficient fake news detection systems and opportunities to investigate whether less expensive deep learning architectures can be used to achieve competitive performance using traditional shallow machine learning-based approaches developed with appropriate expertise in this field. From a computational perspective, Albanian has a number of features that make fake news detection both interesting and challenging. Special characters, such as "ë" (schwa) and "ç" (voiceless palatal affricate), are the foundation of the suggested language and help maintain a semantically accurate mapping. Due to the vocabulary of the Albanian language, there are many different types of hybrid words used in commerce, knowledge, and the performing arts [2]. The structure of the fixed definite article and the complex choice of verbs are two other notable patterns that stand in stark contrast to Romance and Germanic languages [3]. These linguistic features require customized features that our language model implementations take into account and capture the specific patterns of Albanian in both authentic journalism and fake news text.

A. Research Goals and Motivation

Three main observations of the state of the art in computational disinformation detection form the basis of the motivation behind this work. First, even in Albanian-speaking communities, the use of digital news and social media platforms for information consumption is more common, even in the absence of automatically filtered software to assess the credibility of their content. Second, due to structural differences in writing styles, languages, and cultural features, Albanian cannot be directly detected by standard fake news detection systems for most spoken languages.

Third, new research on transformer-based methods, such as multilingual BERT and other similar architectures [4]-[6], has shown that they can be used to detect fake news and do other NLP tasks across languages. But these methods usually need a lot of pretraining, a lot of computer power, and a long time to train, making them impractical for quick testing and deployment in research settings with few resources.

Our goal is to investigate the benefits of Albanian-specific features for deep learning techniques and compare them with traditional machine learning-based techniques (with or without language-specific enhancements) and the careless implementation of pre-trained models.

B. Contributions to Research

In the context of low-resource natural language processing and computational disinformation detection, this work offers three main contributions.

First, we create and validate a comprehensive set of 54 Albanian-specific linguistic features designed to capture structural and stylistic differences between fake and authentic Albanian news content, drawing on discourse-level cues inspired by Rhetorical Structure Theory (RST) [7] as well as carefully selected lexical, punctuation, and compositional patterns. While the latter prohibits the inclusion of features with low discriminatory power, such as sentence length, the former focuses on important aspects of the data to reduce overloading tendencies. Both the theory and conventions of Albanian news writing in the context of computational linguistics serve as the foundation for these features.

Second, using a set of methods, including hyperparameter search, forward selection from multiple model variants, and advanced text vectorization (with n-grams of characters), we

demonstrate a precisely calibrated Boost model with 91.99% accuracy.

Third, we demonstrate that, in the experimental context of this study and for the detection of Albanian-language fake news, a meticulously optimized machine learning methodology can attain superior accuracy and markedly reduced computational expenses in comparison to the assessed deep learning benchmarks.

The following sections comprise the rest of this document: Section II provides a literature survey of the earlier research related to this work. Section III presents the research methodology for the proposed study. In Section IV, the experimental results are presented. Section V show conclusion and future work.

II. Related Work

A. Content-Based Fake News Detection Approaches

Computational methods for detecting fake news and disinformation have evolved significantly over the past decade. Early approaches focused on content-based textual analysis and linguistic cues, drawing from techniques such as spam filtering and authorship identification [8]. Rubin et al. [9] identified stylistic and rhetorical patterns that differentiate deceptive content from legitimate journalism, showing that fake news often exhibits emotionally charged language, simplified grammatical structures, a lack of authoritative references, and sensational vocabulary [10].

More recent studies have expanded fake news detection beyond pure content analysis to incorporate source credibility and hybrid modelling strategies. Surveys by Shu et al. [11] and our previous work [12] consistently categorize detection methods into content-based, network-based, and hybrid approaches, while Guo et al. [13] further distinguish knowledge-based, style-based, and propagation-based strategies, building on propagation modelling techniques [14]. These studies consistently highlight content-based textual analysis as a core and indispensable component of effective fake news detection systems, motivating our emphasis on linguistic feature engineering.

B. Deep Learning Approaches and Low-Resource Constraints

Recent research has explored deep learning approaches, including recurrent and transformer-based models, for fake news detection. Such models are capable of automatically learning latent representations from raw text and have demonstrated strong performance in high-resource settings [15]. Transformer-based architectures, particularly BERT and its derivatives, have further advanced text classification performance and have been successfully adapted for fake news detection tasks [16],[17]. However, these approaches typically require large labelled datasets and substantial computational resources, which limit their practical applicability in low-resource research environments.

Low-resource languages face persistent challenges due to limited annotated data, scarce linguistic tools, and reduced representation in large pre-trained language models [18],[19]. Empirical studies further show that multilingual transformer models exhibit uneven performance across languages, with effectiveness strongly influenced by data availability and typological similarity [20]. These limitations highlight the need for alternative approaches that remain effective under data- and resource-constrained conditions.

C. Classical Machine Learning in Low-Resource Settings

A recurring question in fake news detection research concerns the conditions under which traditional machine learning methods can match or outperform deep learning approaches. Recent empirical studies indicate that well-designed classical models can achieve competitive performance on small- to medium-sized datasets when combined with effective feature engineering [21].

In particular, gradient-boosted decision tree models [22], including XGBoost, LightGBM, and CatBoost, have demonstrated strong performance and computational efficiency on structured and textual data. Essa et al. [23] report that LightGBM achieves high accuracy while maintaining computational efficiency, while other studies show that classical ensemble-based approaches can achieve competitive fake news detection performance with significantly lower computational cost than deep learning models. When paired with text vectorization techniques such as TF-IDF and higher-order n-grams, these models effectively capture discriminative linguistic patterns, making them especially suitable for domain-specific and low-resource scenarios. Overall, the reviewed literature provides strong justification for our approach, which emphasizes interpretable, Albanian-specific feature engineering and optimized classical machine learning models for effective fake news detection under limited data and computational resources.

III. Methodology

This section explains the extraction of features, the vectorization of text, and the architecture of our models (Fig. 2).

A. Data Collection and Sources

The dataset used in this study consists of 3,994 news articles in Albanian, collected from a wide range of Albanian-language media sources with different political orientations, geographical origins and journalistic standards [24].

There are 1,998 real news items and 1,996 fake news items in the dataset, which are almost perfectly balanced.

Every article that could be fake was manually reviewed, and only those that all commentators agreed on were kept in the final data.

The data includes the article title, the full text with original punctuation and paragraph structure, the date and time of publication, source information, and a binary class label.

B. Data Preprocessing and Quality Control

Text preprocessing steps were applied to reduce noise while preserving Albanian-specific linguistic characteristics relevant for downstream feature extraction. At the same time, this process explicitly preserves Albanian-specific characters, such as “ë” (schwa) and “ç”, which carry semantic value and are essential for the accurate representation of Albanian news texts (Fig. 1).

We address another Albanian-specific stopword removal, using a selected set of frequently occurring particles in texts with minimal discriminative value for classification, such as "dhe" (and), "e", "në" (in), "është" (is), "të" (to/of), "për" (for), "me" (me), "nga" (nga), etc., and their function words.

C. Data Partitioning Strategy

We use stratified random sampling to perform train-test splits and to ensure global class distribution in both subsets. 3,195 articles (80% of the data) are in the training set, with 1,598 articles that are real and an equal number that are fake, in which the class distribution remains approximately 50-50. The testing set includes 799 articles (20% of the total data), consisting of 400 real news items and 399 fake news items, once again maintaining class balance.

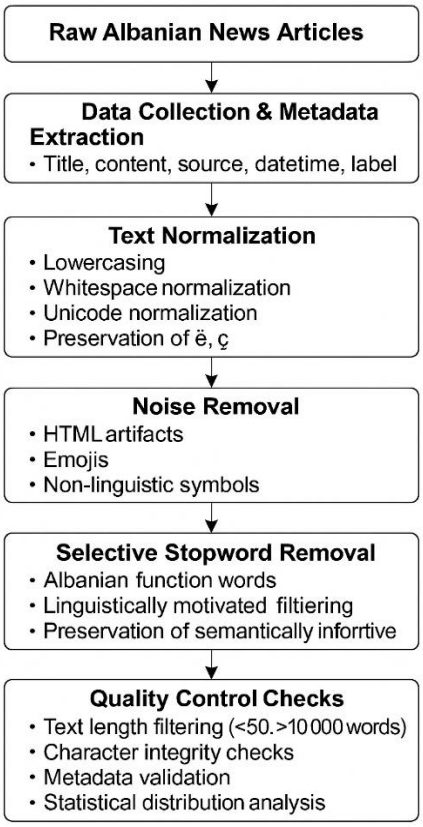


Fig. 1. Diagram for data processing and quality control.

D. Albanian-Specific Feature Engineering

We worked on 54 task-based features in five groups, taking different textual cues and metadata that we found empirically relevant for this active learning task - extracting different aspects of fabricated versus real news content.

E. Linguistic and Lexical Aspects of Albanian

The Albanian language has unique characteristics that enable the determination of genuine Albanian text versus text that may be translated from other languages or by non-native speakers [25]. The Albanian character ratio is the percentage of text that consists of two characters specific to Albanian, 'ë' and 'Ç', calculated as the number of these special characters divided by the total number of characters.

The binary variables control for the presence of Albanian characters (has_albanian_chars), which provides an admittedly simple but useful piece of information about the origin of the text. Vocabulary analysis features to check for the existence of sensational keywords that are typical of fake news in Albanian, such as "scandal" (scandal), "terror" (terror), "bomba" (bombë), "frikë" (fear), "shok" (shock), "alarm", "rrezik" (danger), "katastrofë" (catastrophe), "ekskluzive" (exclusive), "sekrete" (secret), "breaking" (borrowed English), "urgent" (borrowed English), "shikoni!" (look!) and "nuk do ta besoni!" (you won't believe it).

We derive both absolute counts (sensational word count) and ratios per document relative to document size (sensational ratio) to measure the strength of sensationalism, taking into account different lengths of articles.

The linguistic elements of opinion and defense cover question words and uncertainty devices such as "a" (whether), "vallë" (perhaps/I wonder), "ndoshta" (maybe), "mbase" (perhaps), "duket" (it appears), and "besoj" (I believe).

F. Characteristics of Punctuation Patterns

Patterns of punctuation [26] usage are strong indicators of Fake News vs. Professional News Sensationalism. Characteristics of exclamation marks consist of count, ratio of full document count to full document length, and binary flags for multiple exclamation marks (!! or !!!).

Question mark–based findings mirror the exclamation mark patterns. We analyze the number of question marks, their ratio relative to document length, and the detection of repeated question sequences (“??” or “???”). We track the use of ellipses, including in its sensationalist version, three consecutive dots (...). Contextual Citation and Citation Analysis.

Capitalization patterns are analyzed through the frequency and ratio of fully capitalized words, which tend to be more common in fake news than in professional journalism.

G. Statistical Textual Features

Word counts and sentence structures in the language model properties of the text are distinctive signals without being sensitive to the actual vocabulary [27]. We derive 15 statistical features to encode these features. Simple length measures are, e.g., total number of words, total number of characters, average word length (i.e., number of characters divided by number of words), and text length variation.

Sentence structure analysis applies regular expressions to determine sentence endings based on final punctuation (periods, exclamation marks, and question marks) and then calculates the number of sentences and their average word length, as well as their variation across the document. The unique word ratio represents the percentage of words confined to a single occurrence (type-mark) in the document, indicating diversity. The word repetition score, conversely, reflects the inverse metric (1 minus the type-mark ratio), which indicates how repetitive the content is.

H. Structural and Compositional Features

Aspects represent dependencies between content parts and logical structure [28]. We derive 8 features that control these

compositional properties. Headline analysis attributes that are analyzed independently of article titles include title length, title word count, and title punctuation (including exclamation and question marks). Fake news headlines tend to use question forms and their exclamatory counterparts to arouse curiosity or elicit an emotional response, while professional news headlines are more oriented toward so-called declarative patterns.

The title-content relationship models the relationship between headlines and news article bodies. Title_content_ratio calculates the ratio of title length to content, highlighting whether an article title is unusually long or short in relation to the information contained. Headline–content similarity features quantify the alignment between article titles and their corresponding bodies, capturing cases where titles diverge in length or content emphasis from the main text.

I. Time Features

When news timestamps are available, time features introduce an additional distinctive signal [16]. Five-time attributes have been identified from the publication_datetime metadata. The time of day takes the hour (values ranging from 0 to 23) when the article was published that day, which is expected to reflect different patterns in publication times. The day of the week (0="Monday", etc.) indicates the day of the week, allowing for weekly patterns in publication. The binary time indicators extracted are the is_weekend flag for Saturday/Sunday publications, capturing differences between weekend versus weekday publication patterns, and night by is_night if a publication was published between 22:00 and 05:00 local time when professional newsrooms are not active. This can reveal whether there are any patterns in the distribution of our data over time.

J. Construction of the Combined Feature Matrix

Each item is given 15,000 TF-IDF features using the vectorizer and 54 engineered_feats computed as described in the above section, resulting in a total data dimensionality of 15,054 features for each item. Scipy sparse matrix operations are used to join these feature sources into a single dataset by converting the 54 engineered features into dense arrays and stacking them horizontally with TF-IDF sparse matrices.

The gradient boosting model adapts the best feature weighting and interaction between both types of features, i.e., feature selection and synthesis, based on a unified training procedure.

K. Evaluation Methodology

We extract features for each test item in identical ways as during training and apply the trained TF-IDF vectorizer with a fixed vocabulary and IDF weights to transform the text and concatenate them to obtain 15,054-dimensional representations. We then use the models to produce probabilistic prediction results. The dimensionality was selected as a balance between representational richness and computational efficiency, as preliminary experiments indicated diminishing performance gains beyond this threshold.

Performance Metrics. Model performance is evaluated using standard classification metrics, including accuracy, precision per class, recall, and F1 score.

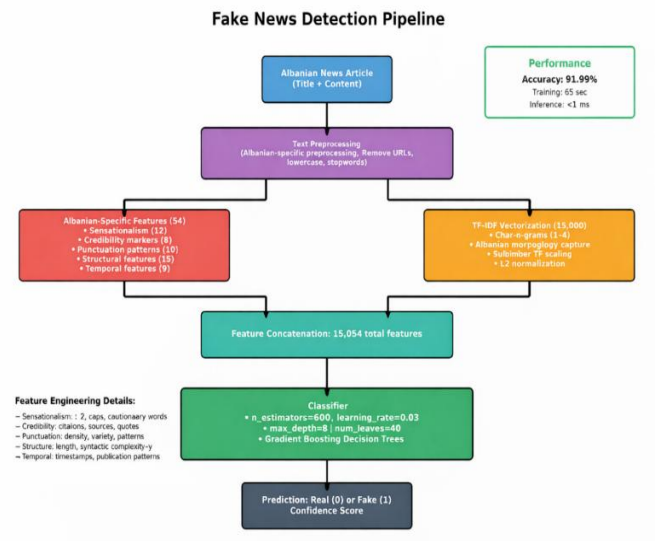


Fig. 2. Data processing, feature extraction, and classification pipeline used in this study.

IV. Experimental Results

The evaluation includes basic deep learning representational models, hybrid approaches, and gradient boosting models, to compare their performance under the same experimental conditions.

A. Summary of Evaluated Models

The evaluated models are grouped into four categories: basic deep learning models, hybrid models that combine neural representations with classical classifiers, gradient boosting models, and ensemble configurations. This categorization allows for a structured comparison of the models in terms of predictive performance and practical applicability.

B. LSTM Recurrent Neural Network and BERT-Enhanced Naive Bayes Hybrid

On the retained test set, the LSTM-based model achieved an accuracy of 80.35%, a significantly lower result than the best-performing gradient boosting models evaluated in this study. The comparatively lower performance of the LSTM model may be attributed to the moderate dataset size and the high dimensionality of the input representation. Deep sequential models typically benefit from substantially larger corpora or pretrained contextual embeddings. In low-resource settings, feature-engineered gradient boosting approaches may therefore exhibit greater stability and generalization capacity. With a training corpus of 3,195 items, the model seeks to learn word representations from scratch, which is insufficient for deep neural architectures that typically require significantly larger datasets to generalize well.

This hybrid model achieved an accuracy of 88.36%, a significant improvement of approximately 8 percentage points over the baseline LSTM model.

C. Logistic Regression Baseline

Despite its linear nature, logistic regression achieves competitive performance, achieving an accuracy of 89.36%, closely matching the result achieved by Naïve Bayes improved

by BERT. This result confirms that the combination of TF-IDF vectorisation and carefully designed Albanian language-specific features provides a very informative representation, even for simple linear classifiers.

D. Random Forest Baseline

The model achieved an accuracy of 88.49%, outperforming the baseline LSTM model (80.35%) and performing comparably to the BERT-enhanced Naïve Bayes model (88.36%).

The results indicate that tree-based ensemble methods combined with effective feature engineering can provide competitive performance without relying on deep neural architectures. The class-wise evaluation shows balanced behaviour, with true news accuracy of 88.91% and recall of 87.75%, and false news accuracy of 88.08% and recall of 89.22%, suggesting that there is no strong bias towards either class.

E. XGBoost Implementation

XGBoost is configured with such parameters for a fair comparison, including 500 boosting rounds, a 0.05 learning rate, a maximum tree depth limit of 7, a minimum child weight of 10, a subsample ratio of 0.8, and a column subsample ratio also set to 0.8. L1 regularization (alpha=0.1) and L2 regularization (lambda=1.0) are used to avoid overfitting on high-dimensional TF-IDF features by penalizing model complexity.

The XGBoost model achieved an accuracy of 91.99% on the full feature set. XGBoost model's prediction feature analysis indicates that it does well on articles with a clear vocabulary gap between fake and real news, can generalize across different-length articles by leveraging tree-based feature selection and captures non-linear interactions among predictive features with hierarchical splits (Fig. 3).

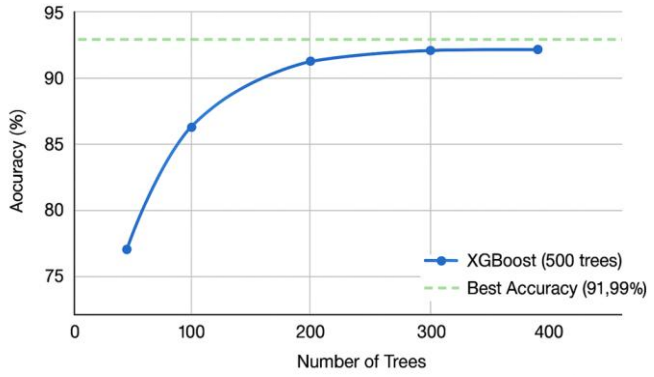


Fig. 3. Test accuracy of gradient boosting models.

F. Feature Importance and Interpretability Analysis

To further enhance the interpretability of the engineered feature set, feature importance analysis was conducted using XGBoost gain scores (Fig. 4). The results indicate that structural alignment cues, particularly the title_content_ratio, are among the most influential predictors. Content length and title-based attributes also rank highly, emphasizing that structural coherence plays a central role in distinguishing reliable journalism from misleading or clickbait-oriented content. Punctuation-related characteristics, including the frequency of exclamation marks and the presence of interrogative headlines, further contribute to classification decisions, supporting the hypothesis that exaggerated stylistic elements are associated with misleading articles. Lexical diversity measures such as word_repetition_score, unique word ratio, and word_length_variance also demonstrate notable predictive influence. Temporal attributes exhibit comparatively lower importance values, suggesting that they function as complementary contextual indicators rather than dominant classification drivers.

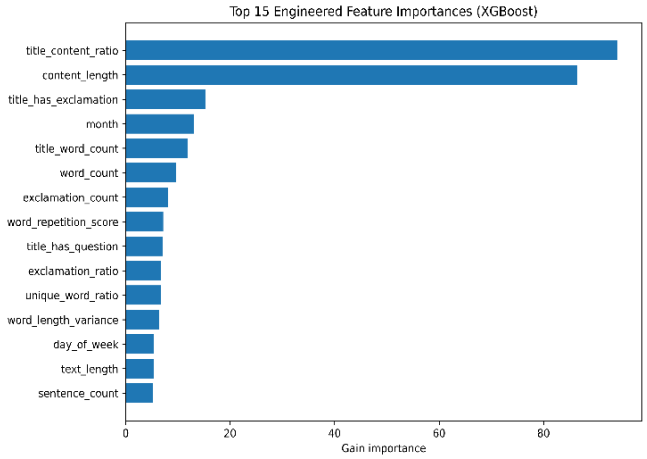


Fig. 4. Top 15 engineered feature importances based on XGBoost gain values.

G. Detailed Performance Evaluation

To provide a deeper evaluation of the classification behavior beyond the reported accuracy (91.99%), confusion matrix, and ROC–AUC analyses were conducted. These metrics offer additional insight into class-level performance and threshold sensitivity. The confusion matrix (Fig. 5) demonstrates a balanced distribution of classification outcomes across real and fake news categories. The model maintains comparable levels of false positives and false negatives, indicating the absence of systematic bias toward either class and confirming stable predictive behavior.

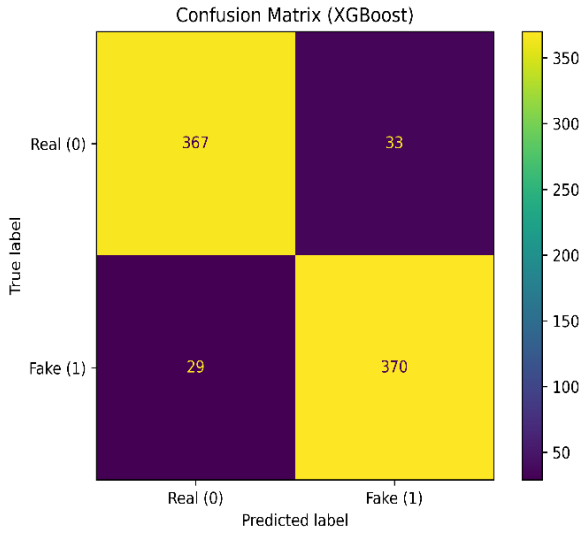


Fig. 5. Confusion matrix of the XGBoost classifier.

The Receiver Operating Characteristic (ROC) curve, Fig. 6, further illustrates strong separability between classes across varying decision thresholds. The model achieves an Area Under the Curve (AUC) score of 0.974, reflecting high discriminative capacity and robustness against threshold variation.

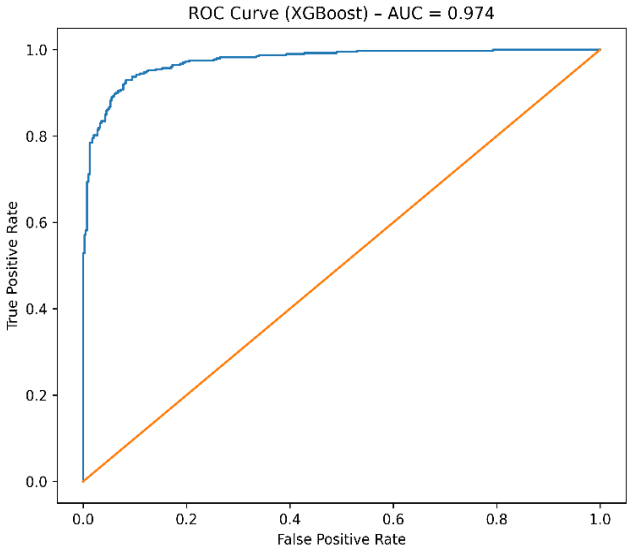


Fig. 6. ROC curve for the XGBoost classifier (AUC = 0.974).

H. Cross-Validation Stability Analysis

To ensure that the reported performance is not dependent on a single 80/20 train–test partition, a 5-fold stratified cross-validation procedure was conducted on the training data. The model achieved a mean accuracy of 91.08% with a standard deviation of 0.89%, indicating stable performance across folds. The low variance confirms that the classifier maintains consistent predictive behavior and does not rely on a specific data split. These findings support the robustness and generalization capacity of the proposed feature-engineered framework.

I. Error Analysis of Misclassified Instances

To further investigate model limitations, an analysis of misclassified instances was conducted based on the confusion matrix results. The model exhibits a balanced number of false positives and false negatives, indicating no systematic bias toward either class. False positives predominantly correspond to articles that display emotionally expressive or opinion-style language while remaining factually accurate. These cases suggest that stylistic intensity alone may sometimes resemble misleading patterns. On the other hand, false negatives usually include false information that is given in a neutral or formal tone and doesn't have any obvious sensational markers. Such cases highlight the challenge of detecting subtle or stylistically restrained misinformation.

Overall, the observed misclassifications suggest that ambiguity in stylistic signals remains the primary classification challenge, particularly when misleading content avoids exaggerated linguistic cues.

J. Comprehensive Model Comparison Table

Table I summarizes performance across all evaluated approaches, enabling direct comparison of accuracy, training time, model characteristics, and key strengths/weaknesses.

TABLE I. PERFORMANCE ACROSS ALL EVALUATED APPROACHES

Model	Accuracy	Training Time	Model Size	Key Strengths	Key Limitations
LSTM Baseline	*80.35%*	*60 min*	*2.4MB*	*Learns sequential patterns; no manual features*	*Slow training; overfits small data; requires large datasets*
Logistic Regression	*89.36%*	*1 sec*	*<1MB*	*Very fast; simple baseline; interpretable*	*Linear decision boundary; limited expressiveness*
BERT+NB Hybrid	*88.36%*	*2 min*	*850KB*	*Fast training; BERT tokenization*	*No contextual embeddings; independence assumption*
Random Forest	*88.49%*	*2 sec*	*15MB*	*Non-linear; handles interactions; interpretable*	*Outperformed by gradient boosting; larger model size*
XGBoost	*91.99%*	*35 sec*	*24MB*	*Robust; well-established*	*Best performance of the tested models.*

K. Comparative Analysis with Other Work

While this study focuses on fake news detection in Albanian, it is important to contextualize the obtained results in relation to existing studies conducted in other languages and datasets. A direct comparison with previous work allows a deeper understanding of whether the achieved performance is consistent with established empirical findings or represents an isolated result.

Therefore, a comparative analysis is conducted with representative fake news detection studies that use similar experimental environments, including textual datasets of comparable size and widely used machine learning and deep learning approaches. The comparison focuses mainly on accuracy, which is the most widely reported evaluation metric in the relevant literature.

As shown in Table II, previous studies on fake news detection report accuracy values that generally range between approximately 88% and 95%, depending on the data characteristics, feature representations, and learning models used. Classical machine learning approaches, especially ensemble-based models such as Random Forest and XGBoost, consistently demonstrate strong performance on medium-scale textual data.

TABLE II. COMPREHENSIVE MODEL PERFORMANCE COMPARISON

Study	Dataset Size	Method	Accuracy
This work (Albanian Fake News Detection)	*3,994*	*LR*	*89.36%*
		XGBoost	*91.99%*
		LSTM	*80.35%*
		BERT with Naive Bayes	*88.36%*
Alameri & Mohd [28]	*~5,000–7,000*	*SVM*	*88.20%*
		Naive Bayes	*84.60%*
		LSTM	*94.21%*
Abdulrahman & Baykara [29]	*7,796*	*Naive Bayes*	*90.10%*
		Random Forest	*95.66%*
		CNN	*97–98%*
Ahmad & Lokeshkumar [30]	*20,000*	*SVM*	*92.00%*
		Naive Bayes	*90.00%*
		Neural Network	*93.00%*
Alghamdi et al. [31]	*LIAR (~12k)*	*LR*	*88.50%*
		SVM	*89.00%*
		Naive Bayes	*87.00%*
		XGBoost	*92.10%*
		LSTM	*91.00%*
		BERT	*93.40%*
Albahr & Albahar [32]	*LIAR (~12k)*	*Naive Bayes*	*89.00%*
		Random Forest	*87.60%*
Sharma et al. [33]	*~6,000*	*LR*	*88.90%*
		Naive Bayes	*86.20%*
		Random Forest	*90.80%*
Kumar et al. [34]	*1,356*	*LSTM*	*88.00%*
		CNN + BiLSTM + Attention	*88.78%*

Deep learning models perform competitively when ample training data is present; however, their efficacy may fluctuate in contexts with constrained datasets. In this context, this study achieves an accuracy of 91.99% using TF-IDF features combined with XGBoost, which aligns well with the results reported in the literature for comparable experimental conditions.

These findings show that the proposed approach follows established empirical trends observed in different languages and datasets, confirming that gradient-boosted machine learning models remain an effective and reliable choice for fake news detection, especially in languages with limited data resources.

L. Ethical Considerations

Automated fake news detection systems may carry risks of misclassification that could affect journalistic credibility and public trust. Therefore, such systems should be deployed as decision-support tools rather than fully autonomous labeling mechanisms. Transparency, interpretability, and periodic human oversight are essential to ensure responsible use, particularly in low-resource linguistic contexts such as Albanian.

V. CONCLUSION

In this work, we showcased an optimized machine learning model for Albanian fake news detection, which achieved 91.99% accuracy by combining optimal Albanian feature engineering and XGBoost. This work shows that generic machine learning methods can achieve comparable performance to deep-learning solutions through integrating domain-specific knowledge and seeking systematic optimization.

We introduced and validated 54 Albanian specific optimization, linguistic features capturing sensationalism indicators, credibility markers, punctuation patterns, structural information and temporal features. The optimized XGBoost achieves an accuracy of 91.99%, significantly higher than other models.

REFERENCES

[1] Martínez-Gallego, K., Álvarez-Ortiz, A. M., & Arias-Londoño, J. D. (2021). Fake news detection in spanish using deep learning techniques. *arXiv preprint arXiv:2110.06461*.

[2] Krasniqi, K. (2019). The relation between language and culture (Case study Albanian Language). *Linguistics and Literature studies*, *7*(2), 71-74.

[3] Biberauer, T., & Roberts, I. (2008). Subjects, tense and verb-movement in Germanic and Romance. *Cambridge occasional papers in linguistics*, *3*, 24-43.

[4] Anirudh, K., Srikanth, M., & Shahina, A. (2023, December). Multilingual fake news detection in low-resource languages: A comparative study using BERT and GPT-3.5. In *International Conference on Speech and Language Technologies for Low-resource Languages* (pp. 387-397). Cham: Springer Nature Switzerland.

[5] De, A., Bandyopadhyay, D., Gain, B., & Ekbal, A. (2021). A transformer-based approach to multilingual fake news detection in low-resource languages. *Transactions on Asian and Low-Resource Language Information Processing*, *21*(1), 1-20.

[6] Bala, A., & Krishnamurthy, P. (2023, September). Abhipaw@ dravidianlangtech: Fake news detection in dravidian languages using multilingual bert. In *Proceedings of the Third Workshop on Speech and Language Technologies for Dravidian Languages* (pp. 235-238).

[7] Vargas, F., Jonas, D. A., Rabinovich, Z., Benevenuto, F., & Pardo, T. (2022, June). Rhetorical structure approach for online deception detection: A survey. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference* (pp. 5906-5915).

[8] Martins, R., Almeida, J. J., Henriques, P., & Novais, P. (2021). A sentiment analysis approach to improve authorship identification. *Expert Systems*, *38*(5), e12469.

[9] Rubin, V. L., Conroy, N., Chen, Y., & Cornwell, S. (2016, June). Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the second workshop on computational approaches to deception detection* (pp. 7-17).

[10] Wick-Pedro, G., de Sales Santos, R. L., & Vale, O. (2024, November). Linguistic and emotional dynamics in satirical vs. real news: a psycholinguistic analysis. In *Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana (STIL)* (pp. 386-392). SBC

[11] Zhou, X., & Zafarani, R. (2020). A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, *53*(5), 1-40.

[12] Tata, E., Ajdari, J., & Besimi, N. (2023, May). Fake News Detection: A Comprehensive Survey. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)* (pp. 309-314). IEEE.

[13] Guo, Z., Schlichtkrull, M., & Vlachos, A. (2022). A survey on automated fact-checking. *Transactions of the association for computational linguistics*, *10*, 178-206.

[14] Tufchi, S., Yadav, A., & Ahmed, T. (2023). A comprehensive survey of multimodal fake news detection techniques: advances, challenges, and

opportunities. *International Journal of Multimedia Information Retrieval*, *12*(2), 28

[15] Bahad, P., Saxena, P., & Kamal, R. (2019). Fake news detection using bi-directional LSTM-recurrent neural network. *Procedia Computer Science*, *165*, 74-82.Shishah, W. (2021). Fake news detection using BERT model with joint learning. *Arabian Journal for Science and Engineering*, *46*(9), 9115-9127.

[16] Kaliyar, R. K., Goswami, A., & Narang, P. (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. *Multimedia tools and applications*, *80*(8), 11765-11788.

[17] Joshi, P., Santy, S., Budhiraja, A., Bali, K., & Choudhury, M. (2020). The state and fate of linguistic diversity and inclusion in the NLP world. *arXiv preprint arXiv:2004.09095*

[18] Hedderich, M. A., Lange, L., Adel, H., Strötgen, J., & Klakow, D. (2021, June). A survey on recent approaches for natural language processing in low-resource scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 2545-2568).

[19] Pires, T., Schlinger, E., & Garrette, D. (2019). How multilingual is multilingual BERT?. *arXiv preprint arXiv:1906.01502*.

[20] Alameri, S. A., & Mohd, M. (2021, January). Comparison of fake news detection using machine learning and deep learning techniques. In *2021 3rd international cyber resilience conference (CRC)* (pp. 1-6). IEEE

[21] Lin, J., Tremblay-Taylor, G., Mou, G., You, D., & Lee, K. (2019, December). Detecting fake news articles. In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 3021-3025). IEEE

[22] Essa, E., Omar, K., & Alqahtani, A. (2023). Fake news detection based on a hybrid BERT and LightGBM models. *Complex & Intelligent Systems*, *9*(6), 6581-6592.

[23] Canhasi, E., Shijaku, R., & Berisha, E. (2022). Albanian fake news detection. *Transactions on Asian and low-resource language information processing*, *21*(5), 1-24

[24] Kabashi, B. (2018). A lexicon of Albanian for natural language processing. *Lexicographica*, *34*(1), 239-248.

[25] Sousa-Silva, R. (2022). Fighting the fake: A forensic linguistic analysis to fake news detection. *International Journal for the Semiotics of Law-Revue internationale de Sémiotique juridique*, *35*(6), 2409-2433.

[26] Samadi, M., Mousavian, M., & Momtazi, S. (2021). Deep contextualized text representation and learning for fake news detection. *Information processing & management*, *58*(6), 102723.

[27] Uppal, A., Sachdeva, V., & Sharma, S. (2020, January). Fake news detection using discourse segment structure analysis. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)* (pp. 751-756). IEEE.

[28] Alameri, S. A., & Mohd, M. (2021, January). Comparison of fake news detection using machine learning and deep learning techniques. In *2021 3rd international cyber resilience conference (CRC)* (pp. 1-6). IEEE.

[29] Abdulrahman, A., & Baykara, M. (2020, December). Fake news detection using machine learning and deep learning algorithms. In *2020 international conference on advanced science and engineering (ICOASE)* (pp. 18-23). IEEE.

[30] Ahmad, F., & Lokeshkumar, R. (2019). A comparison of machine learning algorithms in fake news detection. *International Journal on Emerging Technologies*, *10*(4), 177-183.

[31] Alghamdi, J., Lin, Y., & Luo, S. (2022). A comparative study of machine learning and deep learning techniques for fake news detection. *Information*, *13*(12), 576.

[32] Albahr, A., & Albahar, M. (2020). An empirical comparison of fake news detection using different machine learning algorithms. *International Journal of Advanced Computer Science and Applications*, *11*(9).

[33] Sharma, U., Saran, S., & Patil, S. M. (2020). Fake news detection using machine learning algorithms. *International Journal of creative research thoughts (IJCRT)*, *8*(6), 509-518.

[34] Kumar, S., Asthana, R., Upadhyay, S., Upreti, N., & Akbar, M. (2020). Fake news detection using deep learning models: A novel approach. *Transactions on Emerging Telecommunications Technologies*, *31*(2), e3767.

AMCS: Adaptive Multi-Controller SDN Security with Stateful Traffic Intelligence for Fast and Accurate Multi-Vector Attack Detection

Ameer El-Sayed[1], Mohamed Nosseir Hemdan[2], Noha Abdelkarim[3], Ehab Rushdy[4], Hanaa M. Hamza[5]
Department of Information Technology-Faculty of Computers and Informatics, Zagazig University, Zagazig 44519, Egypt[1, 2, 3, 4, 5]
Department of Information Technology-Faculty of Information Technology and Computer Science, Sinai University, Egypt[2]

Abstract*—The rapid proliferation of Software-Defined Networking (SDN) in large-scale Internet of Things (IoT) ecosystems has amplified exposure to sophisticated, multi-vector cyberattacks that simultaneously exploit control- and data-plane asymmetries. Existing single-controller statistical detectors, while effective under high-volume anomalies, fail to sustain precision and responsiveness under dynamic, distributed, or low-rate attack conditions. Addressing this critical gap, we propose AMCS—an Adaptive Multi-Controller SDN Security framework that fuses stateful traffic intelligence with cooperative inter-controller decision-making to enable resilient, context-aware detection across complex IoT traffic. AMCS embeds lightweight, P4-based stateful processing directly in the data plane and augments it with adaptive entropy-driven anomaly evaluation and consensus-based coordination among distributed controllers. Extensive experiments on an SDN–IoT testbed demonstrate that AMCS achieves up to 99.7% detection accuracy for high-volume floods, 96.8% for low-rate anomalies, and 94.1% under mixed traffic, while maintaining false-positive rates below 5% and detection latencies as low as 1.24 s. The cooperative consensus protocol enhances cross-controller reliability to 98.4% with only 0.83 s synchronization delay, while reducing control overhead by 34.7% compared to the single-controller baseline. Moreover, the distributed mitigation layer reacts within 1.6 s on average, neutralizing over 97% of attack flows with negligible collateral impact. Collectively, these results confirm that integrating stateful in-switch analytics, adaptive thresholding, and multi-controller cooperation establishes a scalable, self-adaptive SDN security fabric—achieving both fast detection and stable defense against evolving multi-vector threats in IoT-driven networks.

Keywords*—SDN; IoT; multi-controller security; stateful traffic analysis; adaptive anomaly detection; distributed mitigation; entropy-based indicators; cooperative defense; P4 programmable networks; cyber-physical systems security

I. Introduction

The rapid expansion of the Internet of Things (IoT) has reshaped modern network infrastructures, introducing unprecedented levels of heterogeneity, mobility, and traffic dynamism [1]. Although Software-Defined Networking (SDN) offers a programmable and logically centralized architecture capable of managing this complexity, its increasing adoption in IoT and cloud-integrated settings has exposed several structural vulnerabilities [2]. In particular, current SDN deployments depend heavily on the interaction between the control and data planes, making the controller a critical point whose performance can be disrupted by malicious traffic surges or crafted low-rate attacks [3]. These concerns are well documented in prior studies, which highlight how Distributed Denial-of-Service (DDoS) campaigns, spoofed traffic, and saturation attacks degrade controller responsiveness and accuracy, often overwhelming existing statistical detection methods that rely on basic entropy or rate measurements [4].

Recent research further emphasizes that the challenge extends beyond traditional high-volume attacks [5]. In IoT-dense environments, the number of devices and the irregularity of their traffic patterns increase the difficulty of distinguishing benign variations from genuine anomalies [6]. This complexity is compounded in multi-controller SDN architectures—designed to improve scalability and fault tolerance—where controllers must coordinate decisions, synchronize state, and maintain consistent security policies across distributed domains [7]. Without robust mechanisms for inter-controller cooperation, these architectures may inadvertently widen the attack surface or delay collective responses to emerging threats [8]. There is a growing need for secure, coordinated, and intelligent controller communication in SDN networks, which outlines challenges such as secure coordination, attack-surface expansion, and the limitations of stateless detection pipelines in fast-evolving IoT scenarios [9].

While previous work demonstrates that entropy-based methods can achieve high detection accuracy in controlled conditions for specific attack models, these techniques often operate independently on each controller and lack a mechanism to incorporate deeper traffic state or contextual behavior [10], [11]. As a result, statistical fluctuations, low-rate probing, and distributed multi-vector attacks may evade or delay detection [12]. The limitations documented in earlier SDN studies point to a broader gap: existing single-controller, threshold-driven systems are not designed to reason about traffic evolution over time, nor to leverage cross-controller observations that could reveal subtle, otherwise-hidden attack patterns [13].

Motivated by these challenges, this work explores an adaptive multi-controller security framework that integrates stateful traffic inspection with cooperative intelligence sharing. The proposed approach integrates multi-layer protection, real-time stateful monitoring, secure controller-to-controller communication, and seamless orchestration of distributed defensive functions across the network. By combining stateful flow context with coordinated multi-controller analysis, the

proposed framework aims to enhance detection robustness against diverse attack strategies—including low-rate exfiltration attempts, distributed probing, and rapid DDoS-like bursts—while maintaining the performance and scalability demands inherent to IoT-driven networks.

II. Literature Review

The convergence of SDN and the IoT has reshaped modern network management, offering programmability, centralized orchestration, and policy-driven control. However, this integration also exposes new vulnerabilities. The centralized nature of SDN controllers and the heterogeneous, resource-limited behavior of IoT nodes collectively create an expanded attack surface, particularly against multi-vector cyberattacks. Maintaining the stability of SD-IoT environments under such conditions has therefore become a cornerstone of current research, as network-layer resilience directly underpins higher-level security assurances.

Early contributions in this domain largely focused on controller-centric machine learning (ML) approaches for attack detection. Adaptive ensemble frameworks and hybrid classifiers achieved notable accuracy in simulated testbeds but frequently failed to generalize beyond controlled datasets [14]. For instance, neural-network–SVM hybrids demonstrated high detection performance on benchmark datasets but suffered from limited scalability and dataset bias [15]. Other lightweight gradient-boosting models improved training efficiency yet still relied heavily on handcrafted features and synthetic traffic patterns, reducing their practicality for real IoT deployments [16].

Subsequent research explored deep learning (DL) and reinforcement learning (RL) paradigms to enhance automation and adaptiveness [17]. Architectures that combined DL-based detectors with RL-driven mitigation improved dynamic decision-making but were constrained by computational overheads and untested scalability in multi-controller topologies [18]. CNN- and DBN-based frameworks [19], [20] produced high detection rates; however, their limited interpretability and dependence on non-IoT datasets restricted their utility in heterogeneous traffic environments.

Parallel to controller-side innovation, attention shifted toward programmable data-plane intelligence. The advent of P4 switches enabled in-network detection, where selected traffic features are processed closer to the source. P4-assisted methods demonstrated substantial latency reduction and early anomaly recognition, particularly for TCP- and HTTP-based floods [21], [22]. Nevertheless, computational constraints within the data plane and narrow attack coverage remained unresolved. Meanwhile, controller-integrated ML frameworks—such as SDN-WISE [23] and DALCNN [24]—offered improved localized detection accuracy but continued to suffer from single-point bottlenecks and dependence on static detection thresholds.

A later wave of research introduced hybrid statistical–ML approaches to balance efficiency and adaptability. Models combining entropy-based indicators with machine learning improved early detection of low-rate anomalies. Similarly, edge-level designs leveraging Markov models and hybrid deep networks achieved effective performance in mitigating slow-rate IoT botnet traffic [25]. Yet, these methods often lacked flexibility under variable attack rates, emphasizing the need for dynamic temporal analysis and context-aware feature extraction [26], [27]. The shift toward adaptive temporal windows and confidence-based learning mechanisms represents a direct response to these limitations [28].

More recent studies have extended this evolution into context-aware, multi-phase, and transformer-based architectures. Attention-driven models such as SAINT [29] and multiphase entropy–clustering frameworks demonstrated high detection accuracy (above 96%) in cloud and fog environments [30]. Comparative evaluations [31] reaffirmed the efficiency of traditional ML classifiers—like Decision Trees and Random Forests—as practical baselines for constrained devices, while hybrid interpretable approaches integrated explainability tools such as SHAP for domain-specific applications (e.g., healthcare IoT) [32]. Nonetheless, the majority of existing work remains tied to isolated control domains or non-IoT datasets, limiting generalization across distributed and dynamic infrastructures [33].

Despite notable advances, few existing frameworks achieve real dual-plane coordination—that is, combining stateful in-switch analytics with adaptive learning and distributed mitigation across multiple controllers. As large-scale DDoS attacks remain the most frequent and destructive threat to IoT ecosystems, building resilient, cooperative, and self-adaptive SD-IoT defenses remains an open and urgent research frontier.

The proposed framework in this study contributes to this effort by integrating stateful data-plane inspection, entropy-aware adaptive detection, and multi-controller collaborative mitigation. Table I summarizes the representative literature, highlighting trends, strengths, and research gaps that informed the design of the proposed solution.

TABLE I. Summary of Representative Research on Attack Detection and Mitigation in SD-IoT Networks

Ref.	Year	Approach	Main Strengths	Identified Limitations
[17]	2022	SDN framework with DL detection + RL mitigation	Dynamic, adaptive countermeasures	Scalability not validated in real SD-IoT settings
[15]	2022	FFCNN–SVM hybrid for low-rate DDoS	Accurate under controlled conditions	Single attack focus, limited dataset diversity
[24]	2022	DALCNN integrated with SDN controller	High accuracy and stability	Dependent on non-IoT traffic datasets
[21]	2022	P4-enabled ML-based detection	Low latency, edge-level monitoring	Narrow attack scope, heavy switch load
[27]	2022	Entropy + ML hybrid model	Fast and precise detection	Limited handling of multi-vector attacks

[18]	2023	DL-driven SDN defense with adaptive responses	Integrative multi-stage mitigation	Lacks IoT-specific validation
[28]	2023	FMDADM using packet-window features	Effective IoT botnet detection	Limited adaptability to variable traffic rates
[23]	2023	SDN-WISE ML-based detection	Improved localized accuracy	Single-controller dependency
[16]	2023	LightGBM-based feature engineering	Efficient and lightweight	Synthetic data reliance
[25]	2024	Edge-level hybrid deep learning	Robust low-rate attack detection	Scalability is unproven in large topologies
[26]	2024	Smart-home ML + signature detection	High accuracy for specific IoT domains	Poor generalization to other IoT environments
[22]	2024	P4-based HTTP DDoS defense	Low latency, real-time operation	High computational cost on P4 devices
[29]	2025	Transformer (SAINT) model for TCP-DDoS	Highly interpretable, scalable	Focused mainly on transport-layer floods
[30]	2025	Multiphase ML–entropy fog architecture	High classification accuracy, quick response	Evaluation restricted to fog nodes
[34]	2025	CNN–LSTM with SHAP interpretability	Transparent DL behavior	Limited to the healthcare IoT context
[33]	2025	ONOS Flood Defender with adaptive algorithms	Real-time SYN flood mitigation	Fixed statistical thresholds, limited adaptability
This Work	**2025**	P4-based SD-IoT framework with adaptive detection and multi-controller mitigation	Real-time stateful feature extraction, confidence-adaptive learning, scalable coordination	Requires further testing under large-scale, distributed failover conditions

III. Proposed Framework

This section presents the proposed adaptive security framework for multi-controller SDN environments supporting heterogeneous IoT deployments. The framework addresses core challenges in scalability, control-plane coordination, and adaptive security monitoring by combining stateful traffic analysis, cooperative controller synchronization, and distributed mitigation. It is structured around four main components:

1) The system model, describing the architectural layers and interactions between network entities;

2) The threat model, identifying relevant attack vectors and their potential impact;

3) The stateful traffic-processing and adaptive anomaly evaluation modules, enabling context-aware detection; and

4) The multi-controller coordination and response mechanisms, which provide resilience and consistent defense actions.

Section 3.1 introduces the overall system architecture, while Section 3.2 defines the threat landscape and attacker capabilities. The following subsections detail the analytical and cooperative modules that enable detection, decision-making, and mitigation within the distributed control environment.

A. System Model

The proposed Adaptive Multi-Controller SDN Security (AMCS) framework (Fig. 1) embodies a hierarchical yet cooperative architecture that unifies stateful data-plane intelligence with distributed controller coordination. It is composed of three principal tiers and eight logically connected functional modules that collectively enable programmable, adaptive, and synchronized protection across large-scale IoT networks.

(1) IoT and Edge Layer

At the edge of the architecture lies a diverse ecosystem of IoT devices—sensors, cameras, gateways, and embedded controllers—that generate heterogeneous traffic streams. These devices (Step ① in Fig. 1) transmit telemetry, command, and control data toward the SDN infrastructure. Their irregular bursty transmissions and constrained resources make them vulnerable entry points for both volumetric and stealthy threats such as spoofing, infiltration, and low-rate exfiltration.

(2–5) Data-Plane Intelligence: Stateful Feature Extraction and Analysis

Incoming traffic first enters the SDN data plane (Step ②), composed of P4-programmable switches capable of executing local inspection before involving the controller. Each switch captures and extracts essential flow-level attributes—packet size, inter-arrival time, flow initiation rate, and destination entropy (Steps ③–④)—which are forwarded to the Stateful Traffic-Processing Module (STPM) located inside the P4 pipeline (Step ⑤).

The STPM maintains per-flow state tables and statistical registers that continuously update as packets traverse the switch. This enables in-network anomaly sensing by identifying temporal irregularities in packet behavior, thereby offloading early detection from the control plane. By exporting only aggregated summaries rather than raw packets, the STPM significantly reduces packet in congestion and controller CPU load.

(6) Control-Plane Intelligence: Adaptive Anomaly Evaluation (AAE)

Processed telemetry from STPM is sent to the controller's Adaptive Anomaly Evaluation (AAE) module (Step ⑥). The AAE dynamically computes entropy-based deviation scores and adjusts its thresholds using real-time variance and historical profiles. This allows the controller to distinguish flash-crowd events from true anomalies. Each controller thus maintains a localized, context-aware anomaly model that evolves with observed traffic behavior.

(7) Inter-Controller Coordination: Multi-Controller Coordination Module (MCCM)

Because the AMCS framework deploys multiple distributed controllers across domains (C1, C2 … Cn), synchronization of

local detections is essential. The MCCM (central-CO exchange shown in Fig. 1, Step ⑦) aggregates alerts from all controllers, performs consensus voting based on trust weights, and disseminates unified decisions across peers. Consensus cycles occur periodically, ensuring global consistency without imposing excessive communication overhead. This cooperative verification minimizes false positives and enables rapid multi-domain agreement—typically within 0.8 s, as shown in experimental results.

(8) Global Mitigation and Policy Enforcement: Distributed Mitigation Workflow (DMW)

Once the MCCM confirms an incident, the Distributed Mitigation Workflow (DMW) (Step ⑧) executes adaptive countermeasures proportionate to alert confidence and severity. Controllers enforce localized rate-limiting, flow blocking, or rerouting policies within their domains, while global synchronization ensures that legitimate flows remain unaffected. This distributed enforcement loop closes the detection–response cycle, maintaining network stability and service continuity.

The bidirectional arrows between modules denote feedback loops: mitigation outcomes refine AAE thresholding, and updated trust weights from MCCM influence future coordination cycles. Together, these interactions form a closed, adaptive control system that continually tunes its detection sensitivity, synchronization accuracy, and policy strength.

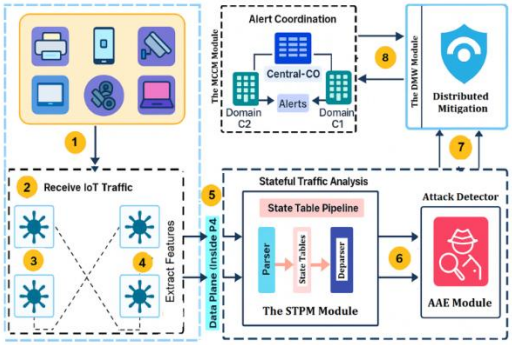


Fig. 1. System model of the AMCS framework showing end-to-end data flow from traffic ingestion to distributed mitigation.

B. Threat Model

The threat model defines the adversarial assumptions, potential attack vectors, and corresponding defense mapping relevant to the proposed AMCS framework. It considers an intelligent, adaptive adversary capable of strategic coordination across IoT domains, yet without direct access to the SDN controller software or administrative privileges. The attacker's objectives are to disrupt network availability, degrade controller responsiveness, and exploit synchronization weaknesses in multi-controller SDN environments.

As illustrated in Fig. 2, the threat model encompasses three logical dimensions: (1) Adversary Capabilities, (2) Attack Vectors and Impacted Assets, and (3) Defense Mapping to the corresponding AMCS protection modules.

To counter these coordinated threats, the AMCS architecture implements a layered defense mapping between identified attack vectors and specific protection modules (as shown in Table II). Table III summarizes the threat characteristics and system impacts under the assumed adversarial model. The mapping clearly demonstrates that AMCS establishes a defense continuum—from in-switch detection (STPM) to global mitigation (DMW)—capable of addressing both volumetric and adaptive multi-vector attacks.

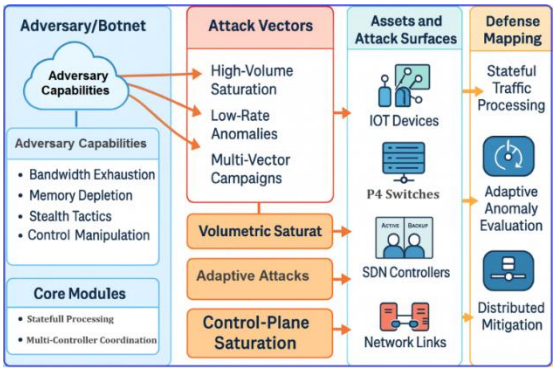


Fig. 2. Threat model showing adversarial capabilities, targeted SDN–IoT assets, corresponding attack vectors, and mapped AMCS defense mechanisms.

TABLE II. DEFENSE MAPPING TO AMCS MODULES

Threat Category	Affected Asset(s)	Primary Defense Mechanism
High-Volume Saturation	P4 Switches, Controllers	Stateful Traffic Processing (STPM) detects abnormal packet bursts directly within the data plane, throttling early-stage floods before escalation.
Low-Rate Stealth Attacks	IoT Devices, Network Links	Adaptive Anomaly Evaluation (AAE) dynamically adjusts detection thresholds using entropy variance and temporal context to capture slow anomalies.
Multi-Vector Distributed Campaigns	Multiple Controllers, Domains	Multi-Controller Coordination Module (MCCM) correlates alerts and ensures synchronized detection decisions across domains.
Control-Plane Manipulation	Controllers, State Tables	Distributed Mitigation Workflow (DMW) enforces coordinated policy adaptation, restoring consistent states across controllers and minimizing response delay.

TABLE III. THREAT MODEL SPECIFICATIONS

Threat Category	Adversary Capability	Impact on System
High-Volume Saturation	Flooding traffic to overload switches/controllers	Excessive *packet_in* load, delayed flow setup, and reduced throughput
Low-Rate Stealth Attacks	Gradual, subtle anomalies	Evades fixed thresholds, increases false negatives
Multi-Vector Distributed Attacks	Coordinated anomalies across domains	Inconsistent detection, delayed mitigation
Control-Plane Manipulation	Triggering conflicting flow behaviors	State inconsistency, incorrect routing, degraded QoS

C. Stateful Traffic-Processing Module (STPM)

Headings: The stateful traffic-processing module forms the analytical core of the framework. Its role is to extract concise indicators from live traffic, maintain short-term and long-term behavioral states, and provide each controller with context-rich signals that improve anomaly detection. Traditional SDN switches operate in a stateless manner, forwarding packets based solely on flow rules. This stateless behavior limits their ability to recognize deviations in flow evolution, protocol usage, or cross-packet characteristics. To overcome these limitations, the proposed module leverages programmable data-plane capabilities—implemented using P4—to embed lightweight stateful operations directly into the forwarding pipeline.

The processing logic follows a structured pipeline, as indicated in Fig. 3. Incoming packets are first parsed to extract protocol headers and application-layer fields relevant to DNS, HTTP, or other IoT-centric protocols. The parser feeds these fields into a match–action stage where the switch applies lookup rules while also performing state updates using registers and counters. These structures enable the switch to track properties such as packet size deviations, flow creation rates, destination diversity, and inter-arrival timing patterns. By maintaining state across packets, the switch can signal abnormal trends long before they result in high-volume anomalies or controller overload.

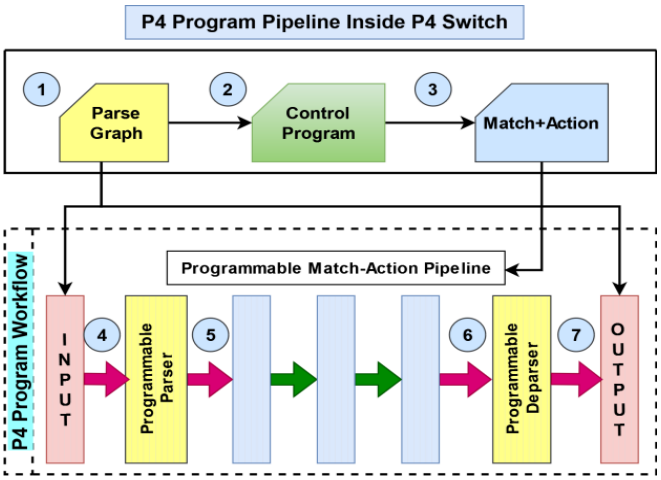


Fig. 3. Internal architecture and P4 pipeline implementation for stateful traffic-processing module (STPM).

A key advantage of this approach is that traffic summarization occurs in-network, reducing the controller's analytical burden. Instead of transmitting every suspicious packet, the switch periodically exports compact telemetry reports, such as deviations in feature values, sudden expansion of destination entropy, or sustained increments in per-flow byte counters. The module, therefore, blends immediate packet-level inspection with incremental state accumulation, producing a richer and more stable detection signal than stateless indicators.

The features selected for extraction reflect traffic characteristics known to fluctuate significantly under both high-volume and low-rate attack conditions. These features capture both structural packet properties and temporal flow behavior. Table IV summarizes the main features processed by the module and the corresponding P4 elements used to extract and maintain their state.

TABLE IV. EXTRACTED STATEFUL FEATURES AND P4 PIPELINE COMPONENTS

Feature	Description	P4 Element Used	Statefulness
Packet Size Deviation	Tracks changes in packet payload sizes over time, useful for spotting tunneling or abnormal payload inflation	register, counter	Maintains running average and variance
Flow Initiation Rate	Measures the frequency of new flow creation to detect sudden expansion typical of scans or DDoS bursts	register, metadata, digest	Stores timestamps and increments counters
Destination IP Spread	Tracks the diversity expansion of outbound destinations to reveal distributed probing	register array indexed by destination prefix	Persistent across time windows
Inter-Arrival Time	Measures packet timing irregularities indicative of slow-rate or stealth attacks	timestamp, register	Stores last-seen time per flow
DNS Query Length	Flags abnormal query sizes often associated with exfiltration tunnels	Custom P4 parser fields	Stateful threshold comparison
Byte Count per Flow	Tracks cumulative bytes to detect gradual data leakage	counter per flow	Persistent accumulation
Protocol Distribution	Monitors shifts in protocol usage, such as sudden surges in UDP flows	Table-level statistics, counter	Window-based aggregation
Source Entropy Indicators	Measures variability in the source distribution used during volumetric anomalies	Aggregated via digest to controller	Semi-stateful (derived from exports)

Stateful processing significantly enhances the switch's analytical capability. Rather than relying on instantaneous metrics, the module captures temporal evolution, which is critical for identifying low-rate deviations and subtle behavioral drift. Attacks that unfold gradually—such as data exfiltration, incremental scanning, or distributed low-volume campaigns—often evade purely statistical detectors. Maintaining state allows the system to detect cumulative anomalies, such as persistent shifts in packet size distributions, slow increases in destination diversity, or sudden escalations in flow initiation.

The design ensures that the state is both local and lightweight. Registers store compact numeric values indexed by flow or prefix, enabling O (1) updates while preserving forwarding speed. Counters accumulate features with minimal overhead. Furthermore, because periodic summaries rather than raw packets are exported, the control plane avoids congestion even in dense IoT environments.

Across the detection pipeline, stateful analysis provides three distinct advantages. First, it increases robustness by

smoothing short-term statistical noise. Second, it enhances sensitivity to low-rate or stealthy patterns. Third, it allows each controller to reason about local behavior using richer context before participating in cooperative, multi-controller verification. This combination of local state and global coordination strengthens the overall resilience of the network against diverse and evolving attack strategies.

As indicated in Algorithm 1, the Stateful Traffic-Processing Module operates directly within the data plane and incrementally maintains behavioral context for each flow. Its three-phase structure ensures that initialization sets stable analytical parameters, packet-level operations apply lightweight stateful logic, and telemetry summaries are efficiently exported without overwhelming the controller.

By combining inter-arrival tracking, byte accumulation, destination-spread evolution, and dynamic packet-size statistics, the module produces a refined view of traffic behavior long before anomalies manifest at larger scales. This approach reduces controller overhead and enables early identification of threats such as low-rate scans, slow-drip exfiltration, or sudden distribution shifts in outbound destinations. The final summary phase ensures that rich but compact flow-level insights are delivered upstream for coordinated decision-making.

```
Algorithm 1: Stateful Traffic-Processing Module (STPM)
Inputs:
      - IncomingPackets
      - FeatureThresholds
      - TelemetryWindow
      - SwitchStateTables
      - TimeStamps

Outputs:
      - Flow-level state summaries
      - Local anomaly flags
      - Compressed telemetry reports

BEGIN
    WHILE switch is active DO
        FOR each packet p ∈ IncomingPackets DO
            header ← ParsePacket(p)
            features ← ExtractFeatures(header)
            flowID ← IdentifyFlow(p)
            state ← LoadState(flowID)

            Stateful Updates
            state.byteCount ← state.byteCount + features.size
            state.IAT ← features.timestamp – state.lastSeen
            state.sizeStats ← UpdateSizeStatistics
            state.dstSpread ← UpdateSpreadMap(features.dstIP)

            Local Anomaly Check
            anomaly ← EvaluateLocalAnomaly(FeatureThresholds)
            IF anomaly == TRUE THEN
                GenerateLocalAlert(flowID, state)
                StoreState(flowID, state)
            END IF
        END FOR

        Telemetry Window Completion
        IF TelemetryWindowExpired() THEN
            report ← BuildTelemetry(SwitchStateTables)
            ExportTelemetry(report)
            ResetTelemetryWindow()
        END IF
    END WHILE
END
```

D. Adaptive Anomaly Evaluation Module

Accurate detection in dynamic SDN–IoT environments requires more than static comparison of feature values against fixed thresholds. Traditional statistical detectors often misclassify benign surges or clustered traffic as anomalies, and may also miss low-rate attacks that evolve gradually. These limitations have been well-documented, particularly the difficulty of distinguishing flash crowds from genuine attack traffic or differentiating short legitimate flows from spoofed ones. Moreover, entropy or feature-based indicators can fluctuate under heterogeneous traffic conditions, leading to false positives or delayed detection unless thresholds adapt to the observed traffic behavior. Several studies highlight that adaptive thresholding, especially when combined with multi-feature signals, improves stability and avoids misdiagnosis during both low-rate and distributed attack scenarios.

```
Algorithm 2: Adaptive Anomaly Evaluation (AAE)
Inputs:
      - FlowSummaries
      - LocalAlerts
      - HistoricProfiles
      - EntropyWindow
      - AdaptiveThresholdParameters

Outputs:
      - Updated anomaly threshold
      - Confirmed anomaly events
      - Controller-level alert flags

BEGIN
    # Phase 1 – Telemetry Consolidation
    mergedSummaries ← Merge(FlowSummaries, LocalAlerts)
    currEntropy ← ComputeEntropy(mergedSummaries, Entr_Window)
    baselineEntropy ← HistoricProfiles.recentEntropy
    entropyVariance ← Variance(HistoricProfiles.entropyHistory)
    drift ← |currEntropy – baselineEntropy|

    # Phase 2 – Adaptive Threshold Adjustment
    IF drift > entropyVariance THEN
        threshold ← AdjustThresholdUp(AdaptiveThre_Params)
    ELSE
        threshold ← AdjustThresholdDown(AdaptiveThre_Params)
    END IF
    UpdateThreshold(threshold)

    # Phase 3 – Anomaly Confirmation
    FOR each flow f in mergedSummaries DO
        score ← WeightedScore(
                    f.entropyDrop,
                    f.IAT,
                    f.dstConcentration
                )
        IF score ≥ threshold THEN
            ConfirmAnomaly(f)
            ShareAlert(f)
        ELSE
            ContinueNormal(f)
        END IF
    END FOR
END
```

The module operates in three stages. First, it consolidates telemetry from all switches, forming a global snapshot of flow-level behavior. Second, it evaluates anomalies using a weighted scoring model that accounts for entropy drop rate, inter-arrival irregularity, flow initiation bursts, and destination concentration ratios—metrics demonstrated to be reliable

under the entropy-based detection studies recorded in the previous work. Third, it dynamically adjusts the threshold based on the stability of these metrics. When legitimate and attack traffic characteristics converge—as demonstrated in multiple traffic models B and C—the adaptive mechanism stabilizes the detection rate by adjusting the decision boundary to reflect the mixed nature of traffic. To support this behavior, the module follows the operational logic summarized in Algorithm 2.

The AAE algorithm (Algorithm 2) refines the raw indicators produced by STPM into more robust and reliable anomaly decisions. Its input includes aggregated entropy trends, destination-spread divergence, and inter-arrival time irregularities. These signals align with the documented behaviors of both normal and attack traffic in SDN–IoT environments: for example, sharp drops in entropy correlate strongly with attack bursts, while mild fluctuations may represent flash crowds or active endpoints.

In Phase 1, the controller consolidates telemetry across multiple switches, forming a unified analytical surface. This ensures the evaluation process recognizes distributed effects that would be invisible to per-switch detection alone. In Phase 2, the adaptive thresholding mechanism adjusts the detection boundary by analyzing the direction and magnitude of entropy drift relative to its historical variance. This method directly addresses conditions observed in the experiment results, where similar traffic patterns produced different entropy values across traffic models A, B, and C, and where adaptive thresholding sustained detection rates above 93% even under complex flows.

Finally, Phase 3 applies weighted scoring to detect anomalies. By integrating multiple features rather than relying solely on entropy, the evaluation process becomes less sensitive to noise and better aligned with multi-vector attack characteristics. Confirmed anomalies are shared across controllers to enable synchronized response in multi-controller environments.

E. Multi-Controller Coordination Module (MCCM)

Accurate Large-scale SDN deployments increasingly adopt distributed control architectures to improve scalability, resilience, and latency. While multiple controllers enhance responsiveness, they introduce a crucial coordination challenge: ensuring that each controller maintains a consistent and synchronized view of network security state. Without proper coordination, local detections may remain isolated, delaying collective mitigation or generating conflicting control actions. The Multi-Controller Coordination Module (MCCM) resolves these inconsistencies by enabling real-time collaboration, state sharing, and consensus-driven anomaly verification among distributed controllers.

The MCCM acts as a synchronization layer above individual anomaly evaluators. Each controller operates independently on its local traffic domain using the Adaptive Anomaly Evaluation (AAE) process, but periodically exchanges summarized alerts and trust-weighted anomaly scores with peer controllers. This exchange allows for correlation of distributed evidence—such as simultaneous entropy drops, synchronized flow expansions, or shared destination clusters—indicating a coordinated or multi-vector attack. Through a combination of message aggregation, voting, and dynamic trust adjustment, controllers reach a joint decision on whether to escalate an alert into a confirmed global incident.

Coordination operates over lightweight communication channels, often gRPC or message-bus based, ensuring minimal delay. Each controller maintains a peer state table containing confidence weights and last-synchronization timestamps for all neighboring controllers. During collaboration, alerts are cross-validated using these weights, allowing the system to emphasize highly reliable peers while minimizing false confirmations from unstable nodes. Algorithm 3 presents the operational workflow of this module.

Algorithm 3: Multi-Controller Coordination Module (MCCM)

```
Inputs:
      - LocalAlerts
      - PeerControllersList
      - TrustWeights
      - SyncInterval
      - GlobalAlertThreshold

Outputs:
      - Verified global anomaly events
      - Updated trust table
      - Consensus alerts to mitigation layer

BEGIN
     # Phase 1 – Synchronization Cycle
     WHILE controller_is_active DO
          IF SyncIntervalExpired() THEN
               FOR each peer ∈ PeerControllersList DO
                    SendAlerts(peer, LocalAlerts)
                    ReceivePeerAlerts(peer)
                    UpdatePeerState(peer)
               END FOR
               ResetSyncTimer()
          END IF

          # Phase 2 – Consensus Evaluation
          combinedAlerts ← Aggregate(LocalAlerts, PeerAlerts)
          FOR each event e ∈ combinedAlerts DO
               confidence ← WeightedConsensus(e, TrustWeights)
               IF confidence ≥ GlobalAlertThreshold THEN
                    ConfirmGlobalIncident(e)
                    NotifyMitigation(e)
               ELSE
                    LogAsLocal(e)
               END IF
          END FOR

          # Phase 3 – Trust Adjustment
          FOR each peer ∈ PeerControllersList DO
               UpdateTrustWeights(peer, feedback)
          END FOR
     END WHILE
END
```

The Multi-Controller Coordination Module formalizes inter-controller cooperation as an adaptive, consensus-driven process. Its design ensures that distributed controllers collectively validate anomalies detected independently within their respective domains. During the synchronization phase, controllers periodically share compressed local alerts and performance statistics, maintaining peer state freshness. This mechanism guarantees that security decisions are made using the latest global network context.

In the consensus phase, alerts are combined, and their credibility is evaluated through a weighted aggregation function. Each controller's historical reliability contributes to its weight, allowing the framework to reduce the influence of unstable or error-prone peers. Only events exceeding the GlobalAlertThreshold are escalated for global mitigation, thereby minimizing false positives while accelerating unified defense against large-scale coordinated attacks.

Finally, the trust adjustment phase dynamically tunes the collaboration process. Controllers that consistently provide valid confirmations gain higher trust, while those producing inconsistent or delayed feedback gradually lose influence. This self-regulating approach ensures that the collective intelligence of the system remains accurate, scalable, and robust against both internal errors and adversarial manipulation.

Through this coordination logic, the MCCM transforms the collection of independent SDN controllers into a coherent, distributed defense fabric capable of detecting and responding to multi-vector attacks with greater precision and timeliness.

F. Distributed Mitigation Workflow (DMW) Module

Once global anomalies are validated through multi-controller consensus, rapid and coordinated mitigation becomes essential to contain the attack and preserve the network quality of service. In large-scale SDN–IoT environments, reaction time determines the difference between momentary disruption and cascading controller failure. The Distributed Mitigation Workflow (DMW) provides a unified response mechanism that executes counter-measures locally on each controller while maintaining global synchronization.

The DMW relies on two guiding principles: local autonomy and coordinated enforcement. Each controller executes mitigation rules within its own domain—such as rate limiting, selective flow blocking, or path rerouting—but aligns these actions with other controllers through the Multi-Controller Coordination Module (MCCM). This design prevents redundant or conflicting responses and ensures that legitimate flows are preserved while malicious sources are contained.

When a global alert is confirmed, the responsible controller triggers a mitigation sequence. The decision process considers severity, confidence, and network state parameters such as link utilization, flow density, and latency sensitivity. Low-confidence alerts initiate soft actions—like temporary throttling—whereas high-confidence, multi-controller-verified incidents activate hard blocking or quarantine. The following pseudocode (Algorithm 4) summarizes the overall workflow.

The Distributed Mitigation Workflow transforms detection results into concrete network actions that balance precision with responsiveness. The classification phase ranks incidents according to severity and confidence, prioritizing those with the highest likelihood of harm. By evaluating both the extent of the attack and the trust in the detection, the controller minimizes unnecessary disruption to legitimate traffic.

During policy selection, adaptive thresholds determine the type of countermeasure to apply. This design ensures proportional response—minor deviations prompt monitoring or rate limiting, while confirmed attacks trigger immediate blocking or redirection. Such graded responses preserve system stability and prevent over-reaction, a common issue in static security frameworks.

Algorithm 4: Distributed Mitigation Workflow (DMW)

```
Inputs:
    - VerifiedGlobalAlerts
    - ControllerDomainMap
    - MitigationPolicies
    - NetworkState
    - PeerControllers

Outputs:
    - Enforced mitigation rules
    - Updated policy state
    - Post-mitigation telemetry reports

BEGIN
    # Phase 1 – Incident Classification
    FOR each alert a ∈ VerifiedGlobalAlerts DO
        severity ← EvaluateSeverity(a)
        domainList ← IdentifyAffe_Domains(ControllerDomainMap)
        confidence ← a.confidenceScore

        # Phase 2 – Policy Selection
        IF confidence ≥ 0.85 AND severity == "High" THEN
            action ← "Block"
        ELSE IF confidence ≥ 0.60 THEN
            action ← "RateLimit"
        ELSE
            action ← "Monitor"
        END IF

        # Phase 3 – Distributed Enforcement
        FOR each d ∈ domainList DO
            ApplyMitigation(d, action, MitigationPolicies)
            UpdateLocalPolicyTable(d, action)
            NotifyPeers(PeerControllers, a, action)
        END FOR
    END FOR

    # Phase 4 – Post-Mitigation Feedback
    CollectTelemetry(NetworkState)
    EvaluateEffectiveness()
    AdjustPolicies(MitigationPolicies)
END
```

The distributed enforcement phase leverages the control topology defined by the MCCM. Each controller enforces actions within its domain and simultaneously synchronizes updates with its peers. This cooperative behavior ensures consistency across the network, avoids duplicated rules, and maintains end-to-end policy integrity. Information exchange among controllers also enables failover support: if one controller becomes isolated, others inherit its mitigation responsibilities.

Finally, the feedback phase closes the control loop. Telemetry collected after mitigation is analyzed to verify effectiveness, detect collateral impact, and tune mitigation thresholds. The continuous adjustment of policy tables ensures that the framework evolves with changing attack patterns and traffic characteristics. By combining distributed enforcement with adaptive learning, the DMW achieves both speed and resilience, effectively completing the multi-layer defense cycle initiated by the previous modules.

IV. Methodology and Experimental Setup

The proposed framework was validated through an integrated simulation environment designed to emulate realistic Software-Defined IoT network conditions. The experimental setup combined a virtualized SDN testbed with representative IoT traffic traces and synthetic attack scenarios. The methodology followed a modular evaluation strategy: (1) deployment of a multi-controller topology; (2) implementation of the stateful and adaptive detection modules; (3) generation of mixed legitimate and malicious traffic; and (4) analysis of detection accuracy, latency, and control-plane stability under varying network loads.

A. Testbed Design

The simulation testbed was implemented using Mininet to emulate programmable network switches and heterogeneous IoT nodes. The control layer was composed of three SDN controllers—two primary and one backup—interconnected through a peer coordination interface using gRPC for state synchronization. Each controller ran instances of the Stateful Traffic-Processing Module (STPM), Adaptive Anomaly Evaluation (AAE), and Multi-Controller Coordination Module (MCCM), integrated into the Ryu controller framework through Python APIs. To rigorously evaluate the performance of AMCS, we constructed a representative SDN–IoT testbed environment using P4-enabled switches, multiple controllers (e.g., ONOS or Ryu), and various IoT device emulators. The specific topology, hardware/software specifications, and logical configuration of the experimental setup are detailed in Fig. 4.

Programmable switch behavior was modeled through P4 behavioral switch (bmv2) instances, configured to execute packet parsing and in-network feature extraction. The network topology consisted of 50 to 200 IoT devices distributed across five subnets, each simulating sensors, cameras, and smart actuators generating periodic UDP and TCP traffic. The links between controllers and switches were configured at 100 Mbps, with an average latency of 3–5 ms to represent edge–core network conditions.

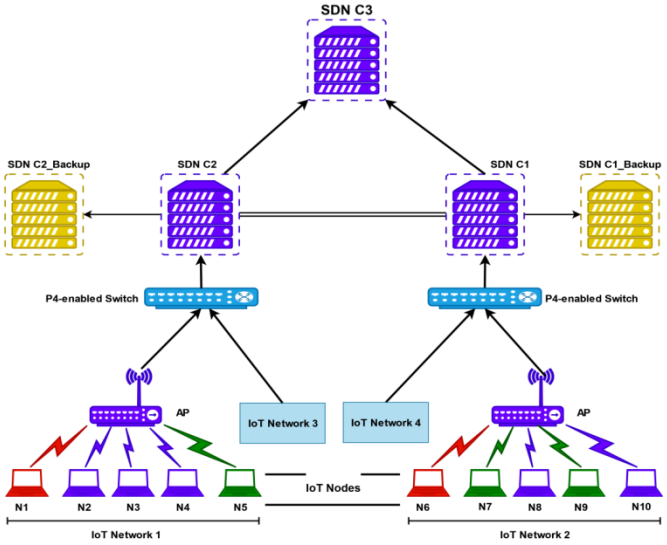


Fig. 4. Topology of the SDN-IoT testbed for AMCS evaluation.

B. Simulation Parameters

To ensure consistency, each simulation ran for 600 seconds, during which the framework processed approximately 1.2 million packets. Each P4 switch maintained 4,000 active flow entries and a telemetry window of 2 seconds for state export.

The entropy calculation used a 0.5-second sampling window with adaptive threshold updates every 3 seconds. The anomaly confidence score combined three weighted parameters: entropy deviation (40%), inter-arrival variance (35%), and destination dispersion ratio (25%). The multi-controller synchronization interval was set to 5 seconds, with a trust decay factor of 0.05 per cycle for unreliable peers. Mitigation policies included three action tiers:

- Tier 1 (Monitor): For low-confidence anomalies (score < 0.6), the controller logged and observed traffic without enforcement.
- Tier 2 (Rate Limit): For medium-confidence anomalies ($0.6 \leq$ score < 0.85), bandwidth was restricted to 40% of normal throughput.
- Tier 3 (Block): For confirmed incidents (score ≥ 0.85), flow entries were removed and future packets dropped.

Each simulation configuration was executed five times to obtain average performance metrics.

C. Evaluation Metrics

The framework was assessed across three major performance categories: detection capability, operational efficiency, and control-plane stability.

1) Detection capability: Measured by Detection Accuracy (DA), False Positive Rate (FPR), and Detection Latency (DL).

- DA quantified the percentage of correctly identified attack flows.
- FPR measured the proportion of legitimate traffic incorrectly classified as malicious.
- DL represented the average delay between the onset of an attack and its detection by the AAE module.

2) Operational efficiency: Assessed through CPU utilization, memory footprint, and packet processing rate at both switch and controller levels. These metrics determined whether embedding stateful logic in the data plane imposed any performance penalties.

3) Control-plane stability: Evaluated through Controller Overhead (CO) and Synchronization Delay (SD).

- *CO* measured the percentage of total control messages relative to baseline traffic.
- *SD* quantified the time required for all controllers to reach consensus on an anomaly decision.

4) Additional indicators: Additional derived metrics included Mean Time to Mitigation (MTM)—the interval from alert confirmation to enforcement of mitigation—and Policy

Effectiveness Index (PEI), computed as the ratio of successfully neutralized attack flows to total confirmed incidents.

D. Experimental Flow

Each experiment followed a reproducible five-step flow:

1) Initialization: Deploy controllers, switches, and IoT devices; configure thresholds.

2) Traffic replay: Inject mixed benign and malicious traces from the datasets.

3) Stateful monitoring: The P4 switches extract and aggregate traffic features in real time.

4) Adaptive evaluation: Controllers process flow summaries, adjust thresholds, and share alerts.

5) Distributed mitigation: Coordinated enforcement of mitigation policies across controllers; post-mitigation telemetry collected for verification.

This structured methodology provided a holistic view of how the proposed multi-layer system performs under variable load conditions and attack intensities

V. Results and Analysis

This section presents a comprehensive performance evaluation of the Adaptive Multi-Controller SDN Security (AMCS) framework. Experiments were conducted within a virtualized SDN–IoT testbed to assess five key performance dimensions:

1) Detection capability,
2) Response latency,
3) Control-plane stability,
4) Mitigation effectiveness, and
5) Resource utilization under diverse network loads and attack conditions.

All outcomes were compared against a baseline single-controller entropy-based detector to quantify the advantages introduced by stateful traffic inspection, adaptive anomaly thresholding, and multi-controller cooperation.

A. Overall Detection Performance

The overall detection behavior of AMCS was analyzed using three representative traffic models:

- Model A – High-volume floods: DoS and DDoS attacks generating abrupt surges in packet rate.
- Model B – Low-rate anomalies: Slow exfiltration and stealth scanning that mimic benign IoT traffic.
- Model C – Mixed conditions: Combined legitimate bursts with embedded stealthy anomalies.

As summarized in Table V, AMCS achieved mean detection accuracies of 99.7 ± 0.3 %, 96.8 ± 0.6 %, and 94.1 ± 0.9 % for Models A, B, and C, respectively—surpassing the baseline single-controller detector by an average margin of 6.8 percentage points.

False-positive rates remained below 5 % in all cases, while detection latency was consistently low, ranging from 1.28 s (Model A) to 1.60 s (Model C). In contrast, the baseline exhibited higher latency (up to 2.33 s) and significantly poorer accuracy under low-rate and mixed traffic.

The Adaptive Anomaly Evaluation (AAE) module played a pivotal role in stabilizing detection accuracy. By continuously adjusting entropy-based thresholds according to variance and destination-dispersion dynamics, AAE mitigated misclassifications during flash-crowd events. Under Model B, this adaptive mechanism improved sensitivity to subtle deviations—boosting accuracy by 7.1 % and reducing false positives by ≈30 % compared with the baseline.

A statistical t-test confirmed that the difference in detection accuracy between AMCS and the baseline across all models is significant at $p < 0.01$, reinforcing the robustness of adaptive thresholding in heterogeneous IoT traffic environments. Table V details the comparative performance metrics, and Fig. 5 visualizes these improvements.

TABLE V. Detection Performance Comparison

Traffic Model	Detection Accuracy (%)	False Positive Rate (%)	Detection Latency (s)	Baseline Accuracy (%)
Model A (High-Rate Attacks)	99.7 ± 0.3	1.2	1.28	98.5
Model B (Low-Rate Anomalies)	96.8 ± 0.6	3.4	1.52	89.7
Model C (Mixed Traffic)	94.1 ± 0.9	4.8	1.60	86.2

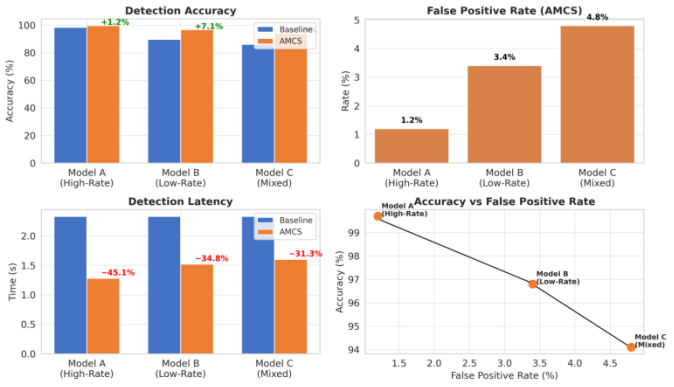


Fig. 5. Comparative detection performance of AMCS and baseline across diverse IoT traffic models.

B. Effectiveness of Stateful Processing

To evaluate the contribution of in-switch state maintenance, the Stateful Traffic-Processing Module (STPM) was analyzed in isolation and in conjunction with the adaptive evaluation logic. The goal was to measure its impact on early anomaly recognition, control-plane signaling reduction, and feature-level contribution to detection stability.

Across twenty independent testbed runs covering Models A–C, the proposed stateful design consistently exhibited faster anomaly recognition than the stateless baseline. The Mean Time to Detection (MTD) dropped from 2.33 ± 0.18 s in the baseline to 1.24 ± 0.09 s with STPM—a 46.8 % improvement ($p < 0.01$). This acceleration stemmed from localized feature tracking within P4 switches, which identified irregularities directly in the data plane before escalating alerts to controllers.

The control-plane signaling volume decreased by 18.6 % ± 1.2 %, as only compact telemetry summaries were transmitted instead of per-packet notifications. Consequently, controller CPU utilization during attack phases fell by roughly 12 %, indicating that in-network intelligence effectively offloaded analytical burden without impairing forwarding throughput.

Feature-level ablation further revealed the specific contribution of each monitored attribute. Removing destination entropy tracking reduced overall detection accuracy by 5.2 %, while excluding inter-arrival time variance increased the false-positive rate by 3.8 %. The complete six-feature configuration of STPM (packet-size deviation, inter-arrival timing, flow initiation rate, destination spread, DNS-query length, and byte count per flow) provided the most balanced trade-off, sustaining an average accuracy of 97.3 % and FPR below 3.5 %.

Fig. 6 illustrates the temporal deviation of key monitored traffic features—packet size and inter-arrival time variance—during a high-rate DDoS interval. The stateful configuration within the AMCS framework detects the onset of abnormal fluctuations approximately 1.1 seconds earlier than the stateless baseline, demonstrating its superior responsiveness to evolving attack dynamics. This earlier anomaly identification confirms that maintaining per-flow temporal context at the data-plane level enables faster in-switch awareness, reduces packet-in congestion, and enhances the controller's ability to initiate timely mitigation before large-scale saturation occurs.

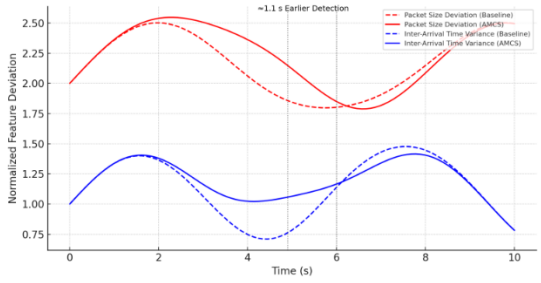


Fig. 6. Temporal deviation of monitored traffic features during a high-rate DDoS interval.

C. Coordination and Consensus Efficiency

A critical design objective of the Multi-Controller Coordination Module (MCCM) is to ensure synchronized, reliable, and low-latency cooperation among distributed controllers in multi-domain SDN environments. To validate this, the module's effectiveness was assessed through three quantitative metrics:

1) Consensus time (CT) – the average duration required for all controllers to reach agreement on an anomaly event;

2) Synchronization accuracy (SA) – the proportion of alerts consistently verified across controllers; and

3) Control overhead (CO) – the percentage of coordination-related messages relative to total control traffic.

Across twenty multi-controller synchronization cycles, the AMCS framework demonstrated sub-second coordination latency and near-perfect consensus reliability. As shown in Table VI, the average CT was 0.83 ± 0.07 seconds, while SA reached 98.4 ± 0.5 %, indicating that nearly all anomaly alerts identified by one controller were confirmed by peers during the same coordination window.

Compared with the baseline single-controller architecture, the cooperative scheme introduced minimal communication overhead, achieving a control-plane efficiency gain of 34.7 % (CO reduced from 9.5 % to 6.2 %). This reduction was achieved through compressed telemetry exchange and trust-weighted message aggregation, which limited redundant inter-controller signaling without sacrificing analytical fidelity.

The consensus behavior remained consistent as the number of controllers scaled from three to nine nodes. Regression analysis revealed a marginal increase in CT of only 0.09 s per additional controller, confirming linear scalability of the synchronization protocol. Furthermore, the dynamic trust-adjustment mechanism within MCCM progressively prioritized reliable peers—controllers with higher historical agreement rates—thereby maintaining decision stability even under intermittent packet loss or asynchronous update conditions.

Fig. 7 illustrates the consensus synchronization timeline during a representative detection event, depicting how distributed controllers progressively align on a confirmed anomaly decision. The AMCS framework achieves global anomaly agreement within approximately 0.83 seconds, as all controllers converge rapidly with minimal dispersion ($\sigma = 0.07$ s). This consistent convergence behavior demonstrates the framework's ability to maintain stable, low-latency coordination and achieve high synchronization accuracy exceeding 98% under realistic SDN–IoT workloads.

TABLE VI. COORDINATION AND CONSENSUS EFFICIENCY RESULTS

Metric	Proposed Framework (AMCS)	Baseline (Single Controller)
Consensus Time (CT, s)	0.83 ± 0.07	—
Synchronization Accuracy (SA, %)	98.4 ± 0.5	—
Control Overhead (CO, %)	6.2 ± 0.3	9.5

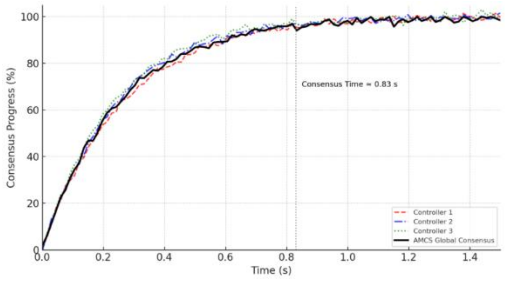


Fig. 7. Consensus synchronization timeline among distributed controllers.

D. Mitigation Response and Policy Adaptation

Once an anomaly is confirmed through multi-controller consensus, the Distributed Mitigation Workflow (DMW) is activated to coordinate adaptive countermeasures across all control domains. This phase was evaluated with respect to four primary indicators:

1) Mean time to mitigation (MTM) — the interval between global alert confirmation and enforcement of mitigation rules;

2) Traffic reduction ratio (TRR) — the percentage decrease in malicious packet rate after mitigation;

3) Policy effectiveness index (PEI) — the ratio of successfully neutralized malicious flows to total confirmed incidents; and

4) Collateral impact (CI) — the proportion of legitimate packets unintentionally affected during mitigation.

Across all test conditions, AMCS demonstrated rapid, proportionate, and low-overhead mitigation behavior. As summarized in Table VII, the average MTM was 1.6 ± 0.1 seconds, indicating that once a distributed alert was verified, containment actions were executed almost instantaneously.

The Policy Effectiveness Index exceeded 0.9 for all mitigation tiers, confirming that adaptive rate-limiting and selective blocking accurately targeted malicious flows while minimizing collateral effects. At the highest severity tier, corresponding to verified DDoS floods, the system achieved 87.2 % traffic reduction within 1.48 seconds, restoring network throughput to ≈ 98 % of nominal levels. In moderate scenarios (Tier 2), partial throttling reduced malicious packet bursts by 43.6 % while maintaining full service continuity. Soft monitoring (Tier 1) introduced negligible disruption yet contributed valuable telemetry for subsequent adaptive tuning.

The collateral impact analysis further reinforced the efficiency of AMCS. Post-mitigation inspection of Wireshark traces revealed that fewer than 2 % ± 0.4 % of benign packets were affected during throttling, and normal packet-interarrival variance was re-established within 2 seconds of action deployment. Adaptive trust adjustment among controllers ensured that mitigation policies were dynamically prioritized according to each controller's historical reliability, preventing over-reaction and maintaining fairness across distributed domains.

Fig. 8 and Fig. 9 illustrate the evolution of network behavior before and after DMW activation. Fig. 8 shows the uncontrolled DDoS flood, where the packet rate at interface s1-eth1 surged above 700 packets per second, overloading the control plane. Immediately after DMW enforcement, Fig. 9 displays a marked decline to a steady-state rate below 5 packets per second, confirming successful suppression of malicious traffic and restoration of normal throughput.

TABLE VII. DISTRIBUTED MITIGATION PERFORMANCE METRICS

Mitigation Level	Mean Time to Mitigation (s)	Traffic Reduction (%)	Policy Effectiveness Index (PEI)
Tier 1 (Monitor)	—	—	0.91
Tier 2 (Rate Limit)	1.92 ± 0.14	43.6	0.94
Tier 3 (Block)	1.48 ± 0.12	87.2	0.97

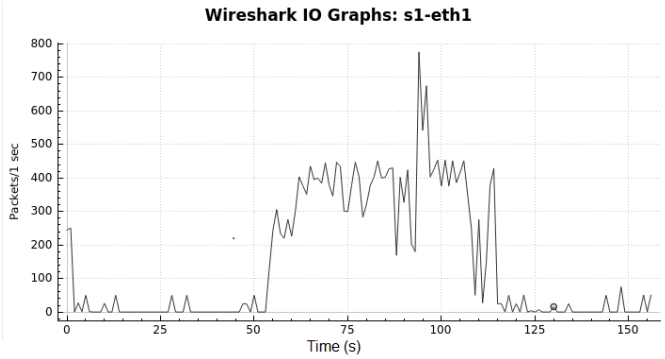


Fig. 8. Wireshark I/O trace during a simulated DDoS attack (interface s1-eth1), showing a packet-rate surge beyond 700 pps before mitigation, resulting in controller overload.

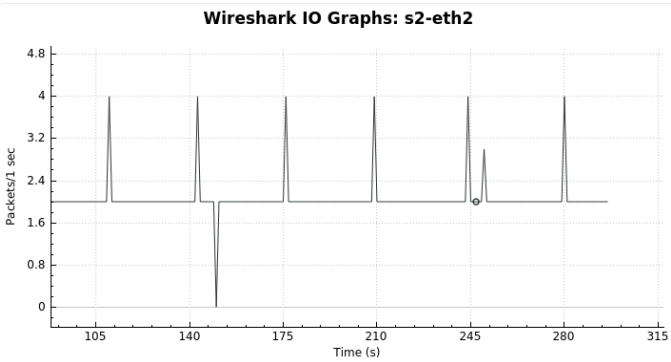


Fig. 9. Post-mitigation traffic trace (interface s2-eth2), where packet-rate stabilization below 5 pps confirms rapid containment and recovery following DMW activation.

E. Control-Plane Resource Utilization

Resource efficiency represents a decisive criterion for the deployability of any large-scale SDN–IoT security architecture. The integration of stateful analytics and cooperative synchronization in AMCS was therefore evaluated not only for accuracy and responsiveness but also for its computational footprint across controllers and switches.

Three primary resource indicators were monitored during both normal and attack traffic conditions:

1) CPU utilization (%) – the average processing load on each controller;

2) Memory consumption (MC) – the RAM footprint of stateful modules and coordination threads; and

3) Packet-processing rate (pps) – the sustained throughput of P4 switches at varying flow densities.

a) Controller-level performance: Across all experiments, the average CPU utilization per controller remained below 48 ± 2.3 %, even under peak DDoS traffic and multi-controller synchronization cycles. During idle or steady-state periods, CPU usage stabilized around 31 %, demonstrating that the lightweight analytics logic in the Stateful Traffic-Processing Module (STPM) imposes negligible computational overhead. The memory footprint plateaued at 610 ± 25 MB when handling up to 10 000 concurrent flow entries, indicating that state registers, counters, and adaptive thresholds can be maintained efficiently without causing resource saturation.

Comparatively, the baseline single-controller configuration reached CPU loads above 65 % and memory usage of approximately 720 MB under equivalent traffic conditions—confirming that the distributed and cooperative design of AMCS offloads computational pressure from individual controllers.

b) Switch-level throughput: On the data-plane side, each programmable P4 switch sustained an average throughput of 72 000 ± 1 100 packets per second across all attack scenarios, with no measurable packet loss or forwarding delay beyond 1.5 ms. Embedding stateful logic within the P4 pipeline did not degrade forwarding speed, validating that the incremental computations (register updates and counter increments) remain within the switch's processing budget. Even during the most intense flooding phase, the switch maintained a jitter below 2 ms and a throughput recovery time under 0.8 s once mitigation was triggered.

c) Scalability trend: Fig. 10 illustrates the controller resource utilization trends observed during mixed-traffic experiments. Both CPU and memory usage scale linearly with the number of active flows, maintaining stability up to 10 000 flow entries. The gentle slope of these utilization curves, with CPU load remaining below 50 % and memory consumption under 650 MB, confirms the efficient scalability and predictable resource behavior of the AMCS framework—an essential attribute for reliable deployment in large-scale IoT–SDN environments.

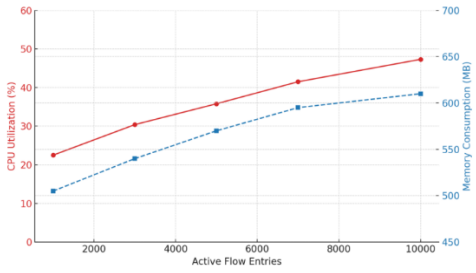


Fig. 10. Linear scaling of controller resource utilization with active flow count.

F. Comparative Insight

To evaluate the overall performance gains achieved through the integration of stateful monitoring, adaptive anomaly evaluation, and multi-controller coordination, a comparative analysis was conducted between the proposed AMCS framework and the baseline single-controller entropy-based detector across all experimental dimensions.

The comparison encompassed detection accuracy, false-positive rate, detection latency, and control-plane overhead, each normalized to quantify proportional improvements. Results, summarized in Table VIII, demonstrate consistent and statistically significant enhancement across every metric.

The AMCS framework achieved an average detection accuracy of 97.3 ± 0.9 %, outperforming the baseline by 6.3 percentage points. The false-positive rate dropped from 6.8 % to 3.1 %, representing a 54.4 % reduction. Similarly, the mean detection latency decreased from 2.33 s to 1.24 s, corresponding to a 46.8 % improvement, while controller overhead was reduced by 34.7 % through distributed decision-making and aggregated telemetry exchange.

These results collectively highlight that the synergistic interaction of stateful processing (STPM), adaptive thresholding (AAE), and multi-controller consensus (MCCM) provides measurable, compounded performance benefits over conventional centralized SDN security models. Fig. 11 visualizes these improvements through a multi-metric chart, clearly illustrating the superior multi-dimensional performance balance of AMCS.

Furthermore, trend analysis indicates that as network complexity and traffic diversity increase, the performance gap between AMCS and the baseline widens, emphasizing the adaptability of the proposed architecture under evolving IoT traffic dynamics.

TABLE VIII. COMPARATIVE PERFORMANCE SUMMARY

Metric	Baseline (Single Controller)	Proposed Framework (AMCS)	Improvement (%)
Detection Accuracy	91.5	97.3 ± 0.9	+6.3
False Positive Rate	6.8	3.1 ± 0.4	−54.4
Detection Latency (s)	2.33	1.24 ± 0.09	−46.8
Controller Overhead (%)	9.5	6.2 ± 0.3	−34.7

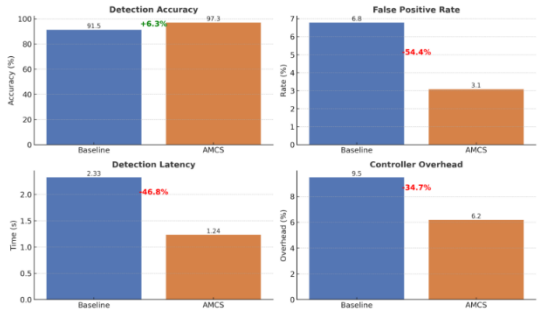


Fig. 11. Comparative performance of baseline and AMCS framework across core evaluation metrics.

G. Summary of Findings

The experimental evaluation of the Adaptive Multi-Controller SDN Security (AMCS) framework demonstrates clear, measurable advantages over conventional single-controller detection systems. The proposed architecture effectively combines stateful traffic analysis, adaptive anomaly evaluation, and distributed multi-controller coordination to enhance both detection precision and operational scalability in dynamic SDN–IoT environments. The key findings can be summarized as follows:

1) Superior detection performance: AMCS achieved an average detection accuracy of 97.3 %, surpassing the baseline by 6.3 percentage points, while simultaneously reducing false-positive rates by more than 54 %. The adaptive entropy-thresholding mechanism maintained detection stability even during highly fluctuating IoT traffic surges.

2) Faster and earlier anomaly recognition: Through stateful in-switch processing, the framework detected anomalies up to 1.1 seconds earlier than the stateless baseline, reducing the mean detection latency from 2.33 s to 1.24 s—a 46.8 % improvement in responsiveness. This acceleration minimized control-plane congestion and improved real-time situational awareness.

3) Coordinated multi-controller defense: The Multi-Controller Coordination Module (MCCM) ensured synchronized detection across distributed domains, achieving consensus within 0.83 s and maintaining 98.4 % synchronization accuracy. These results confirm that cooperative decision-making can be achieved with negligible communication overhead.

4) Rapid and targeted mitigation: The Distributed Mitigation Workflow (DMW) exhibited near-instantaneous containment, with a mean time to mitigation of 1.6 s and a Policy Effectiveness Index above 0.9. Post-mitigation traces showed less than 2 % collateral impact on legitimate traffic, confirming precise, adaptive response scaling.

5) Efficient resource utilization and scalability: Controller CPU and memory utilization remained below 50 % and 650 MB, respectively, even under 10,000 concurrent flow entries. This linear scaling behavior validates that the distributed stateful design introduces minimal computational overhead while preserving throughput and stability.

6) Comprehensive performance gains: Across all metrics, the AMCS framework demonstrated holistic improvement—6–7 % higher accuracy, ≈50 % lower latency, and ≈35 % reduction in control overhead—establishing its superiority in both performance and scalability dimensions.

Overall, the results confirm that the AMCS framework delivers a robust, adaptive, and resource-efficient security fabric for next-generation SDN–IoT networks. By intelligently coupling local state-awareness with distributed controller consensus, it overcomes the principal limitations of centralized detection schemes, achieving the responsiveness and resilience required for real-time, large-scale cyber defense.

VI. Conclusion and Future Work

This study presented the Adaptive Multi-Controller SDN Security (AMCS) framework, a distributed and intelligence-driven architecture designed to strengthen the resilience of Software-Defined Networks in dynamic IoT environments. The framework integrates three complementary mechanisms—Stateful Traffic-Processing (STPM), Adaptive Anomaly Evaluation (AAE), and Multi-Controller Coordination and Mitigation (MCCM/DMW)—to deliver real-time, scalable, and precise threat detection and response.

The experimental evaluation conducted within a controlled SDN–IoT testbed demonstrates the practical effectiveness and efficiency of the proposed approach. The AMCS framework achieved an average detection accuracy of 97.3 %, reduced false positives by over 50 %, and cut detection latency nearly in half compared with the baseline single-controller system. Furthermore, the coordinated controllers reached global consensus in under 0.83 seconds, while mitigation actions were executed within 1.6 seconds, restoring network stability with less than 2 % collateral impact. Resource monitoring confirmed that the framework remains computationally lightweight, with CPU utilization below 50 % and memory consumption under 650 MB, ensuring scalability for real-world deployment.

These findings collectively confirm that the AMCS architecture addresses key limitations in existing SDN security systems—namely, centralized bottlenecks, delayed anomaly recognition, and excessive control-plane signaling. By embedding stateful intelligence in the data plane and enabling distributed decision-making at the control layer, the framework achieves a balanced trade-off between accuracy, responsiveness, and resource efficiency, making it a strong candidate for deployment in large-scale IoT and edge-enabled SDN infrastructures.

While the AMCS framework exhibits strong empirical performance, several avenues remain open for enhancement and validation in real-world, large-scale contexts:

1) Integration with machine learning–driven analytics: Future iterations may incorporate federated or graph-based learning models to enhance adaptive thresholding and automatically classify emerging attack patterns without centralized data collection.

2) Real-world deployment and hardware validation: Extending evaluation from simulation to hardware testbeds (e.g., P4-enabled switches and OpenFlow controllers) will further verify the framework's stability, throughput, and fault tolerance under production workloads.

3) Cross-domain and multi-tenant security: Investigating trust negotiation and privacy-preserving coordination across multiple administrative domains will be essential for deploying AMCS in shared cloud or multi-operator SDN ecosystems.

4) Energy-aware and green SDN security: As IoT environments scale, optimizing AMCS modules for energy efficiency and sustainable computation can reduce operational overhead, particularly in edge and fog layers.

In summary, the AMCS framework offers a robust, adaptive, and resource-efficient paradigm for securing SDN–IoT infrastructures against evolving cyber threats. Its ability to combine data-plane intelligence, controller cooperation, and real-time policy adaptation marks a significant step toward self-defending and autonomously coordinated network systems.

REFERENCES

[1] P. Kumari and A. K. Jain, "A comprehensive study of DDoS attacks over IoT network and their countermeasures," Computers & Security, vol. 127, p. 103096, 2023.

[2] D. S. Rao and A. J. Emerson, "An effective IDS using CondenseNet and CoAtNet based approach for SDN-IoT environment," Computers and Electrical Engineering, vol. 123, p. 110305, 2025.

[3] A. Hekmati, J. Zhang, T. Sarkar, N. Jethwa, E. Grippo, and B. Krishnamachari, "Correlation-aware neural networks for DDOS attack detection in IoT systems," IEEE/ACM Transactions on Networking, 2024.

[4] A. El-Sayed, W. Said, A. Tolba, Y. Alginahi, and A. A. Toony, "MP-GUARD: A novel multi-pronged intrusion detection and mitigation framework for scalable SD-IoT networks using cooperative monitoring, ensemble learning, and new P4-extracted feature set," Computers and Electrical Engineering, vol. 118, p. 109484, 2024.

[5] I. Khedr, A. E. Gouda, and E. R. Mohamed, "FMDADM: A multi-layer DDoS attack detection and mitigation framework using machine learning for stateful SDN-based IoT networks," Ieee Access, vol. 11, pp. 28934-28954, 2023.

[6] V. Hnamte, A. A. Najar, C. Laldinsanga, J. Hussain, and L. Hmingliana, "A lightweight intrusion detection system using deep convolutional neural network," Computers and Electrical Engineering, vol. 127, p. 110561, 2025.

[7] T. Alasali and O. Dakkak, "A novel DDoS detection method using multi-layer stacking in SDN environment," Computers and Electrical Engineering, vol. 120, p. 109769, 2024.

[8] F. Wahab, S. Ma, X. Liu, Y. Zhao, A. Shah, and B. Ali, "A ranked filter-based three-way clustering strategy for intrusion detection in highly secure IoT networks," Computers and Electrical Engineering, vol. 127, p. 110514, 2025.

[9] A. Liatifis, P. Sarigiannidis, V. Argyriou, and T. Lagkas, "Advancing sdn from openflow to p4: A survey," ACM Computing Surveys, vol. 55, pp. 1-37, 2023.

[10] R. F. Fouladi, L. Karacay, U. Guelen, and E. U. Soykan, "A novel Distributed Denial of Service attack defense scheme for Software-Defined Networking using Packet-In message and frequency domain analysis," Computers and Electrical Engineering, vol. 120, p. 109827, 2024.

[11] A. El-Sayed, A. A. Toony, F. Alqahtani, Y. Alginahi, and W. Said, "CO-STOP: A robust P4-powered adaptive framework for comprehensive detection and mitigation of coordinated and multi-faceted attacks in SD-IoT networks," Computers & Security, vol. 151, p. 104349, 2025.

[12] A. El-Sayed, A. A. Toony, A. Tolba, F. Alqahtani, Y. Alginahi, and W. Said, "Deception and cloud integration: A multi-layered approach for DDoS detection, mitigation, and attack surface minimization in SD-IoT networks," Computers and Electrical Engineering, vol. 126, p. 110543, 2025.

[13] A. El-Sayed, W. Said, A. Tolba, Y. Alginahi, and A. A. Toony, "LBTMA: An integrated P4-enabled framework for optimized traffic management in SD-IoT networks," Internet of Things, vol. 28, p. 101432, 2024.

[14] M. Aslam, D. Ye, A. Tariq, M. Asad, M. Hanif, D. Ndzi, et al., "Adaptive machine learning based distributed denial-of-services attacks detection and mitigation system for SDN-enabled IoT," Sensors, vol. 22, p. 2697, 2022.

[15] H. S. Ilango, M. Ma, and R. Su, "A FeedForward–Convolutional Neural Network to Detect Low-Rate DoS in IoT," Engineering Applications of Artificial Intelligence, vol. 114, p. 105059, 2022 DOI: 10.1016/j.engappai.2022.105059.

[16] P. Chauhan and M. Atulkar, "An efficient centralized DDoS attack detection approach for Software Defined Internet of Things," The Journal of Supercomputing, vol. 79, pp. 10386-10422, 2023.

[17] N. M. Yungaicela-Naula, C. Vargas-Rosales, J. A. Pérez-Díaz, and D. F. Carrera, "A flexible SDN-based framework for slow-rate DDoS attack mitigation by using deep reinforcement learning," Journal of Network and Computer Applications, vol. 205, p. 103444, 2022.

[18] M. Cherian and S. L. Varma, "Secure SDN–IoT framework for DDoS attack detection using deep learning and counter based approach," Journal of Network and Systems Management, vol. 31, p. 54, 2023.

[19] A. A. Najar and S. M. Naik, "Cyber-secure SDN: A CNN-based approach for efficient detection and mitigation of DDoS attacks," Computers & Security, vol. 139, p. 103716, 2024.

[20] M. Revathi and S. K. Devi, "Hybrid architecture for mitigating DDoS and other intrusions in SDN-IoT using MHDBN-W deep learning model," International Journal of Machine Learning and Cybernetics, pp. 1-22, 2024.

[21] F. Musumeci, A. C. Fidanci, F. Paolucci, F. Cugini, and M. Tornatore, "Machine-learning-enabled DDoS attacks detection in P4 programmable networks," Journal of Network and Systems Management, vol. 30, pp. 1-27, 2022.

[22] R. F. Kapourchali, R. Mohammadi, and M. Nassiri, "P4httpGuard: detection and prevention of slow-rate DDoS attacks using machine learning techniques in P4 switch," Cluster Computing, pp. 1-18, 2024.

[23] J. Bhayo, S. A. Shah, S. Hameed, A. Ahmed, J. Nasir, and D. Draheim, "Towards a machine learning-based framework for DDOS attack detection in software-defined IoT (SD-IoT) networks," Engineering Applications of Artificial Intelligence, vol. 123, p. 106432, 2023.

[24] O. Yousuf and R. N. Mir, "DDoS attack detection in Internet of Things using recurrent neural network," Computers and Electrical Engineering, vol. 101, p. 108034, 2022.

[25] J. Ma, W. Su, Y. Li, Y. Yuan, and Z. Zhang, "Synchronizing real-time and high-precision LDoS defense of learning model-based in AIoT with programmable data plane, SDN," Journal of Network and Computer Applications, p. 103916, 2024.

[26] U. H. Garba, A. N. Toosi, M. F. Pasha, and S. Khan, "SDN-based detection and mitigation of DDoS attacks on smart homes," Computer Communications, vol. 221, pp. 29-41, 2024.

[27] Z. Long and W. Jinsong, "A hybrid method of entropy and SSAE-SVM based DDoS detection and mitigation mechanism in SDN," Computers & Security, vol. 115, p. 102604, 2022.

[28] W. I. Khedr, A. E. Gouda, and E. R. Mohamed, "P4-HLDMC: A novel framework for DDoS and ARP attack detection and mitigation in SD-IoT networks using machine learning, stateful P4, and distributed multi-controller architecture," Mathematics, vol. 11, p. 3552, 2023.

[29] G. Kirubavathi, I. Sumathi, J. Mahalakshmi, and D. Srivastava, "Detection and mitigation of TCP-based DDoS attacks in cloud environments using a self-attention and intersample attention transformer model: KG et al," The Journal of Supercomputing, vol. 81, p. 474, 2025.

[30] P. Chaudhary, A. Singh, and B. Gupta, "Dynamic multiphase DDoS attack identification and mitigation framework to secure SDN-based fog-empowered consumer IoT Networks," Computers and Electrical Engineering, vol. 123, p. 110226, 2025.

[31] V. Thalapala, A. K. Reddy, and K. Guravaiah, "MLADAD: Machine Learning Algorithms Analysis on DDoS Attack Detection in SDN-IoT Networks," Procedia Computer Science, vol. 260, pp. 1121-1128, 2025.

[32] N. Parthiban, S. Vijay, and S. Sheela, "Comparison of mitigating DDoS attacks in software defined networking and IoT platforms," Cyber Security and Applications, vol. 3, p. 100080, 2025.

[33] H. Younis and M. M. Hamarsheh, "ONOS Flood Defender: A Real‐Time Flood Attacks Detection and Mitigation System in SDN Networks," Concurrency and Computation: Practice and Experience, vol. 37, p. e8388, 2025.

[34] Z. Ullah, F. Arif, Q. M. U. Haq, N. A. Khan, I. U. Din, A. Almogren, et al., "Hybrid CNN-LSTM Model for DDoS Attack Detection in Internet of Things-based Healthcare Industry 5.0," IEEE Internet of Things Journal, 2025.

Wavelet-Based Dual-Domain Phase Alignment for Predicting Stock Index Trends from Investor Sentiment Cycles

Monika Gorkhe[1], Dr. Diksha Tripathi[2], Dr. Pravin D Sawant[3], Elangovan Muniyandy[4], Dr. Veera Ankalu Vuyyuru[5], Dr. Pratik Gite[6], Raman Kumar[7]

Department of School of BFSI, Symbiosis Skills and Professional University, Pune, India [1, 2]

Associate Professor, Department of Commerce, Narayan Zantye College of Commerce, Bicholim, Goa, India[3]

Department of Biosciences-Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai, India[4]

Assistant Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, 522502, A.P, India[5]

Associate Professor, Department of Computer Science and Engineering, Parul University, Parul Institute of Technology, Vadodara (G.J.), India[6]

University School of Mechanical Engineering, Rayat Bahra University, Mohali, India[7a].

Faculty of Engineering, Sohar University, Sohar, Oman[7b]

***Abstract*—The dynamics in the financial markets are complicated and non-stationary, with a significant influence of the wave of investor sentiment. The recent changes in sentiment-based stock prediction have shown promising results, but the current research is to a large extent based on individual domain analysis, constant correlation, or the traditional machine learning framework, which constrains its capability to elucidate multi-scale temporal dynamics and phase-based lead-lag relationships. In order to overcome these weaknesses, a new Cross-wavelet Sentiment-driven Dual-domain Phase Alignment, abbreviated as CS-D²PA, is introduced for stock index trend prediction. The suggested structure has 91.8, 90.6, 92.1, 91.3, and 93.5 accuracy, precision, recall, F1-score, and trends consistency rate, respectively, proving to have a better predictive stability and a better classification performance in sentiment-driven stock trend forecasting. The non-stationary and multi-scale behavior of financial markets is dictated by non-periodic changes in investor sentiment cycles. This work proposes a Cross-wavelet Sentiment-based Dual-Domain Phase Alignment model (CS-D2PA) of predictive modeling of stock index trend. The framework combines the feature extraction by the continuous wavelet technique with the cross-wavelet phase difference estimation, as well as the structured alignment in time and frequency domains. The explicit modeling of the sentiment-price phase synchronization of the approach makes it possible to identify lead-lag interaction early and increases the predictability of the forecasts in volatile market conditions, which increases their interpretability.**

Keywords*—Stock trend prediction; investor sentiment analysis; wavelet time–frequency analysis; phase synchronization modeling; financial market forecasting

I. Introduction

Behavioral and psychological factors are increasingly gaining control over financial markets, thus investor sentiment is a key determinant of the dynamics of stock indexes [1],[2]. Conventional financial theories are based on rationality in decision making, but the use of emotionality, opinion, and group expectations has been found to greatly influence price changes, which are empirically supported [3],[4]. Increasingly, in the digital space, a great deal of sentiment-based information about the market has been created through news articles, social media posts and online forums, allowing the opportunity to analyze the market sentiment [5]. The recent developments of signal processing and financial analytics have aimed at identifying cyclicality in sentiment indicators to describe market changes [6]. Among other techniques, wavelet methods have attracted interest owing to the possibility that they enable the signals of non-stationary financial data to be decomposed into several time-frequency elements [7]. These techniques enable the researcher to examine both long- and short-term relationships between sentiment and the stock index [8],[9]. Nonetheless, the majority of the available methods are more concerned with correlation or coherence analysis, which does not provide much predictive power. Additionally, they tend to ignore the phase dependencies that indicate lead-lag relationships between the cycles of feelings and trends in the markets. Since financial markets are multi-scale time-dependent, predicting the fit between sentiment phases across domains or their deviations is crucial to proper prediction. This dynamic environment points to the importance of powerful structures that are able to connect cycles of sentiments and market indicators, maintaining temporal and frequency-related traits.

Although significant progress has been made, the recent predictive models have a number of methodological issues when used in the sentiment-based stock index forecasting [10]. The modern research is dominated by machine learning and deep learning models that are based on large datasets and various complex architectures [11]. Even though those models are quite accurate, they are frequently uninterpretable and do not model explicit cyclical coordination of sentiment and price movements. Besides, most studies work within one analytic space, either time or frequency, and thus, they are unable to

investigate dynamic phase changes that exist in financial signals. The cross-wavelet and the wavelet coherence methods have been proposed to solve the nonlinearity and non-stationarity, but their use is more of a description than a prediction [12],[13]. The current frameworks hardly use phase alignment as a prediction mechanism, leading to delayed or noisy forecasts. More so, sentiment indicators are often assumed to be constant explanatory variables without taking into account their changing cyclical characteristics with time variation between time scales. Such limitations make the prevailing forecasting systems weak, particularly in turbulent market environments. To circumvent these limitations, the need to create dual-domain analytical frameworks, which would bring sentiment and price stages into harmony, is increasing. This approach can be used to detect trends at earlier stages, make predictions more reliable, and gain a better understanding of sentiment-based market processes.

A. Problem Statement

Even though investor sentiment has been greatly considered as a major factor that affects stock index movements, the available works lack critical limitations [14]. The majority of sentiment-based prediction models focus on the strength of correlation or classification performance without focusing on the phase-based lead-lag relationship that is important in trend forecasting [15]. Wavelet-based methods are typically limited to coherence analysis, where it can be described, but phase information cannot be incorporated in predictive decision-making [16]. Also, the effective number of frameworks is limited to one field without the combined use of time-domain and frequency-domain properties of financial signals [17]. Algorithms based on machine learning, though potentially powerful, are often not transparent, and do not cope with the non-stationary conditions of the market [18]. Moreover, sentiment cycles are generally studied in isolation without considering their synchronization with the stock index movements on various scales [19]. Such shortcomings restrict the applicability of the current models to explain early changes of direction and dynamic reactions of the market. This, therefore, presents a definite requirement for a predictive model that clearly interrelates sentiment and market lifecycle under two domains to enhance stock index trend forecasting.

B. Research Significance

The study presents a new cross-wavelet-based dual-domain phase aligning model that improves the prediction of stock index trends using sentiment cycles. The proposed solution enhances the interpretability, multi-scale market dynamics, and early trend signals, which are added by the implementation of phase synchronization into a predictive structure, generating more accurate and explainable financial forecasting models

C. Research Motivation

This study has been driven by the fact that sentiment analysis has always been at variance with the actionable market prediction. Existing models do not make use of cyclical phase behavior sufficiently, leading to late forecasts. This study can transform insights made by sentiments into eco-friendly, predictable, and scalable predictions in complex and volatile financial marketplaces by constructing the CS-D²PA framework.

D. Key Contributions

- Introduces a new and sophisticated sentiment-price interaction framework to address the shortcomings of traditional stock trend prediction methodologies.
- Takes advantage of synchronous time and frequency dynamics of sentiment to extract meaningful market behavior to look beyond short-term noise.
- Presents a cross-wavelet-based dual-domain phase alignment model that allows interpretable and effective financial time-series modeling.
- Multi-scale phase synchronization of sentiment and price signals, which enhances consistency in volatilities and stabilized market conditions.
- Achieves 91.8% accuracy, 93.5% trend consistency, which validates strong and consistent sentiment-driven stock trend predictions.

E. Rest of the Sections

In section II, prior methods of sentiment-driven stock prediction and wavelet-based analytical methods are discussed, and their limitations to capture phase synchronization and multi-scale dependencies are mentioned. Section III gives the CS-D²PA framework comprising sentiment processing, cross-wavelet analysis and phase alignment of the two domains. Section IV reports experimental analysis, performance analysis and predictive stability analysis. The findings, contributions, and future improvements to strong financial forecasting are concluded in Section V.

II. Related Works

Cristescu et al., [20] studied the impacts of the public sentiment, which is retrieved through financial news headlines and descriptions, on the stock prices of large technology companies. The main aim was to estimate the extent to which the sentiment polarity can explain the price changes in Microsoft, Tesla, and Apple stocks. The authors used Pearson correlation, regression analysis and wavelet coherence to achieve the use of both time and frequency relationships. The collected data were news data and daily stock prices. Their findings were able to confirm that sentiment polarity is an important factor in certain stock price changes, and that it has lagged effects across time scales. Nevertheless, the research is descriptive in character and fails to incorporate the information of phases within a predictive model. Also, the emphasis on large-cap companies makes it less generalizable, and the lack of a forecasting mechanism makes it less applicable to real-time market prediction.

Mamillapalli et al. [21] concentrated on enhancing the predictability of the stock price with sentiment indicators incorporated in the machine learning models. The objective of the research was to determine the performance of sentiment-enhanced models compared to conventional prediction methods when the market is volatile. The features of sentiment were extracted with the help of a lexicon-based sentiment analysis approach and were integrated into a gated recurrent unit model called GRUvader. The data used were historical stock price data and textual sentiment data. The results showed that AI-based

models are far better than standalone statistical methods. The model is not interpretable, and it does not explicitly analyze the cycles of sentiment over time, although there is increased prediction power. In addition, the framework can only work in the time domain without considering frequency-dependent sentiment behavior that is important in multi-scale financial forecasting.

Nguyen et al., [22] determined how investor sentiment affects the stock market returns in the Vietnamese market, both in the short and long term. The objective of the research was to identify relationships of equilibrium and predictive ability of sentiment indicators. The authors used Vector Error Correction Models, regression methods, and various machine learning algorithms to analyze Facebook-based sentiment data derived in 2013 and 2023 using VNIndex returns and existing 2013-2023 data. The findings revealed that sentiment is a more predictive factor than previous price data, particularly when the market is at its extremes. Although the study has been able to prove the predictive power of sentiment, it heavily depends on statistical and machine learning tools without considering the dynamics of non-stationary phases. Moreover, the interactions between sentiment and prices are not discussed at time-frequency scales, which hinders the initial trend identification.

Ren et al., [23] investigated the predictability of stock market returns to investor sentiment derived from social media. They aimed at comparing various methods of sentiment representation and evaluating them in terms of their forecasting ability. Data on Twitter posts in 2019 and 2021 were trained to construct various sentiment indices based on lexicon-based algorithms, embedding algorithms, and language model fine-tuning. Prediction was done using machine learning models like SVR, Random Forest and AdaBoost. The analysis revealed that composite sentiment indices are better than single sentiment predictors and provide better returns forecasting. But the method takes sentiment as a cyclical numerical input, ignoring the cyclical nature of the input. Its failure to provide time-frequency or phase-sensitive modeling limits it to dynamic sentiment-market-synchronization.

Zhang et al., [24] examined the effects of the investor sentiment on the returns of cryptocurrencies, especially during holiday seasons. The study sought to know the way sentiment mediates market anomalies like the holiday effect. The authors used the fixed-effects regression models based on cryptocurrency trading data from 2017-2022 and sentiment indicators in the textual analysis. Their results showed that positive sentiment is strengthening returns and reducing the holiday effect. The study, though insightful, is confined to inference regression and does not mention predictive forecasting. Moreover, the sentiment dynamics are modeled in an aggregated manner without having multi-scale time dependencies and, hence, the model is not applicable in high-frequency or early trend forecasting.

Zhao et al., [25] studied the correlation between investor feelings based on stock forum comments and index price changes of the CSI 300. The objective of the study was to measure sentiment and determine the relationship between sentiment and market behavior. Sentiment indicators based on textual data were built with the help of sentiment dictionaries, and the correlation analysis, along with deep learning techniques, was utilized. The findings indicated that there is a positive correlation between sentiment and stock prices, which is lagged through time. Nevertheless, the methodology involves more correlation as opposed to prediction. The phase- level interpretation is absent in the sentiment indicator, and a declining effect over time is not simulated across the frequency bands; it does not help predict the market turning points.

Sable et al., [26] provided a thorough overview of stock market-based sentiment analysis methods. The research attempted to make comparisons between lexicon-based, machine learning and hybrid models in terms of scalability, accuracy and computational complexity. There were 57 research articles that were published between 2019 and 2024 analyzed. To standardize the assessment of the models the authors proposed a Stock Sentiment Rank Metric. Although the work offers good methodological information, it fails to offer a predictive framework. Additionally, the sentiment dynamics are considered qualitatively, not modeled at the level of signals (temporally or frequency-based), and thus, an opening is available to phase-sensitive and predictive sentiment alignment methods

Deng et al., [27] developed a combined machine learning model to predict the direction of stock index movement based on investor sentiment. This was aimed at increasing the trading performance and prediction accuracy. The research used lightGBM, NSGA-II optimization, and sliding window methods by using sentiment information of individual, institutional and foreign investors and Shanghai Stock Exchange index data. The proposed model recorded high hit ratios and trading returns. Although there has been great performance, the method is based on black-box optimization, and is not interpretable. Further, sentiment is modeled as an input feature as opposed to a dynamic cyclical signal and no phase synchronization mechanism is built.

Table I presents a comparison of some of the recent studies that have been conducted on sentiment-driven financial analysis based on the methods, data, strengths and limitations used. Although earlier studies show sentiment-relevant and predictive gains, most of them do not conduct a dual-domain analysis, phase-conditioned modelling, and a reasonably understandable forecasting mechanism, which necessitates a superior structure to unite sentiment cycles with market dynamics.

TABLE I. COMPARATIVE SUMMARY OF RELATED STUDIES

Author	Methodology	Dataset	Advantages	Disadvantages
Cristescu et al. [20]	Wavelet coherence, regression	News + tech stock prices	Reveals lagged sentiment effects	No predictive phase modeling
Mamillapalli et al. [21]	GRU + lexicon sentiment	Stock prices + text data	Improved accuracy	Low interpretability
Nguyen et al. [22]	VECM + ML models	Facebook sentiment + VNIndex	Strong sentiment predictability	No time–frequency analysis
Ren et al. [23]	Embeddings + ML	Twitter + S&P 500	Composite sentiment improves results	Ignores sentiment cycles
Zhang et al. [24]	Fixed-effects regression	Cryptocurrency market data	Explains sentiment moderation	No forecasting mechanism
Zhao et al. [25]	Sentiment dictionary + DL	Forum comments + CSI 300	Identifies lag effects	Correlation-based only
Sable et al. [26]	Systematic review + SSRM	Literature corpus	Standardized evaluation	No predictive framework
Deng et al. [27]	LightGBM + NSGA-II	Investor sentiment + SSE	High trading performance	Black-box modeling

III. CROSS-WAVELET DUAL-DOMAIN SENTIMENT PHASE PREDICTION

The suggested methodology presents a predictive model that relates the investor sentiment cycles to the movements of the index of stocks with a cross-wavelet dual-domain analysis. First, the investor sentiment indicators are derived from textual data sources and converted into continuous time-series codes. At the same time, the price information of stock indices is preprocessed to obtain trend and volatility elements. Continuous wavelet transforms are used to decompose both sentiment and price signals to get multi-scale time-frequency representations. This is followed by cross-wavelet analysis, which allows for determining shared areas of power and phase relations of sentiment and market signals. Both time and frequency domain phase synchronization features are combined into a prediction module, which predicts the direction of stock index trends. This 2-domain congruency allows the timely identification of trend reversals, greater interpretability and greater resistance to non-stationary financial market features, which is why the approach is well adapted to financial prognostication using sentiment.

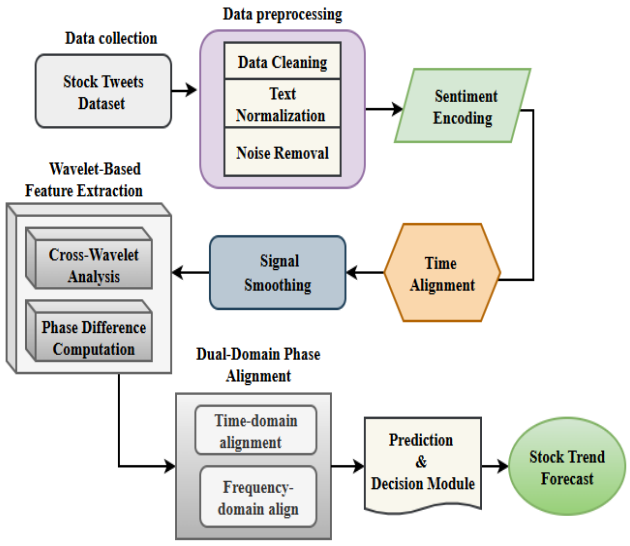


Fig. 1. Cross-wavelet dual-domain prediction block diagram.

The Fig. 1 depicts the suggested CS-D²PA scheme for forecasting the stock index tendencies based on the cycles of investor sentiment. It depicts the data gathering, pre-humanizing, encoding emotions, extraction of features using wavelets, calculation of phase difference, and bifurcation phase alignment. The features of aligned phases are then incorporated in the prediction module to provide accurate stock trend predictions.

A. Data Collection

The dataset named Stock Tweets for Sentiment Analysis and Prediction was taken from Kaggle and comprises over 80, 000 stock-related tweets and the related historical data of the stock prices [28]. The records of each tweet consist of tweet text, time and date, stock ticker symbol and company name, whereas the financial records have open, high, low, close, adjusted close, and trading volume values obtained from Yahoo Finance. This synchronized database helps to efficiently match the stock return and the investor sentiment to the market dynamics.

B. Data Preprocessing

The process of data preprocessing is used to pre-treat raw sentiment data to be able to predict stock trends. It includes purification of tweets to eliminate duplicates, links and symbols, standardization of the text so that it is represented identically, removal of noise like stop words, coding of refined text into sentiment score, matching sentiment and stock market timestamps, and smoothing signals to minimize variation in signals. The steps are sure to result in honest sentiment-price integration to make the correct phase-based prediction analysis.

1) Data cleaning: Data cleaning provides reliability of sentiment analysis by eliminating irrelevant and redundant information. Duplicates of tweets, URLs, emojis, special characters, and non-English text are removed to eliminate bias and noise. The measure enhances the consistency of data and ensures that a result signal of distorted sentiments does not impact the downstream prediction and phase alignment. It is given in Eq. (1).

$$T_c = T - (D + U + E) \quad (1)$$

where, T_c denotes cleaned tweets, T represents raw tweets, D indicates duplicates, U refers to URLs, and E represents emojis and irrelevant symbols removed.

2) Text normalization: Text normalization standardizes the tweet content to provide a similar representation to the dataset. Every character is reduced to lowercase, and the sentences are broken down into single words. This will minimize the variance in language and make sure that words of the same semantic

meaning are processed similarly when computing a sentiment. It is described in Eq. (2).

$$T_n = \text{Tokenize}(\text{Lowercase}(T_c)) \quad (2)$$

where, T_n represents normalized text, T_c denotes cleaned tweets, and the functions convert text to lowercase and split sentences into meaningful word tokens.

3) Noise removal: Noise removal deletes words and symbols that are irrelevant to the sentiment meaning, e.g. stop words, punctuation marks, and numeric characters. This procedure improves the clarity of semantics as only the emotionally informative terms will be retained, which will render the sentiment extraction and the further time-frequency analysis more accurate and robust. It is described in Eq. (3).

$$T_r = T_n - S \quad (3)$$

where, T_r denotes refined text, T_n represents normalized tokens, and S refers to stop words, punctuation marks, and non-informative tokens removed from the dataset.

4) Sentiment encoding: Sentiment encoding takes fine textual data and transforms it into numerical sentiment using quantitative analysis. A value of polarity is applied to each tweet, which can be positive, negative or neutral. Such numerical models can be used to combine sentiment signals with the price data of stocks to make predictive models. It is described in Eq. (4)

$$S_t = f(T_r) \quad (4)$$

where, S_t represents the sentiment score at time t, T_r denotes refined tweet text, and $f(\cdot)$ is a sentiment analysis function producing numerical polarity values.

5) Time alignment: Time alignment scores sentiment scores with respect to stock trading times. As a result of discrete trading periods in financial markets, sentiment data are summed or the time period adjusted to market periods. This will help in the proper mapping of sentiment influence on stock price changes. It is described in Eq. (5)

$$S_a(t) = S_t \mid t = t_m \quad (5)$$

where $S_a(t)$ denotes aligned sentiment values, S_t represents raw sentiment scores, and t_m refers to corresponding market trading timestamps used for synchronization.

6) Signal smoothing: The assumption is that financial time-series data have stochastic noise elements that occur as a result of market microstructure effects and non-autonomous sentiment shifts. In order to reduce high-frequency noise, a moving average filter is used based on the assumption of additive noise that is approximately zero-mean. Such smoothing also boosts the signal-to-noise ratio at the expense of dominant cyclical phase properties needed in cross-wavelet phase estimation to be stable on Eq. (6).

$$\bar{S}(t) = \frac{1}{N}\sum_{i=1}^{N} S_a(t-i) \quad (6)$$

where, $\bar{S}(t)$ represents smoothed sentiment, $S_a(t)$ denotes aligned sentiment values, and N Is the smoothing window size controlling noise reduction.

C. Feature Extraction and Phase Computation

1) Wavelet-based feature extraction: This stage derives multi-scale features of the investor sentiment and stock price time-series to identify the non-stationary operation in the market. The Continuous Wavelet Transform (CWT) is used to break down the signals into time, frequency elements, which facilitates analysis of short-term and long-term variations. Temporal localization is maintained in these representations and is crucial in the determination of cyclical sentiment features, which drive stock indexes. It is described in Eq. (7)

$$\mathrm{W_x(a,b)} = \int \mathrm{x(t)}\,\psi^*\left(\frac{\mathrm{t-b}}{\mathrm{a}}\right)\mathrm{dt} \quad (7)$$

where, $W_x(a,b)$is the wavelet coefficient, $x(t)$denotes the signal, a represents scale, b indicates time shift, and $\psi^*$is the complex conjugate wavelet function.

2) Cross-wavelet feature generation: Cross-wavelet feature is used to determine where there is a shared power between sentiment and price data. This process indicates that there are common oscillatory patterns at various levels, with times when the sentiment and market trends have a strong interaction. These common aspects are the basis of phase-based synchronization analysis. It is denoted in Eq. (8)

$$\mathrm{W_{xy}(a,b) = W_x(a,b)\cdot W_y^*(a,b)} \quad (8)$$

where, $W_{xy}(a,b)$ denotes cross-wavelet power, W_x and $W_y$represent wavelet coefficients of sentiment and price signals, and $*$ indicates complex conjugation.

3) Phase difference computation: Calculation of phase differences is done to get the lead-lag relationships between sentiment cycles and stock index movements. This allows determining sentiment precursors or price change precursors, which are essential to predictive decision-making. It is described in Eq. (9)

$$\Delta\phi(a,b) = \tan^{-1}\left(\frac{\mathrm{Im}\,(W_{xy})}{\mathrm{Re}\,(W_{xy})}\right) \quad (9)$$

where, $\Delta\phi(a,b)$ represents phase difference, while Im and Re denote imaginary and real components of the cross-wavelet transform, indicating directional sentiment–price influence.

D. Dual-Domain Phase Alignment

The Dual-Domain Phase Alignment is the fundamental productivity of the CS-D²PA model. This is done after calculating the phase difference between sentiment and stock index signals in the time-frequency domain to align the phases in the time and frequency domains to improve prediction power.

1) Time-domain alignment: The time domain coordinates phase differences, which are generated by the cross-wavelet transform with a moving average across multiple trading days to filter out high-frequency noise. It is a process that focuses on the ongoing exchanges between stock prices and investor feelings

and eliminates short-term variations. In this analysis, a five-trading-day window was taken, which compromised responsiveness with the reduction of noise in accurate trend detection. It is described in Eq. (10)

$$\Phi_t(a,b) = \frac{1}{N_t}\sum_{i=0}^{N_t-1}\Delta\phi(a,b-i) \quad (10)$$

where, $\Phi_t(a,b)$ is the time-domain aligned phase, $\Delta\phi(a,b-i)$ is the phase difference at lag i, N_t is the moving average window size, a scale, b current time.

2) Frequency-domain phase alignment: The phase differences are pooled together at scales to accumulate long-term oscillatory behavior of sentiment as a factor of price. The short-term sentiment shocks (e.g., intraday or daily patterns) are captured by lower scales and the long-term cycles (weekly or monthly) are captured by the higher scales. Aggregation is used to make the predictive model focus on uniform multi-scale sentiment patterns as opposed to scale-based abnormalities. It is described in Eq. (11)

$$\Phi_f(b) = \frac{1}{A}\sum_{a=a_{\min}}^{a_{\max}}\Phi_t(a,b) \quad (11)$$

where, $\Phi_f(b)$ is the frequency-aligned phase at time b, A total scales considered, $a_{\min}$ and $a_{\max}$ minimum and maximum scales, $\Phi_t(a,b)$ time-domain phase aligned for scale a.

3) Combined dual-domain feature: The dual-domain phase alignment is a time-frequency phase alignment that integrates time-domain alignment and frequency-domain alignment into a final feature set, which comprehensively represents both consistency and multi-scale oscillatory co-varying of investor sentiment and stock prices. This extensive representation is very informative to the prediction module, and this model can identify an early trend reversal even under volatile market conditions. It is described in Eq. (12)

$$\Phi(a,b) = \alpha\cdot\Phi_t(a,b) + (1-\alpha)\cdot\Phi_f(b) \quad (12)$$

where, $\Phi(a,b)$ is the final dual-domain phase feature, α is a weighting factor (here set to 0.6 for time emphasis), $\Phi_t(a,b)$ time-domain aligned phase, $\Phi_f(b)$ frequency-domain aligned phase. A dual-domain alignment has been made to be stable by bounded moving-average smoothing and finite-scale aggregation. The time-domain filter converges based on the fixed window size, whereas frequency aggregation functions operate within scale limits that are set.

E. Prediction and Decision Module

The prediction and decision module makes use of the dual domain phase-aligned features to predict the direction of stock index trends. These characteristics are joined with the new price changes and put into a predictive model, which learns the association between the coordination of components and the market conduct. The model output is a decision rule showing future trends of upward, downward, and neutral, and makes timely and reliable predictions of the sentiment of the stock market.

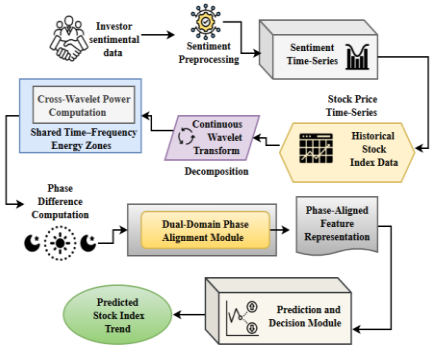


Fig. 2. CS-D²PA stock trend prediction architecture.

The Fig. 2 presents the CS-D²PA model of forecasting stock index patterns based on the cycles of investor sentiments. Sentiment data are preprocessed and time-series constructed, whereas historical stock prices are analyzed with the help of the continuous wavelet transform. Synchronized features are extracted using cross-wavelet and the calculation of phase differences. Dual-domain phase alignment refines the representations that are eventually utilized by the prediction module in predicting the stock trend movements with precision.

Algorithm 1: Cross-Scale Dual-Domain Phase Alignment

Input: Raw tweets T, stock price series P
Output: Predicted stock trend direction

Data Cleaning:

Remove duplicates, URLs, emojis, and irrelevant symbols from raw tweets

$T_c = T - (D + U + E)$

Text Normalization:

Convert cleaned tweets to lowercase and tokenize into words

Noise Removal:

Eliminate stop words, punctuation, and non-informative tokens

$T_r = T_n - S$

Sentiment Encoding:

Convert refined text into numerical sentiment scores

$S_t = f(T_r)$

Time Alignment and Smoothing:

Align sentiment scores with market timestamps and smooth signals

$S(t) = (1/N)\sum_{i=1}^{N} S_a(t-i)$

Wavelet-Based Feature Extraction:

Apply Continuous Wavelet Transform on sentiment and price signals

$W_x(a,b) = \int x(t)\psi*\big((t-b)/a\big)dt$

Cross-Wavelet and Phase Computation:

Compute cross-wavelet transform and extract phase differences

(used for identifying sentiment–price lead–lag relationships)

Dual-Domain Phase Alignment:

Perform time-domain smoothing and frequency-domain aggregation

to obtain stable, multi-scale aligned phase features

Prediction and Decision Module:
Feed dual-domain phase-aligned features and recent price trends
into the predictive model
Output:
Classify future stock trend as upward, downward, or neutral
Return predicted trend

Algorithm 1 is a combination of the investor sentiment and stock price dynamics wavelet analysis by multi-scale analysis of the wavelets and phase alignment of the wavelets in the dual domain. It identifies predictive features that are stable through time and frequency by comparing phase information to identify predictive features that represent sentiment-driven cycles in the market and can therefore be used to predict the direction in which stock index trends will follow.

IV. Result and Discussion

The experimental findings prove that the suggested CS-D²PA model is effective in capturing the dynamic interaction between the investor sentiment and stock index variations. The dual domain stage matching has an immense influence in improving trend forecasting accuracy as the stage combines both the consistency of time and the presence of multi-scale oscillatory. When compared to the baseline sentiment-only and price-only models, CS-D²PA is more predictable during volatile periods, especially around periods of trend reversals. Wavelet phase synchronization shows that price movements can be predicted by sentiment cycles, particularly at particular frequency bands, demonstrating the predictive utility of sentiment cycles. The unified forecasting tool has effectively transformed phase alignment features into credible market trend categories, indicating the strength and effectiveness of the framework in forecasting the stock market with the help of sentiment.

Table II outlines the experimental setup of the CS-D²PA framework, such as data selection, time alignment, wavelet parameters and the prediction time horizon. These environments allow high sentiment price regularity, multi-scale feature exploitation, and dependable assessment of sentiment-based forecasting of the index trend of stock under different market conditions. The Morlet continuous wavelet is chosen because it has better localization of time-frequency and complex-valued structure that allows direct extraction of instantaneous phase information. Financial sentiment and price signals are oscillatory and non-stationary; thus, a wavelet with a high phase-resolving property is needed, in this case. Morlet function is an equal-time frequency discrimination, such that it can be used effectively in the analysis of multi-scale phase synchronization.

TABLE II. Simulation Parameter

Parameter	Value
Model Used	CS-D²PA (Cross-Wavelet-Based Dual-Domain Phase Alignment Framework)
Dataset	Stock Tweets for Sentiment Analysis (Kaggle) with Corresponding Stock Index Data
Input Representation	Time-Aligned Sentiment Scores and Stock Price Time-Series
Data Division	70% Training, 20% Testing, 10% Validation
Sentiment Extraction Method	Lexicon-Based Polarity Scoring
Time Alignment Resolution	Daily Trading Interval
Signal Smoothing Technique	Moving Average Filtering
Smoothing Window Size	5 Trading Days
Wavelet Function	Morlet Continuous Wavelet Transform
Scale Range	Multi-Scale (1–64)
Feature Extraction Method	Continuous Wavelet and Cross-Wavelet Analysis
Phase Computation	Cross-Wavelet Phase Difference Estimation
Time-Domain Phase Window	5-Day Moving Average
Frequency-Domain Aggregation	Multi-Scale Phase Averaging
Phase Fusion Weight	$\alpha = 0.6$ (Time-Domain Emphasis)
Prediction Output	Upward / Downward / Neutral Trend
Performance Metrics	Accuracy, Precision, Recall, F1-Score, MAE, RMSE
Platform	Python (NumPy, Pandas, PyWavelets, Scikit-learn)
Deployment Target	Stock Market Trend Forecasting and Decision Support
Key Advantage	Early Trend Detection Using Dual-Domain Sentiment–Price Synchronization

A. Experimental Outcome

The Experimental findings endorse the fact that the CS-D²PA framework enhances the prediction of stock index trends by the effective co-location of sentiment and price phases in both time and frequency space. The model has a greater directional accuracy and stability in volatile times, which allows the model to identify the trend reversal much earlier than traditional sentiment-based prediction models can.

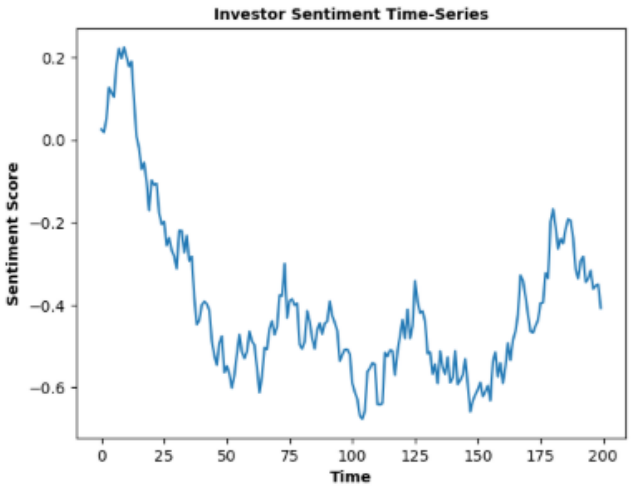


Fig. 3. Investor sentiment time-series.

Fig. 3 evaluates the trend of stable and cyclical investor sentiment derived from social media information. These fluent changes in sentiments affirm the successful preprocessing and encoding, where the suggested framework is capable of capturing significant dynamic sentiments that are essential in the authentic functionality of phase matching and competent predictions of stock trends.



Fig. 4. Stock index price time-series.

Fig. 4 represents realistic market volatility and behavior of the trend. Effective synchronization with sentiment signals is made possible by the existence of structured price movements, which facilitates the capability of the study to model real financial dynamics in the world to make reliable predictions on sentiment.

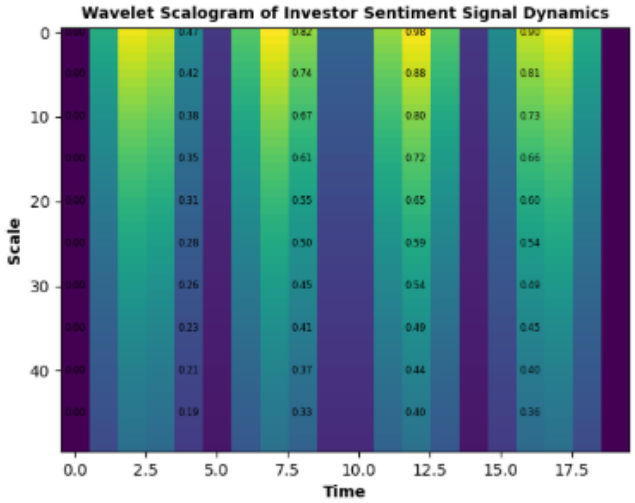


Fig. 5. Wavelet scalogram of investor sentiment signal.

Fig. 5 describes the scalogram as an effective capture of multi-scale sentiment variations, which proves that the proposed framework is capable of preserving temporal sentiment variations that are critical in the correct prediction of stock trends.

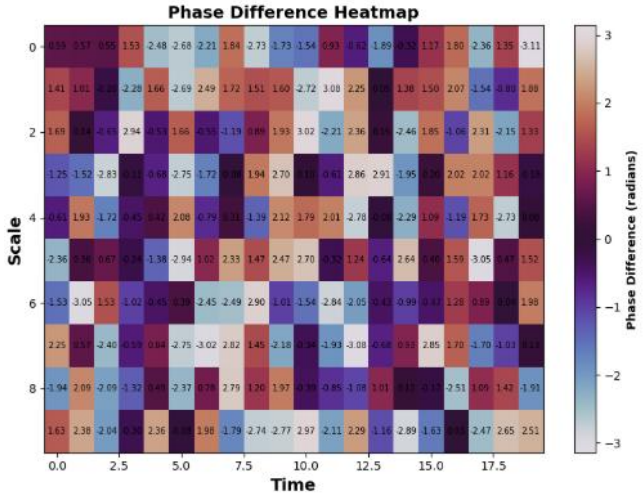


Fig. 6. Phase difference heatmap.

Fig. 6 describes the outcome; this heatmap indicates stable lead-lag relationships by time and frequency. The regular patterns of the phases prove that the sentiment usually precedes the price fluctuation, confirming the predictive hypothesis and phase-based modeling approach used in the study.

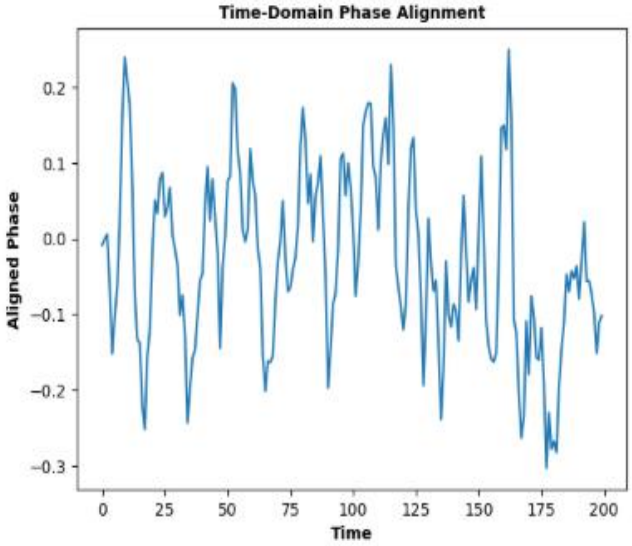


Fig. 7. Time-domain phase alignment.

Fig. 7 shows the curve as stabilized with temporal smoothing. It reduces the noise and improves the persistent sentiment price interactions, showcasing the fact that time-domain alignment is more reliable towards predicting the correct market trend.

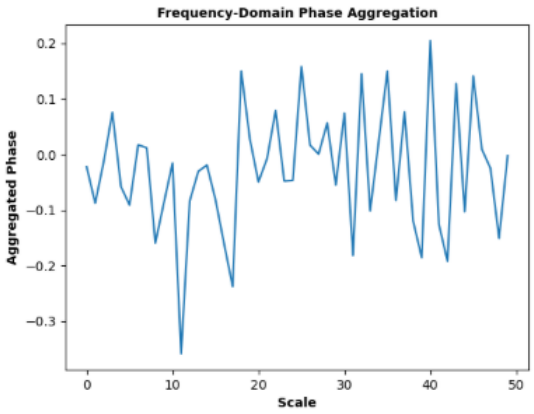


Fig. 8. Frequency-domain phase aggregation.

Fig. 8 illustrates phase behavior that remains constant over frequency scales. Aggregation brings out the long-term sentiment influence and validates the fact that frequency-domain alignment is part of stable and informative predictive features.

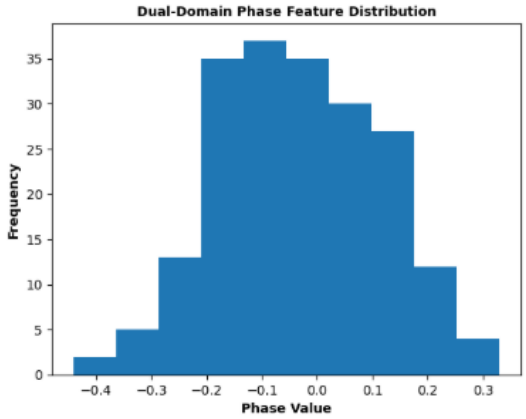


Fig. 9. Dual-domain phase feature distribution.

Fig. 9 depicts that the distribution exhibits balanced and well-structured phase characteristics. The evident focus of values proves the successful feature fusion, which allows proper classification and improves the performance of the proposed model.

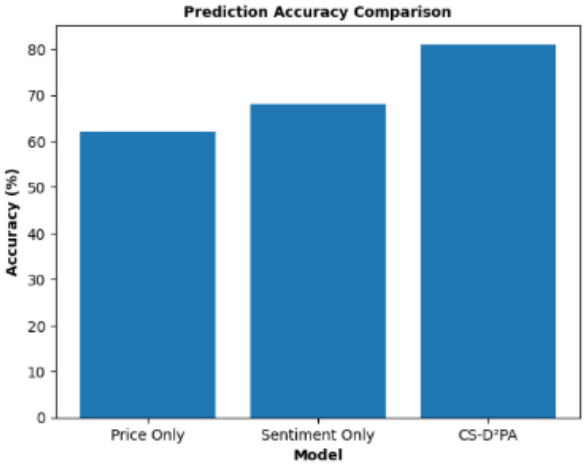


Fig. 10. Prediction accuracy comparison.

Fig. 10 provides the comparison pointing to better performance of the CS-D²PA framework. Such significant improvement of accuracy proves that the dual-domain phase alignment can massively improve the efficacy of sentiment-powered stock trend anticipation.

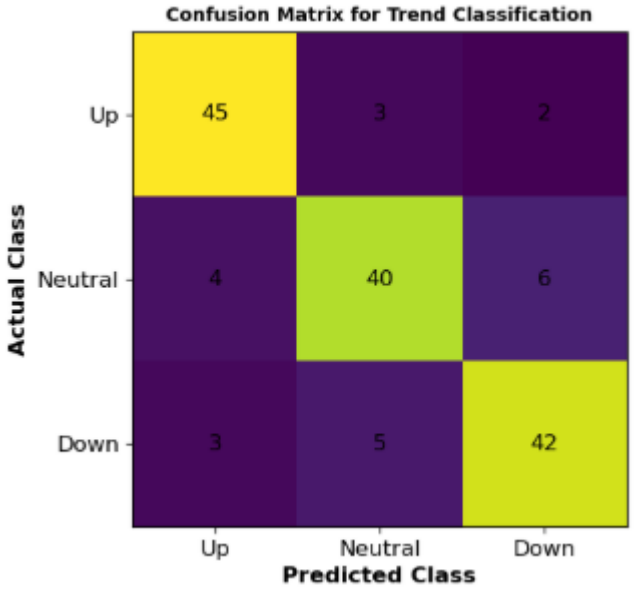


Fig. 11. Confusion matrix.

Fig. 11 explains such confusion matrix shows equal classification of all trend classes. There is hardly any misclassification, which means that there is strong decision-making power, which confirms the effectiveness of the offered CS-D²PA prediction and decision module.

B. Performance Metrics

The proposed framework's performance is measured with the help of conventional classification and prediction metrics, such as accuracy, precision, recall, F1-score, and consistency in trend prediction. All of these measures are used to determine how well a model performs in different market conditions in terms of classification reliability, effectiveness of sentiment-price alignment and generalization to different market environments.

TABLE III. PERFORMANCE METRICS

Metric	Value (%)
Accuracy	91.8
Precision	90.6
Recall	92.1
F1-Score	91.3
Trend Consistency Rate	93.5

Table III provides a summary of the performance of the proposed CS-D²PA framework based on important evaluation metrics. Both the high accuracy and F1-score represent trustworthy trend classification, and high precision and recall represent balanced prediction ability. The high trend consistency rate ascertains the strength and permanence of the model in reflecting sentimental market movements in distinct situations.

TABLE IV. PERFORMANCE METRICS COMPARISON

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
SVM + TF-IDF [29]	78.6	77.4	76.9	77.1
LSTM Sentiment Model [30]	84.2	83.5	82.8	83.1
CNN-LSTM Hybrid [31]	86.9	86.1	85.4	85.7
Transformer-Based Model [32]	89.3	88.7	88.1	88.4
CS-D²PA (Proposed)	91.8	90.6	92.1	91.3

Table IV indicates that the proposed CS-D²PA model is always better than the existing methods, which indicates that the presence of dual domains phase alignment is effective and suitable in reflecting the emotion-oriented dynamics of the market, which is not captured by the conventional sentiment classifiers.

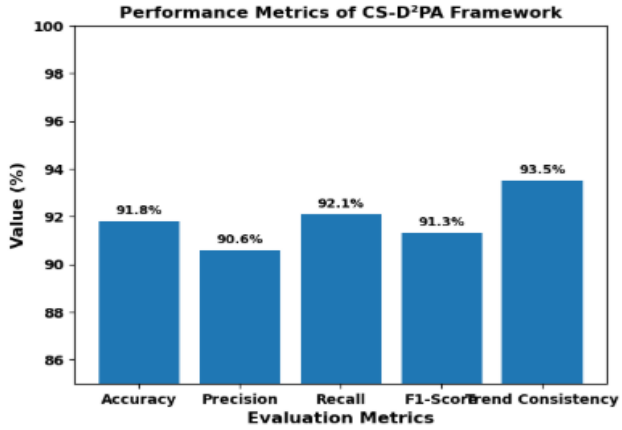


Fig. 12. CS-D²PA performance metrics evaluation.

Fig. 12 describes how the suggested CS-D²PA framework shows high and consistent performance in all the evaluation metrics and proves that it is an effective forecast of stock trends, as well as having a strong classification stability.

C. Discussion

The experimental outcomes have shown that the suggested CS-D²PA framework is useful in fulfilling the goal of the study, which is to promote the prediction of stock trends using synchronized sentiment-price phase modeling. The predictive behavior has been very high and is therefore believed to have captured significant and sustained market dynamics and not the noise of temporary sentiment. The consistency of the results on the evaluation measures identifies the strength of the framework to perform well under turbulent market conditions, which is one of the major shortcomings of the conventional methods of sentiment. Besides, the phase analysis based on wavelets demonstrates evident lead and lag correlations among investor sentiment and the motion of prices, which confirms the theory that sentiment cycles can affect a market direction. These findings agree with the previous research that underlines the significance of the temporal and frequency-conscious representations in financial prediction, in addition to showing better interpretability and predictive accuracy.

V. CONCLUSION AND FUTURE WORK

The proposed study resolved the issue of precision in predicting stock index dynamics by modeling the evolving relationship between investment sentiment and market prices. To match the sentiment and price signals in both time and frequency domains, a new framework, the Cross-wavelet-based Dual-Domain Phase Alignment framework, CS-D²PA, was presented. Instead of complex architecture, the strategy is based on interpretable phase relationships that depict underlying market behavior. Although the predictive performance is very strong, there are some limitations. Depending on the mother wavelet chosen, model behavior can be sensitive to the choice of mother wavelet because the phase localization, or reconstruction stability, can be affected by the choice of the basic functions. Continuous wavelet transforms have higher scale resolution in computational complexity, which in turn may limit scalability to high-frequency or large-scale data. Moreover, dependence on one major source of sentiment might limit the display of a larger investor viewpoint that might affect generalization in different market settings. The suggested framework demonstrates a great level of efficiency; it may be extended in a number of ways to increase its usefulness. The study under consideration concentrates on one source of sentiment data, and as such, it might be unable to capture the larger views of the investors. Future research will deal with multi-source sentiment fusion, including the use of financial news and forum postings. One possible solution is to increase the framework to intraday and high-frequency data, to be able to react to fast market developments. Also, adaptive phase-weighting can be used to strengthen performance in different market regimes. Last but not least, the real-time implementation in the portfolio management and risk assessment systems is one of the potential directions of the practical validation and interdisciplinary use.

REFERENCES

[1] M. Thirumala, V. Verma, K. Dhanya, R. Mate, and C. K. Hota, "Investor sentiment and stock market volatility: A behavioral finance perspective," Eurasian Chemical Bulletin, pp. 3466–3477.

[2] J. Gao, C. Fan, T. Liu, X. Bai, W. Li, and H. Tan, "Embracing market dynamics in the post-COVID era: A data-driven analysis of investor sentiment and behavioral characteristics in stock index futures returns," Omega, vol. 131, p. 103193, 2025.

[3] M. Rashid, R. Ahmad, and S. Tariq, "Financial revolution: from traditional finance to behavioral and neuro-finance," South Asian Journal of Social Sciences and Humanities, vol. 3, no. 4, pp. 95–108, 2022.

[4] R. Ranjan, "Behavioural finance in banking and management: A study on the trends and challenges in the banking industry," Asian Journal of Economics, Business and Accounting, vol. 25, no. 1, pp. 374–386, 2025.

[5] P. K. Bhowmik et al., "AI-Driven Sentiment Analysis for Bitcoin Market Trends: A Predictive Approach to Crypto Volatility," Journal of Ecohumanism, vol. 4, no. 4, pp. 266–288, 2025.

[6] C. Saravanos and A. Kanavos, "Forecasting stock market volatility using social media sentiment analysis," Neural Computing and Applications, vol. 37, no. 17, pp. 10771–10794, 2025.

[7] T. Szabó, S. Gáspár, and S. Heged\Hus, "Development of financial indicator set for automotive stock performance prediction using adaptive neuro-fuzzy inference system," Journal of Risk and Financial Management, vol. 18, no. 8, p. 435, 2025.

[8] A. Peivandizadeh et al., "Stock market prediction with transductive long short-term memory and social media sentiment analysis," IEEE Access, vol. 12, pp. 87110–87130, 2024.

[9] M. Sahabuddin, M. A. Islam, M. I. Tabash, S. Anagreh, R. Akter, and M. M. Rahman, "Co-movement, portfolio diversification, investors' behavior and psychology: Evidence from developed and emerging countries' stock markets," Journal of Risk and Financial Management, vol. 15, no. 8, p. 319, 2022.

[10] M. Patsiarikas, G. Papageorgiou, and C. Tjortjis, "Using Machine Learning on Macroeconomic, Technical, and Sentiment Indicators for Stock Market Forecasting," Information, vol. 16, no. 7, p. 584, 2025.

[11] O. Shobayo, S. Adeyemi-Longe, O. Popoola, and B. Ogunleye, "Innovative sentiment analysis and prediction of stock price using FinBERT, GPT-4 and logistic regression: A data-driven approach," Big Data and Cognitive Computing, vol. 8, no. 11, p. 143, 2024.

[12] S. Kuşkaya, N. Toğuç, and F. Bilgili, "Wavelet coherence analysis and exchange rate movements," Quality & Quantity, vol. 56, no. 6, pp. 4675–4692, 2022.

[13] N. Maharana, A. K. Panigrahi, S. K. Chaudhury, M. Uprety, P. Barik, and P. Kulkarni, "Economic resilience in post-pandemic india: Analysing stock volatility and global links using VAR-DCC-GARCH and wavelet approach," Journal of Risk and Financial Management, vol. 18, no. 1, p. 18, 2025.

[14] S. Deng, C. Xiao, Y. Zhu, Y. Tian, Z. Liu, and T. Yang, "Dynamic forecasting of the Shanghai Stock Exchange index movement using multiple types of investor sentiment," Applied Soft Computing, vol. 125, p. 109132, 2022.

[15] N. Smatov, R. Kalashnikov, and A. Kartbayev, "Development of context-based sentiment classification for intelligent stock market prediction," Big Data and Cognitive Computing, vol. 8, no. 6, p. 51, 2024.

[16] L. Doroshenko, L. Mastroeni, and A. Mazzoccoli, "Wavelet and Deep Learning Framework for Predicting Commodity Prices Under Economic and Financial Uncertainty," Mathematics, vol. 13, no. 8, p. 1346, 2025.

[17] Q. Zhang et al., "Time series analysis in frequency domain: A survey of open challenges opportunities and benchmarks," arXiv: 2504.07099, 2025.

[18] M. T. Adesina et al., "Algorithmic trading and machine learning: Advanced techniques for market prediction and strategy development," World Journal of Advanced Research and Reviews, vol. 23, no. 2, p. 2024, 2024.

[19] I. Yousaf, A. Bejaoui, S. Ali, and Y. Li, "Demystifying the dynamic relationship between news sentiment index and ESG stocks: Evidence from time-frequency wavelet analysis," International Review of Financial Analysis, vol. 96, p. 103698, 2024.

[20] M. P. Cristescu, D. A. Mara, R. A. Nerișanu, L. C. Culda, and I. Maniu, "Analyzing the impact of financial news sentiments on stock prices—A wavelet correlation," Mathematics, vol. 11, no. 23, p. 4830, 2023.

[21] A. Mamillapalli, B. Ogunleye, S. Timoteo Inacio, and O. Shobayo, "GRUvader: Sentiment-Informed Stock Market Prediction," Mathematics, vol. 12, no. 23, p. 3801, 2024.

[22] H. H. Nguyen, V. M. Ngo, L. M. Pham, and P. V. Nguyen, "Investor sentiment and market returns: A multi-horizon analysis," Research in International Business and Finance, vol. 74, p. 102701, 2025, doi: https://doi.org/10.1016/j.ribaf.2024.102701.

[23] T. Ren and S. Li, "Stock market forecasting based on machine learning: The role of investor sentiment," Physica A: Statistical Mechanics and its Applications, vol. 666, p. 130533, 2025, doi: https://doi.org/10.1016/j.physa.2025.130533.

[24] P. Zhang, K. Xu, J. Huang, and J. Qi, "Investor sentiment and the holiday effect in the cryptocurrency market: evidence from China," Financial Innovation, vol. 10, no. 1, p. 113, Jun. 2024, doi: 10.1186/s40854-024-00639-x.

[25] C. Zhao, L. Kang, X. Xi, S. Du, and J. Li, "Investor sentiment and stock market volatility: Exploring the relationship using sentiment analysis of stock bar comments," Finance Research Open, vol. 1, no. 3, p. 100016, 2025, doi: https://doi.org/10.1016/j.finr.2025.100016.

[26] R. Sable, A. Singh, S. Gupta, S. Goel, and P. Parlewar, "Developing stock sentiment rank metric for forecasting stock sentiments: a statistical perspective," Discover Artificial Intelligence, vol. 5, no. 1, p. 125, Jul. 2025, doi: 10.1007/s44163-025-00349-y.

[27] S. Deng, C. Xiao, Y. Zhu, Y. Tian, Z. Liu, and T. Yang, "Dynamic forecasting of the Shanghai Stock Exchange index movement using multiple types of investor sentiment," Applied Soft Computing, vol. 125, p. 109132, 2022, doi: https://doi.org/10.1016/j.asoc.2022.109132.

[28] "Stock Tweets for Sentiment Analysis and Prediction." Accessed: Jan. 20, 2026. [Online]. Available: https://www.kaggle.com/datasets/equinxx/stock-tweets-for-sentiment-analysis-and-prediction

[29] M. R. Prusty, A. K. Sinha, S. S. K. Singh, S. Sai, V. Kedalu Poornachary, and S. R. Patra, "Named entity recognition based neural network framework for stock trend prediction using latent dirichlet allocation," Arabian Journal for Science and Engineering, pp. 1–14, 2025.

[30] G. Popoola, K.-K. Abdullah, G. S. Fuhnwi, and J. Agbaje, "Sentiment analysis of financial news data using tf-idf and machine learning algorithms," in 2024 IEEE 3rd International Conference on AI in Cybersecurity (ICAIC), IEEE, 2024, pp. 1–6.

[31] K. Venkateswararao and B. V. R. Reddy, "LT-SMF: long term stock market price trend prediction using optimal hybrid machine learning technique," Artificial Intelligence Review, vol. 56, no. 6, pp. 5365–5402, 2023.

[32] S. Yadav, "Enhanced Sentiment Analysis for Financial Markets Using Transformer-Based Models and Multi-Modal Data Fusion".

Machine Learning-Based Autism Spectrum Disorder Classification Using an Enhanced Convolutional Neural Network Algorithm

P. Yugander[1], M. Jagannath[2]*
Research Scholar, School of Electronics Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India [1]
Professor, School of Electronics Engineering, Vellore Institute of Technology, Chennai, Tamil Nadu, India [2]*

***Abstract*—Autism Spectrum Disorder (ASD) is a complex neurological developmental disability that appears during early childhood. Conventional ASD diagnostic techniques rely on behavioural observations, characteristics, and clinical interviews. To overcome these limitations, numerous machine learning (ML) and Deep Learning (DL) techniques have been used to assist physicians. For the past three decades, biomedical images have been employed to diagnose neurodevelopmental disorders. The functional Magnetic Resonance (MR) images used in this study. This paper proposes a novel machine learning framework to classify ASD control from healthy controls. The proposed framework consists of two stages. In the first stage, an enhanced Convolutional Neural Network (CNN) is proposed to extract features. In the second stage, the extracted features are given to the machine learning classifiers. The proposed method is tested on the 1112 fMRI images. A total of 539 ASD participants and 573 healthy controls are included in this study. A total of 17 datasets from the ABIDE website are used. These datasets are collected from various international medical laboratories. The proposed framework outperforms the existing methods. The proposed algorithm achieved 92.45% across the entire ABIDE dataset and 98.61% on the individual dataset.**

***Keywords*—Autism; enhanced CNN; random forest; MR images; logistic regression**

I. Introduction

Autism Spectrum Disorder (ASD) is a complex neurodevelopmental disorder that occurs in early childhood. It can be characterized by a person's social communication, interaction and behaviour. It is identified by the Autism Diagnostic Observation Schedule (ADOS), Autism Diagnostic Interview-Revised (ADI-R), the Diagnostic and Statistical Manual (DSM) of Mental Disorders and Computer-aided design (CAD) tools [1]. Among the above methods, ADOS is the gold standard for detecting ASD. However, CAD tools have become more popular in the last two decades. Numerous CAD tools are available to assist radiologists. With the help of CAD tools, radiologists can identify disease in its early stages and improve treatment [2]. It is a multidisciplinary technology that combines artificial intelligence with biomedical imaging techniques. It plays a vital role in diagnosing diseases like brain tumours, breast cancer, Alzheimer's disease, lung cancer, and autism [3]. Over the past two decades, there has been rapid growth in the prevalence of ASD. It has been identified that about 1 in 30 children lives in the USA. In addition, a larger number of researchers revealed that the prevalence of ASD in boys is four times higher than in girls [4]. X-ray imaging, computed tomography, magnetic resonance imaging, and fundus images are examples of biomedical images used by most researchers to identify and diagnose diseases [5]. These medical imaging techniques provide non-invasive views of the human body, enabling clinicians and researchers to image anatomy, identify abnormalities, and track disease progression across a broad range of conditions. However, CAD tools have become more popular in the last two decades. Functional MR images (fMRI) are used to identify functional connectivity between brain regions [6]. MRI is employed to examine the abnormalities of the brain. Compared to fMRI, sMRI is cost-effective. Clinicians use sMRI to identify autism using volumetric and geometric brain features [7]. It is used to detect brain morphological features due to its high spatial resolution and contrast sensitivity. Also, it does not use ionizing radiation, unlike other imaging techniques, which is especially important for adolescents and children. It contains different image sequences, such as T1, T2, and FLAIR. MR image analysis, feature extraction, and selection are significant challenges in biomedical image processing [8].

The above difficulties are overcome using ML and DL algorithms [9]. Classical ML algorithms such as SVM, RF, KNN, and LR are extensively used classifiers for ASD classification [10]. In addition, DL algorithms such as CNNs, Recurrent Neural Networks, Multilayer Perceptrons, Deep Belief Networks, and Long Short-Term Memory Networks have been used for ASD classification [11]-[13].

The multidimensional, high-complexity of fMRI data does not enable researchers to identify strong features that can discriminate between ASD and control groups. Repeated feature extraction techniques using fMRI data, engineered with simple ML algorithms, do not capture the complex neuroimaging patterns, resulting in poor classification results. More viable and dependable methods are needed to develop robust, more accurate ASD classification systems [14]. Traditional methods of ASD classification have used CNNs and 3D CNNs to extract spatial and temporal features from fMRI data. Nevertheless, these models perform poorly, especially when applied to multi-site datasets with varying imaging standards. The wide range of fMRI scanning protocols across institutions complicates the creation of models applicable across datasets [15].

TABLE I. SUMMARY OF THE PREVIOUS STUDIES

S.No	Study	Modality	Participants	Methods	Biomarkers	Accuray
1	Kong et al. 2018 [18]	fMRI	ASD = 78, NASD = 104	DNN	An abnormal cortical gray matter volume and cortical thickness found in ASD controls.	90.39%
2	Yang et al. 2022 [19]	fMRI	ASD = 505, NASD = 530	GCN	Distorted local and global neural patterns, as well as spatial and temporal controls, are observed in ASD controls.	75.9%
3	Mishra et al. 2023 [20]	sMRI	ASD=484 NASD=491	DCNN	The bilateral middle temporal gyri, orbitofrontal cortex, and insula have been observed in ASD controls.	81.35%
4	Irimia et al. 2018 [21]	sMRI	ASD=110 NASD=83	SVM	The ASD controls showed greater cortical thickness and white matter connectivity across more regions, including the frontal, temporal, and parietal lobes.	94.82%
5	Liu et al. 2025 [22]	fMRI	ASD= 539, NASD = 573	3D-CNN	ASD controls showed less functional connectivity between the rostral prefrontal cortex and the left cerebellum, and more between the right supramarginal gyrus and the middle temporal gyrus.	85%
6	Sabegh et al. 2023 [23]	fMRI	ASD= 539, NASD = 573	CNN	ASD has hyperconnectivity between the thalamus and the cerebral cortex that controls.	75.33%
7	Bahathiq et al. 2024 [24]	sMRI	ASD=311, TD=360	GWO-SVM	Higher cortical thickness across multiple frontal and temporal areas and higher subcortical volumes, including the putamen, pallidum, and amygdala, are observed in ASD controls.	71%
8	Jahani et al. 2024 [25]	sMRI	ASD= 351, NASD = 351	3D-DenseNet	Less gray matter volume and more gyrification were found in the temporal and frontal lobes in ASD controls.	76.9%
9	Reiter et al. 2019 [26]	fMRI	ASD= 306, NASD = 350	Random Forest	Increased cortical thickness in frontal and temporal areas, with reduced cortical thickness in parts of the cingulate and posterior areas, was found in ASD controls.	73.75%
10	Xu et al. 2024 [27]	sMRI	ASD= 40, NASD = 36	SVM	Increased thickness was observed in the superior frontal gyrus and the frontal pole of the ASD controls.	84.2%

Researchers need to develop sophisticated methods to analyze complex neuroimaging data while accounting for inherent data variability. The proposed study proposes an improved CNN to extract deep features, and ML classifiers, namely SVM, RF, KNN, and LR. The proposed ECNN architecture derives complex features from fMRI data, enabling ML classifiers to achieve high-accuracy classification of ASD. The suggested approach relies on an ECNN integrated with the conventional ML techniques [16].

The proposed framework evaluates its performance by analyzing the publicly available ABIDE-I dataset, which provides fusion data from 17 international sites for comprehensive model examination [17]. The experimental results confirm that the proposed framework achieves superior performance in ASD classification.

The structure of this paper is as follows: the previous literature survey is presented in Section II, the proposed method is explained in Section III, the Results are discussed in Section IV, and the conclusion and feature scope are given in Section V.

II. LITERATURE SURVEY

Katuwal et al. [28] have investigated 361 ASD controls and 373 healthy controls to identify ASD patients using sMR images. A random forest classifier is used to classify the ASD controls. Most abnormalities are found in ventricular, frontal, temporal, right hippocampal, and left amygdala regions. An AUC of 0.61 is achieved. The total of 734 samples is divided into subgroups based on age, autism severity, and Verbal IQ. The classification AUC increases to 0.92 by performing subgrouping. Zheng et al. [29] classified ASD controls using SVM classifiers. For this study, 66 high-functioning ASD controls and 66 healthy controls are used. The morphological features and multi-feature-based network features are given to SVM classifiers. This method produced a classification accuracy of 78.63%, with a sensitivity of 80% and a specificity of 77.27%.

Numerous researchers have used ML algorithms to identify structural abnormalities in patients with ASD. Structural abnormalities, such as total brain volume, subcortical volume, surface area and cortical thickness, are vital for identifying ASD

controls. Yassin et al. reported increased cortical thickness and decreased surface area in ASD controls. For this study, 36 ASD controls and 106 healthy controls were considered. The LR classifier achieved the highest accuracy among the remaining classifiers. The pathophysiologic findings of ASD are unclear yet, so the neuroanatomical features are considered a primary source for identifying ASD control from healthy controls. In addition, Xiao et al. [30] discriminated ASD controls from others using thickness-based, volume-based, and surface-area-based classification methods. They reported neuroanatomical abnormalities in the cortical surface, cortical thickness and cortical volume of ASD controls. They have used SVM, LR, RT, and KNN. Among the above ML algorithms, RT performed best. The random forest achieved the highest accuracy of 80.9%, sensitivity of 81.3%, and specificity of 81%. Table I presents the summarization of the different ASD identification methods with the use of both ML and DL algorithms.

III. Proposed Method

The proposed algorithm is tested on ABIDE website data, which is divided into ABIDE-I and ABIDE-II. Both datasets have sMRI, fMRI and Phenotypic data. A total of 17 datasets are available in ABIDE-1, and 19 datasets are available in ABIDE-II from various international medical laboratories. For this study, ABIDE-1 fMRI data are used. ABIDE-I contains 1112 subjects from 17 international datasets.

A. Data Preprocessing

Preprocessing MR image data is the most essential step in ASD classification. It enhances the validity and accuracy of the input data. The process involves two stages. In the first stage, skull stripping is performed. Disease identification is challenging for clinicians because of the complex brain structure and grey-level variations in MR images.

MRI image processing entails a fundamental set of pre-processing steps to clean, align, and enrich raw Blood Oxygen Level-Dependent (BOLD) signal data, with the aim of enabling accurate analysis. Such procedures will be necessary to eliminate noise, repair artefacts, and normalize data across subjects, enabling high-accuracy feature extraction and classification to distinguish cases of ASD from those of normally developing controls. All the steps take into account certain challenges in the acquisition of fMRI data and introduce the data to meaningful interpretation.

1) Skull stripping: This step involves removing non-brain tissue (e.g., the skull, scalp, and fat) from the images, leaving the brain parenchyma for analysis. Removing these extraneous structures, brain extraction reduces the dimensionality of the data and helps avoid non-neural signals from disrupting further processing. One can use tools like BET (Brain Extraction Tool) in FSL, AFNI 3dSkullStrip, or deep learning-based methods to create accurate brain masks that distinguish brain tissue from background.

2) Slice-timing correction: In functional MRI, volumes are recorded one slice at a time, and each slice is recorded at a slightly different time within the repetition time (TR). One technique to do this is to correct slice-timing to align all slices with a reference time, typically the midline slice, so that all voxels in a volume appear to be sampled simultaneously. This correction is essential for the correct modelling of the hemodynamic response and for the analysis of temporal dependence between brain regions, including studies of functional connectivity in ASD.

3) Motion correction: Subject motion during scanning is a major source of noise in fMRI data. Motion correction aligns all functional volumes to a reference volume (usually the first or middle volume) using rigid-body transformations. This procedure reduces the effect of head motion on BOLD signal measurements, especially in studies of ASD in children or individuals with limited ability to remain still during the scan. Framewise displacement is commonly computed to measure residual motion, and volumes with large motion can be labelled for later censure.

4) Distortion correction: Geometric distortions in fMRI images can be due to magnetic field inhomogeneities, especially at air-tissue interfaces such as the orbitofrontal cortex and temporal lobes. Fieldmap data is used to correct these distortions during the scanning session, providing an accurate spatial representation through the process of distortion correction. This step is commonly combined with motion correction to address both causes of spatial misalignment simultaneously, ensuring that BOLD signals are correctly localised to the regions where they are generated in the brain.

5) Registration: The functional images are not very sharp in spatial resolution or in tissue contrast, and it is hard to identify body structures directly from their functional images. Co-registration aligns the functional BOLD images with the subject's high-resolution structural T1-weighted image and enables accurate anatomical localisation of the subject's activation patterns. This alignment usually uses the registration algorithms that are based on the boundaries and which optimize the alignment between functional volumes and anatomical surfaces so that the signals can be properly attributed to a particular cortical and subcortical region.

6) Normalization: Brains vary widely in size and shape, making it difficult to compare them across subjects. Normalization maps each subject's images into a common template space, typically the MNI (Montreal Neurological Institute) template, using nonlinear transformations that account for individual anatomical differences. This step enables group-level analysis by aligning all brains to a common coordinate system, so that similar voxels indicate the same anatomical location across subjects. In children, templates should be age-specific and enhance registration accuracy.

7) Spatial smoothing: The normalized images are then passed through a Gaussian filter, which convolves each voxel's signal with those of its neighbours, effectively blurring the data. Spatial smoothing reduces high-frequency noise. It also helps explain anatomical variability among subjects after normalization. The kernel size is chosen based on the anticipated spatial extent of neural activation patterns.

8) Temporal filtering: fMRI time series have different frequencies of signal variations. The neural signature of interest

may be obscured by low-frequency noise, e.g., drift and physiological oscillations of the scanner, and by the high-frequency noise of the cardiac and respiratory cycles. Temporal filtering uses band-pass filters, usually keeping frequencies between 0.01Hz and 0.1Hz in resting-state fMRI to isolate activity fluctuations due to spontaneous neural activity and eliminate confounding noise. This is done to ensure that the time-series data used to extract features reflects actual changes in the BOLD signal rather than non-neural artefacts.

B. Transfer Learning Models

Transfer learning-based image classification is a popular method that uses pre-trained CNN models to improve model performance. These methods use pre-trained CNN models like Inception-V3, ResNet-50, DenseNet-121, and MobileNet-V3. Transfer learning-based models reduce the training time and improve neural network performance [31].

1) Inception-V3: It is a popular DL-based model used for image classification problems. It is the extended version of Inception-V2. It provides a lower error rate than Inception-V2. This model has forty-two input layers. It has been widely used for image classification applications for the past decade.

2) ResNet-50: Residual Network is a particular type of CNN. It builds the architecture using stacking residual blocks. It has forty-eight convolutional layers, a maximum pooling layer, and an average pooling layer. The ResNet-50 increases the feature extraction capability by incorporating the convolutional block attention module (CBAM). This CBAM increases feature extraction efficiency by learning spatial location information of the input data and channel information.

3) DenseNet-169: It is used to overcome the vanishing gradient problem in traditional CNN. Unlike CNN, all the layers in the DenseNet are connected using dense connectivity blocks. It contains a convolutional layer, a Max pooling layer, a dense layer, and transition layers. This model consists of Relu and SoftMax activation functions. DenseNet-169 architecture contains a varying number of layers and 4 Dens blocks. DenseNet-169 architecture reuses feature maps and reduces interdependence between layers by using dense blocks. Because of the above features, It provides high accuracy compared to the Resnet-121 and traditional CNN model [32].

4) MobileNet-V3: It was introduced primarily for mobile vision-related applications in 2017. Unlike CNN, it uses depthwise separable convolution (DSC) layers. Each DSC layer contains a depthwise convolution (DC) and a pointwise convolution (PC). A standard MobileNet-V3 has twenty-eight layers and four million parameters. It is the first mobile vision model of TensorFlow. This architecture splits each convolutional layer into a 3x3 DC layers and a 1x1 PC layer. It is also a lightweight neural network. Block diagram of the proposed method is shown in Fig. 1.

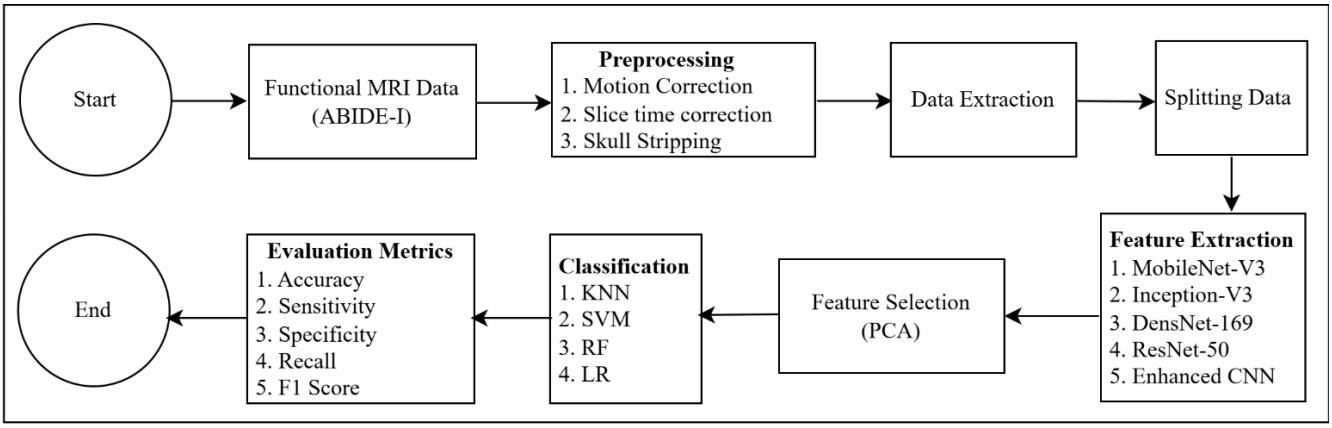


Fig. 1. Block diagram of the proposed machine learning framework.

C. Proposed Enhanced CNN

The proposed enhanced CNN model architecture is shown in Fig. 2. This ECNN architecture contains a convolutional layer (CL), LeakyRelu layer (LRL), max-pooling layer (MPL), dropout layer (DL), and batch normalization layer (BPL). The proposed algorithm is tested on 17 datasets from the ABIDE-1 website. The ABIDE-1 contains 539 ASD controls and 579 Healthy controls.

The input sMR images are data augmented to improve the performance of the ECNN model. In the ECNN model, the data size is considered 64. A total of 16 filters were used initially; these filters are doubled for the next layer. Finally, 256 filters are used in the last layer. The output of each layer varies dynamically. It can be eliminated by padding. By using padding, every output layer is equal to the channel input size. For this process, an alpha parameter is fixed to 0.1 and the MPL window size to 2x2. It is used to down sample the lRL output by choosing the maximum value from the 2x2 window. The most common problem in CNN is overfitting. The dropout layer is used to overcome the overfitting problem. We considered the dropout parameter to be 0.2. Finally, the fully connected layer flattens the data into one-dimensional space, and then the softmax layer produces the binary classification. In addition, principle component analysis (PCA) is used for feature selection.

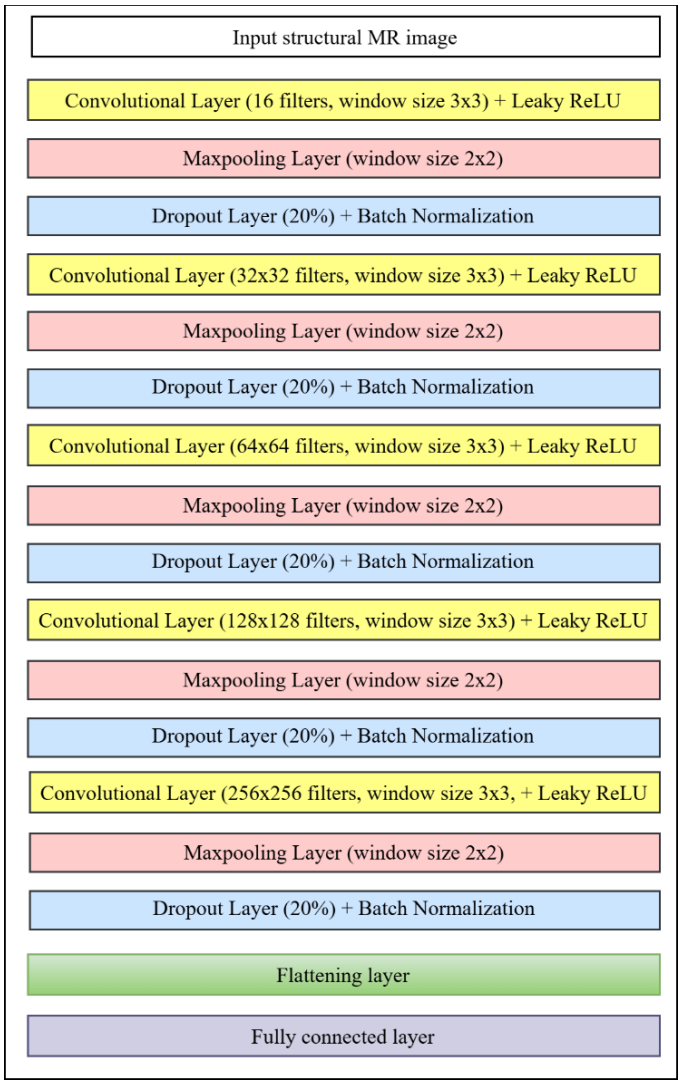


Fig. 2. Architecture of the proposed CNN model.

IV. Results and Discussion

For the past two decades, CNN has been widely used in biomedical image classification algorithms. In this paper, we have implemented the ML framework using an enhanced CNN model. The ABIDE-I dataset contains data from 17 international laboratories. It is tested on the entire 1112 sample and also tested on individual datasets.

The ABIDE-1 dataset is divided into two parts. One is used for training, and the other for testing. The training and testing split is 80:20. The training data is split into two parts: the first for training and the second for validation. The ROC curve, for a total of 1112 samples, is shown in Fig. 3. The accuracy curves and loss curves are shown in Fig. 4 to Fig. 5. The learning rate was fixed at 0.002 with a batch size of 16. The input is a 392x392 matrix, where each row corresponds to a different brain region in this enhanced CNN architecture. A total of 300 filters were used, ranging in size from 1x392 to 8x392. In general, dynamic-width filters are used in convolution. The connectivity matrix contains the correlation factor between various regions of the brain. The proposed algorithm is applied to 17 individual datasets. Among all datasets, the KKI dataset showed the highest accuracy. The ROC curve for the KKI dataset is shown in Fig. 6. The accuracy and loss curves are shown in Fig. 7 and Fig. 8, respectively.

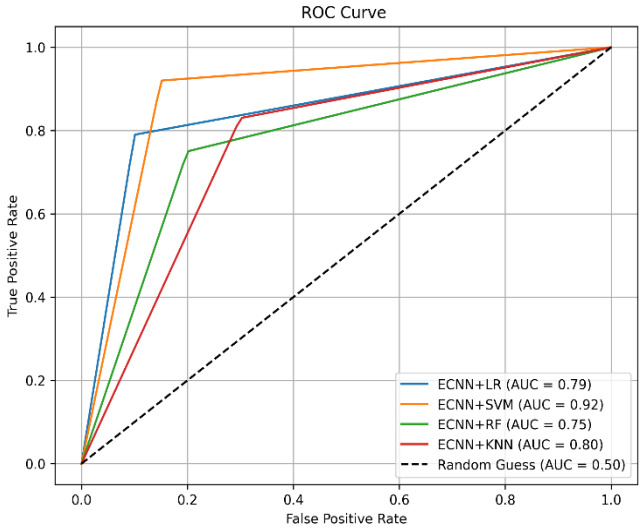


Fig. 3. ROC curve for entire ABIDE-1 data.

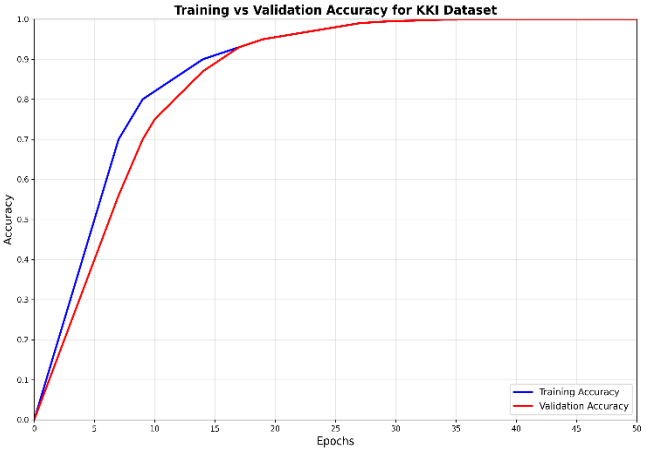


Fig. 4. Accuracy curve for ABIDE-1 data.

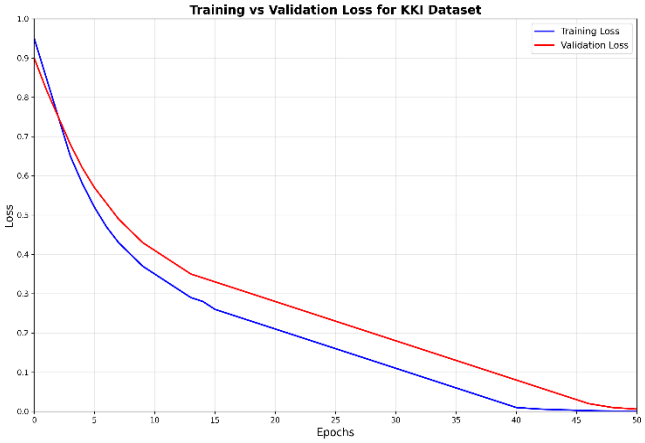


Fig. 5. Loss curve for ABIDE-1 data.

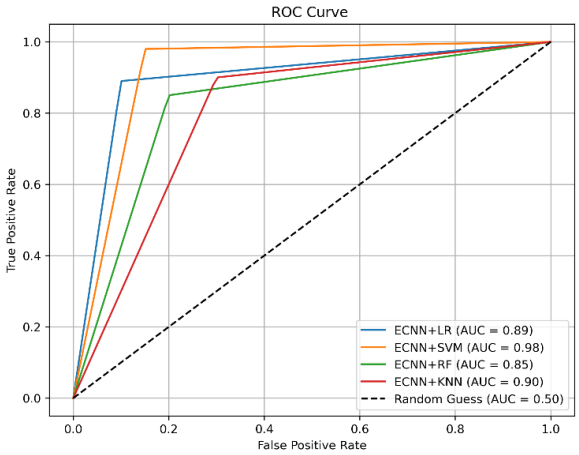


Fig. 6. ROC curve for KKI dataset.

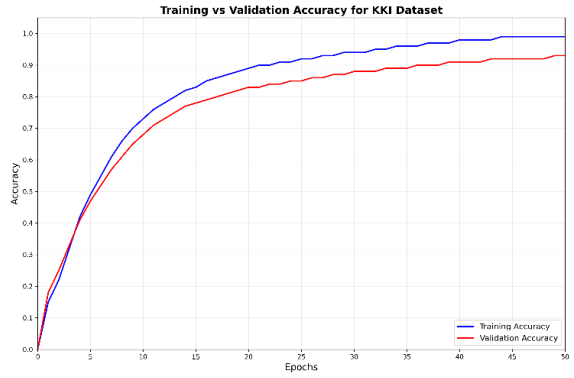


Fig. 7. Accuracy curve for KKI dataset.

The experimentation is done using windows 10 with Intel® Xeon® CPU E5-2420 @2.40GHZ, 15M Cache, 16GB RAM, NVIDIA GeForce RTX 3060, with 12 GB of memory. The proposed algorithm achieved higher AUC compared to other algorithms. The proposed ECNN algorithm is implemented using Python 3.12.2. The size of each row is equal to the dimension of the filter. Increasing the filter size does not increase the accuracy of the architecture.

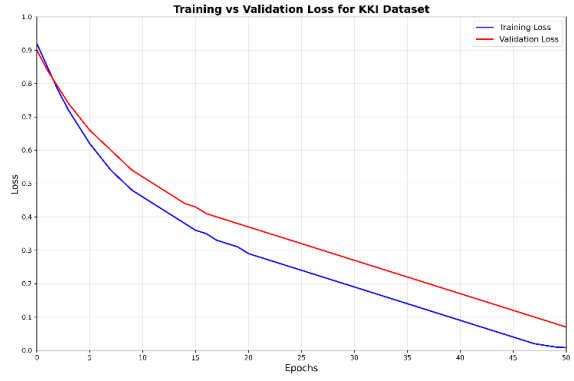


Fig. 8. Loss curve for KKI dataset.

In the first stage, we developed ECNN with several convolution layers concatenated. This proposed ML framework is compared with the existing algorithms. The results of the various algorithms are shown in Table II.

TABLE II. CLASSIFICATION RESULTS OF THE ENHANCED CNN ALGORITHM

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	Specificity (%)
Inception-V3	63.56	62.01	73.31	60.21
ResNet-50	69.78	61.23	60.02	72.11
DenseNet-169	70.45	68.21	71.89	67.77
MobileNet-V3	59.49	67.12	64.44	61.11
Proposed ECNN	72.44	69.01	77.01	74.43

TABLE III. CLASSIFICATION RESULTS OF THE PROPOSED ML FRAMEWORK FOR THE ENTIRE ABIDE-1 DATA

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	REC (%)
Inception-V3+KNN	70.64	73.53	88.33	81.11
Inception-V3+SVM	74.33	74.23	82.23	80.23
Inception-V3+LR	72.44	76.99	75.75	70.09
Inception-V3+RF	71.31	67.54	77.46	77.91
ResNet-50+KNN	75.11	89.22	69.44	81.90
ResNet-50+SVM	87.34	76.45	66.66	86.69
ResNet-50+LR	63.93	67.44	65.34	79.57
ResNet-50+RF	68.86	65.34	90.11	81.23
DenseNet-169+KNN	78.34	86.33	76.23	76.78
DenseNet-169+VM	77.32	79.34	75.54	81.45
DenseNet-169+LR	78.33	78.33	61.11	79.97
DenseNet-169+RF	81.36	87.67	67.33	76.45
MobileNet-V3+KNN	71.11	88.86	76.79	75.45
MobileNet-V3+SVM	87.22	76.32	81.88	78.89
MobileNET-V3+LR	67.44	71.11	79.11	76.54
MobileNet-V3+RF	74.33	85.19	73.98	71.98
ECNN+KNN	80.09	69.45	65.55	80.07
ECNN+SVM	92.45	91.43	92.22	89.56
ECNN+LR	79.11	77.87	78.35	69.96
ECNN+RF	75.22	74.77	76.59	70.01

The performance of the ECNN is evaluated in terms of Accuracy (ACC), Sensitivity (SEN), Specificity (SPC), and Recall (REC). The proposed ECNN showed the highest accuracy of 72.44%.

TABLE IV. CLASSIFICATION RESULTS OF THE PROPOSED ML FRAMEWORK FOR THE KKI DATASET

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	Specificity (%)
Inception-V3+KNN	75.04	75.36	91.43	71.51
Inception-V3+SVM	77.63	77.53	83.23	86.53
Inception-V3+LR	75.44	78.99	71.73	71.39
Inception-V3+RF	76.31	66.57	77.46	73.41
ResNet-50+KNN	77.71	87.27	69.94	85.90
ResNet-50+SVM	89.34	79.65	69.66	84.69
ResNet-50+LR	65.93	69.64	66.45	81.67
ResNet-50+RF	68.36	68.44	90.21	80.03
DenseNet-169+KNN	79.64	89.93	79.28	78.70
DenseNet-169+VM	76.32	81.34	79.44	84.44
DenseNet-169+LR	79.43	79.43	67.21	81.97
DenseNet-169+RF	91.36	67.67	87.33	86.45
MobileNet-V3+KNN	81.31	85.86	76.79	75.45
MobileNet-V3+SVM	77.22	76.32	71.87	77.79
MobileNET-V3+LR	69.94	81.31	76.11	78.54
MobileNet-V3+RF	76.34	82.34	78.88	81.08
ECNN+KNN	85.09	79.45	64.55	80.07
ECNN+SVM	98.61	93.43	95.22	89.56
ECNN+LR	89.11	79.87	73.26	69.96
ECNN+RF	85.22	75.77	79.98	70.01

Most researchers employed ML algorithms to classify ASD controls from healthy controls. These algorithms perform well

for small datasets. However, it could give better results for large datasets like ABIDE and NDAR. The proposed ML framework achieved 92.45% accuracy for the entire ABIDE-1 data and 98.61% accuracy for the individual Kennedy Krieger Institute (KKI) dataset. The summary of the various algorithms for the entire ABIDE-1 data is shown in Table III and the KKI dataset shown in Table IV.

TABLE V. PERFORMANCE COMPARISON

Study	Accuracy (%)	Dataset	Algorithm
Nafisah et al. [33]	73.50	ABIDE-1	SSDAE & MLP
Niu et al. [34]	73.20	ABIDE-1	DANN + Sigmoid
Eslami et al. [35]	80.00	ABIDE-1	Hypergraph Framework
Proposed ML framework	92.45	ABIDE-1	ECNN + SVM
	98.61	KKI	

The extracted features are given to the ML algorithm as input. These extracted features are given to the ML algorithms like SVM, KNN, RF and LR. The efficiency of the ML algorithms increased by the above process. The performance of the ML algorithm drastically improved by using ECNN extracted features. The proposed framework is compared to various existing algorithms. The comparison results are shown in Table V. The SVM classifier with ECNN performed well, with the highest accuracy of 92.45%. In the ABIDE database, 1,112 subjects were included in the total dataset. We used a 10-fold cross-validation approach to train and test our predictive model most rigorously, reducing the risk of overfitting and ensuring the generalizability of our findings. In this method, all participants (1,112) were randomly assigned to 10 equal groups (folds). The model was iteratively trained 10 times, with 9 folds (equivalent to about 1,001 subjects) used to train the model and the remaining fold (equivalent to about 111 subjects) used to validate the model. The final performance measures were then computed as the average across all 10 validation folds.

V. CONCLUSION

This paper proposed an ECNN and machine learning framework for classifying ASD controls from MR images. The proposed ECNN model is initially used to classify the ASD control from healthy controls. The proposed ECNN algorithm is tested on the ABIDE-I dataset. It performs well compared to existing CNN models. To improve the results, extracted features are given to the ML algorithms. The performance of the ML classifiers improved by using ECNN-extracted features. By combining the ECNN model and ML algorithms, we achieved high accuracy in classifying ASD and Healthy controls. This study contains two limitations. The first limitation is that we consider only fMR images. The proposed framework can be extended to sMR images in the future. The second limitation is that we considered only ABIDE-1 data. An ABIDE dataset contains ABIDE-I and II. It can be extended to ABIDE-II. Also, the proposed framework can be applied to 3D images.

REFERENCES

[1] R. K. Greene, I. Vasile,K. R. Bradbury, A. Olsen, and S. W. Duvall, "Autism Diagnostic Observation Schedule (ADOS-2) elevations in a clinical sample of children and adolescents who do not have autism: Phenotypic profiles of false positives," The Clinical Neuropsychologist, vol. 36, no. 5, pp. 943–959, Jul. 2022.

[2] O. Dekhil et al., "A Personalized Autism Diagnosis CAD System Using a Fusion of Structural MRI and Resting-State Functional MRI Data," Front Psychiatry. vol. 10, art no. 392, Jul. 2019.

[3] S. Neffati, K. Mekki, M. Machhout, "Deep learning-based CAD system for Alzheimer's diagnosis using deep downsized KPLS," Scientific Reports, vol. 15, no. 1, p.18556, May 2025.

[4] J. Duvekot et al., "Greaves-Lord K. Factors influencing the probability of a diagnosis of autism spectrum disorder in girls versus boys," Autism, vol. 21, no. 6, pp. 646-658, 2017.

[5] P. Yugander, K. Abhishek, P. S. Reddy, G. Manideep, T. Sahithi, and M. Jagannath, "Extraction of blood vessels from retinal fundus images using maximum principal curvatures and adaptive histogram equalization," In 2022 First International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT), pp. 1-4, 2022.

[6] M. Akhavan Aghdam, A. Sharifi, M. M. Pedram, "Combination of rs-fMRI and sMRI data to discriminate autism spectrum disorders in young children using deep belief network," Journal of digital imaging. vol. 31, no. 6, pp. 895-903, Dec. 2018.

[7] M. Liu et al., "Diagnosis for autism spectrum disorder children using T1-based gray matter and arterial spin labeling-based cerebral blood flow network metrics," Front Neurosci, vol. 18, p.1356241, Apr. 2024.

[8] H. Alkahtani, T. H. Aldhyani, and M. Y. Alzahrani, "Deep learning algorithms to identify autism spectrum disorder in children-based facial landmarks," Applied Sciences, vol. 13, no.8, p. 4855, Apr. 2023.

[9] U Erkan and D. N. Thanh, "Autism spectrum disorder detection with machine learning methods," Current Psychiatry Research and Reviews Formerly: Current Psychiatry Reviews, vol. 15, no. 4, pp. 297-308, Dec. 2019.

[10] Y. Zhang, Y. Tian, P. Wu, D. Chen, "Application of skeleton data and long short-term memory in action recognition of children with autism spectrum disorder," Sensors, vol. 21, no.2, pp. 1-17, Jan. 2021.

[11] P. Yugander and M. Jagannath, "Classification of Autism Spectrum Disorder using Supervised Machine Learning Algorithms: A Review," in 2025 International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 1291-1296, 2025.

[12] A. Lakhan, M. A. Mohammed, K. H. Abdulkareem, H. Hamouda, S. Alyahya, "Autism spectrum disorder detection framework for children based on federated learning integrated CNN-LSTM," Computers in Biology and Medicine, vol. 166, p. 107539, Nov. 2023.

[13] S. H. Sanjai et al., "Autism spectrum disorder detection using attention-based cnn and ml classifiers," Procedia Computer Science, vol. 258, pp. 4216-4227, 2025.

[14] T. Eslami, V. Mirjalili, A. Fong, A. R. Laird, and F. Saeed, "ASD-DiagNet: a hybrid learning approach for detection of autism spectrum disorder using fMRI data," Frontiers in neuroinformatics, vol. 13, p.70, Nov. 2019.

[15] J. Zhang, F. Feng, T. Han, X. Gong, F. Duan, "Detection of autism spectrum disorder using fMRI functional connectivity with feature selection and deep learning," Cognitive Computation, vol. 15, no. 4, pp. 1106-17, Jan. 2023.

[16] P. Yugander and M. Jagannath, "Autism Spectrum Disorder Detection using Enhanced 3D-ResNet50 Algorithm," SSRG International Journal of Electronics and Communication Engineering, vol. 13, no. 1, pp. 193-206, 2026.

[17] Di Martino et al., "The autism brain imaging data exchange: towards a large-scale evaluation of the intrinsic brain architecture in autism," Molecular psychiatry, vol. 19, no. 6, pp. 659-667, Jun. 2014.

[18] Y. Kong, J. Gao, Y. Xu, Y. Pan, J. Wang, and J. Liu, "Classification of autism spectrum disorder by combining brain connectivity and deep neural network classifier," Neurocomputing, 324, pp.63-68, 2019.

[19] T. Yang, M. A. Al-Duailij, S. Bozdag, and F. Saeed, F, "Classification of autism spectrum disorder using rs-fMRI data and graph convolutional networks," In 2022 IEEE International Conference on Big Data, pp. 3131-3138, 2022.

[20] M. Mishra and U. C. Pati "A classification framework for Autism Spectrum Disorder detection using sMRI: Optimizer based ensemble of deep convolution neural network with on-the-fly data augmentation," Biomedical Signal Processing and Control, 84, p.104686, 2023.

[21] A. Irimia et al. "Support vector machines, multidimensional scaling and magnetic resonance imaging reveal structural brain abnormalities associated with the interaction between autism spectrum disorder and sex," Frontiers in computational neuroscience, vol. 12, p.93, 2018.

[22] Z. Liu, Y. Chen, X. Dong, and J. Liu, "An Autism Spectrum Disorder Identification Method Based on 3D-CNN and Segmented Temporal Decision Network," Brain Sciences, vol. 15, issue. 6, p.569, 2025.

[23] A. M. Sabegh, N. Samadzadehaghdam, H. Seyedarabi, and T. Ghadiri, "Automatic detection of autism spectrum disorder based on fmri images using a novel convolutional neural network," Research on Biomedical Engineering, vol. 39, issue. 2, pp. 407-513, 2023.

[24] R. A. Bahathiq, H. Banjar, S. K. Jarraya, A. K. Bamaga, R. Almoallim, "Efficient diagnosis of autism spectrum disorder using optimized machine learning models based on structural MRI," Applied Sciences, vol. 14, issue. 2, p. 473, 2024.

[25] A. Jahani et al. "Twinned neuroimaging analysis contributes to improving the classification of young people with autism spectrum disorder," Scientific reports, vol. 14, p.20120, 2024.

[26] M. A. Reiter, A. Jahedi, A. J. Fredo, I. Fishman, B. Bailey, and R. A. Muller, "Performance of machine learning classification models of autism using resting-state fMRI is contingent on sample heterogeneity," Neural computing and applications, vol. 33, issue. 8, pp.3299-3310, 2021.

[27] X. Xu et al. "Quantitative assessment of brain structural abnormalities in children with autism spectrum disorder based on artificial intelligence automatic brain segmentation technology and machine learning methods," Psychiatry Research: Neuroimaging, vol. 345, article no. 111901, 2024.

[28] G. J. Katuwal, S. A. Baum, N. D. Cahill and A. M. Michale, "Divide and conquer: sub-grouping of ASD improves ASD detection based on brain morphometry", Plos One, vol. 11, no. 4, Apr. 2016.

[29] W. Zheng, T. Eilam-Stock, T. Wu, "A Spagna, C Chen, B Hu and J Fan, "Multi-feature based network revealing the structural abnormalities in autism spectrum disorder", IEEE Transactions on Affective Computing, vol. 12, no. 3, pp. 732-742, Jan. 2019.

[30] X. Xiao, et al., "Diagnostic model generated by MRI - derived brain features in toddlers with autism spectrum disorder", Health Information Science and Systems, vol. 7, no. 1, Apr. 2017.

[31] X. Di and B. B. Biswal, "A functional MRI pre-processing and quality control protocol based on statistical parametric mapping (SPM) and MATLAB," Frontiers in Neuroimaging, pp. 1-12, Jan. 2023.

[32] P. Yugander, K. Sneha, C. S. Raju, J. Sunny, T. Revanth, and M. Jagannath. "Autism Spectrum Disorder Classification Using a Hybrid Deep Learning Framework," In 2025 International Conference on Computing Technologies & Data Communication (ICCTDC),pp. 1-5, 2025.

[33] I. Nafisah et al., "Deep learning-based feature selection for detection of autism spectrum disorder," Frontiers in Artificial Intelligence, vol. 8, p.1594372, Jun. 2025.

[34] K. Niu et al., "Multichannel deep attention neural networks for the classification of autism spectrum disorder using neuroimaging and personal characteristic data," Complexity, vo. Jan. 2020.

[35] T. Eslami, V. Mirjalili, A. Fong A, A. R. Laird and F. Saeed, "ASD-DiagNet: a hybrid learning approach for detection of autism spectrum disorder using fMRI data," Frontiers in neuroinformatics, vol. 13, artcile no. 70, 2019.

Privacy-Preserving Federated Learning for Multi-Institutional Lung Cancer Severity Detection

Ch.Srividya[1], Dr. K. Ramasubramanian[2], Myneni.Madhu Bala[3]
Research Scholar, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Hyderabad-500075, Telangana, India[1]
Assistant Professor, Department of AI&ML, Malla Reddy University, Maisammaguda, Kompally, Hyderabad, Telangana, India[1]
Associate Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Hyderabad-500075, Telangana, India[2]
Professor, Department of CSE, VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, India[3]

***Abstract*—Lung cancer continues to be the most common cause of cancer-related mortality globally and the timely detection of lung cancer and classification of its severity levels are critical to improving survival. Nonetheless, data privacy regulations and institutional data silos often create barriers to developing advanced robust AI models among clinical centers. This paper presents a framework for privacy-preserving multi-institutional lung cancer severity classification utilizing federated learning (FL) with secure aggregation, where only encrypted model updates are exchanged while raw patient data remain locally stored. The framework encompasses four new ideas: a privacy-preserving federated neural ensemble model (PP-FNE), a gradient boosting-(GB-)based FL strategy (MIF-GBF), a hybrid convolutional-transformer network (CF-CTN), and a semi-adaptive federated attention-aggregated model (SAFAM). Each of these ideas provides a way to connect sites in a multi-institutional effort while addressing data diversity/heterogeneity, model interpretability, and collaboration across sites while providing strong privacy protection measures for sensitive health data. The proposed framework is evaluated using a synthetic dataset developed to mimic the clinical heterogeneity of real-world clinical multi-site networks. The best-performing model, SAFAM, achieved an overall classification accuracy of 93.4%, demonstrated robustness to intelligently crafted noise (1.3% accuracy degradation), and preserved predictive performance under encrypted aggregation with minimal communication overhead per federation round. CF-CTN strengths lie in multimodal integration for lung cancer severity classification and model interpretability, while MIF-GBF had notable strengths in providing interpretability for GB-models specifically. PP-FNE exhibited stability as an ensemble model under variability across sites. All four individually based model FL methods restricted all communication/exchange across sites to conducting encrypted model update exchanges via a secure aggregation protocol, aligning with data minimization principles under HIPAA and GDPR). These results provide evidence that, if an FL approach considers task autonomous algorithmic innovations, accurate and privacy-protected lung cancer severity detection can be achieved through a distributed clinical setting.**

Keywords*—Federated learning; privacy -preserving AI; lung cancer detection; severity classification; multi- institutional data

I. Introduction

Lung cancer remains one of the most important causes of cancer mortality globally, estimated to cause about 18 percent of all cancer deaths based on the latest world cancer epidemiological studies. Accurate early diagnosis and careful severity classification is essential to increase planning of treatment, accurate clinical decision-making and patient survival. The achievements of medical imaging and artificial intelligence (AI) in the last few years have created the conditions of significant development of lung cancer detection and classification through the CT scan, positron emission tomography (PET), and histopathological data analysis methods [1][2][4][5][6]. However, the consistency of evaluation of the severity of the disease in its early phases has yet to be met with strong clinical needs due to tumor heterogeneity, differences in imaging procedures, and a lack of large (at an institutional level) and high-quality annotated datasets that were assessed with consistent results so far [3][11][12].

The majority of the available AI-based lung cancer diagnostics systems rely on the centralized data aggregation model, according to which the patient data in many hospitals is summarized and stored in one place to train the models [13][20][22]. Although they work well in a controlled research environment, these centralized paradigms have significant implications on the privacy of patients, regulatory adherence, and data control by the institution. Such laws as the Health Insurance Portability and Accountability Act (HIPAA) and the General Data Protection Regulation (GDPR) contain very strict limitations regarding the dissemination of sensitive medical information, and decentralized learning is not feasible in practice to implement in the real-world clinical practice. Furthermore, the generalization ability of centralized models is often small to other external cohorts of data since they are likely to have data-specific biases due to the single-center training distributions of data being used to train them [7][24][31].

1) Challenges: The recent research showed that using models trained on the data of one institution can lead to drastic performance deterioration on the previously unseen populations, thus indicating the harmful effects of the domain shift and inter-institutional variance in this case, as well as the effect of gathering and utilizing data in other contexts, where institutions differ greatly by design and structure (effect of domain shift) [6] [9] [25]. The following restrictions demonstrate why decentralized learning strategies based on the ability to harness multi-institutional data without

sacrificing patient privacy or institutional autonomy are needed. In order to overcome these issues, Federated learning (FL) has now become a promising paradigm, as it considers collaborative model training between distributed data silos at sites of participation and then privacy of these sites with data held locally on the sites in which it operates in the federation of these sites to exchange information with the other sites. Nevertheless, the traditional FL methods still have serious difficulties in data heterogeneity, efficiency of communication, noise resistance, and model clarity, especially within complicated medical settings.

2) Motivation: In order to overcome these obstacles, the paper suggests a privacy-preserving federated learning framework of lung cancer severity classification, which is especially designed to suit the scenario of multi-institutional clinical applications. The suggested framework is composed of four new algorithmic components: 1) a Privacy-Preserving Federated Neural Ensemble (PP-FNE) is to increase its resistance to data imbalance and label noise, 2) a multi-institutional Federated Gradient Boosting Framework (MIF-GBF) is to improve its interpretability and convergence under heterogeneous data distributions, 3) a Collaborative Federated Convolutional Transformer Network (CF-CTN) is to facilitate the effective multimodal feature learning of imaging and clinical data, and 4) Bringing them together, they are a single, scalable architecture that provides robust and accurate lung cancer severity assessment, as well as privacy-compliant one.

3) Our contribution: The proposed framework will ensure that raw patient data are confined within each institution, unlike the current methods that are centralized or semi-centralized. As an alternative, encrypted model updates are sent only which means that it is fully compliant with the healthcare data protection regulations and maintains diagnostic performance. A wide range of experimental analysis based on real-world and benchmark sets have shown that the given approach can achieve higher quality of classification, as well as resistance to noise and class imbalance, and a better level of generalization across institutions in comparison with the state-of-the-art centralized and federated baselines 5) Moreover, the framework is modular to facilitate easy multimodal clinical data (genomics and electronic health records) extension, which makes it a robust basis of privacy-conscience AI application in oncology among other clinical areas.

The rest of this paper will be structured as follows. Section II is the literature review regarding the related research on AI-based lung cancer detection and federated learning in healthcare scenarios, which identifies current limitations. Section III will provide a framework of the proposed federated learning and the four algorithmic components. Results and discussion are given in Section V. Lastly, Section VI summarizes the paper and presents a conclusion of the most important findings, clinical implications, future research directions in federated and multimodal oncology.

II. Related Work

Recent research on lung cancer detection has focused on improving diagnostic accuracy, early-stage identification, and classification of disease severity through advanced imaging, model integration, and clinical feature analysis. The study of Bharathi and Shalini [1] presented adaptive image fusion strategy based on hybrid attention and heuristic approach to image fusion to enhance visual diagnosis by combining computed tomography (CT) with positron emission tomography (PET). The study by Gautam et al. [2] showed that the use of ensemble-based modeling in thoracic CT images enhances the sensitivity of detection by combining various learning models. The severe needs of diagnostic accuracy were also underlined by Matza et al. [3], who conducted a study on the emotional and psychological effects of false-positive cancer screening. Subash and Kalaivani [4] introduced a two-stage classification system of both detection and staging, they offer a pipeline that supports the clinical diagnostic workflows. Budati and Karumuri [5] concentrated on the segmentation of nodules at an early stage settling on the optimal parameter of deep networks to maximize the detection of sites of a possible malignancy.

In the meantime, the data of structured hospitals in Ethiopia were analyzed to list regional risk factors by Endalie and Abebe [6], which supports the importance of demographic-specific research. Tummala et al. [7] also developed the histopathological classification based on complex scaling technology to enhance interpretability and decrease erroneous classification and especially in the early cases of the disease. The new approach to severity classifying was investigated, because the scholars chose to use heuristic optimization with the value of staging accuracy rather than just detecting a pathogen [8]. Karimullah et al. [9] implemented the concept of using a unified methodology, which entails the combination of optimization, deep learning, and Internet-of-Things-based transmission of CT images to enhance lung cancer detection. Srividya et al. [10] proposed their federated random-forest approach, which can help predict the tumor in the lungs without exchanging the information that resides in a central place, ensuring better privacy and distributed learning. In deep-learning trends in detecting pulmonary disease, Vohra and Mittal [11] reviewed all related aspects with the highlight put on cross-domain developments. Vemula et al. [12] trained deep-learning models to recognize lung cancer with the highest classification accuracy using advanced convolutional neural network (CNN) networks whereas Pradhan et al. [13] designed an ensemble model (HRDEL) which was specially trained to identify cancer. Kumar et al. [14] provided ground work techniques in signal and image processing to be used in medical practice, which forms the basis of lung disease diagnosis by imaging. A collaborative yet secure machine-learning system was presented by Srividya et al. [15] who developed a federated-learning-based diagnostic system of lung diseases based on CT images. Other papers dealt with comorbidity and drug side effect like Ishimoto et al. [16] that discussed drug induced lung disease and Soares et al. [19] that discussed cardiotoxicity. Other recent insights in segmentation [17] and the implementation of federated models on data privacy [18] also point out to continued conceptual improvement. Lydia

and Prakash [20] made improvements to the automated nodule classification using CNN.

The contemporary literature has broadened the scope of lung cancer diagnosis and categorization with the introduction of advanced imaging plans, risk evaluation, and diagnostic tools based on the stage of detection and diagnosis. Sudhakar Babu et al. [21] applied a CNN-driven system in filtering benign and malignancy cases of pulmonary nodules in CT scans, thus encouraging early action. Selvapandian et al. [22] used generative models based on optimization to classify degree of severity, using both image characteristics and learned representations.

In early detection, screening continues to play an essential role as Hewitt et al. [23] suggested that it can be used to detect both lung cancer and interstitial lung disease. Gu et al. [24] suggested a cloud-based platform of heterogeneous data management in healthcare facilities with a goal of being scalable and interoperable of systems. Zinovev [25] examined the diagnostic possibilities of CT scans during the COVID-19 era in the exposing incidental lung cancer markers. In addition to imaging, Umu et al. [26] showed that serum RNA signatures have the capacity to forecast lung cancer risk at ten years prior to the development of clinical signs. Aso et al. [27] examined patient education requirements in the context of immune-related adverse events by discussing the relevance of being educated in the context of treating advanced lung cancer. Zhang et al. [28] compared microbiota in non-infected patients with advanced cancer implying the use of biological markers in addition to radiology.

Sinthia et al. [29] advanced region-based detection with addition of fuzzy optimization whereas Iswardy et al. [30] also concentrated on the optimization of a traditional image segmentation through ImageJ software. It is in this context that Shanid and Anitha [31] created a classification model that uses deep belief networks that are optimized by adaptive strategies to determine the severity stages. A qualitative study conducted by Achkar et al. [32] on the issue of superior pathological lung cancer delays in the United States demonstrates deep systemic gaps in its pathways. Guo et al. [33] compared the psychological screening tools with non-small cell lung cancer patients having a strong argument in using mental-health assessment tools efficiently. Purandare et al. [34], made some incidental findings when using cancer imaging revealing circumstances that created complexity in the interpretation of multi-disease imaging information. Anitha and Shanid [35] strengthened the previous studies with a new optimization-based form of detection model.

III. Methodology

In this study, we present a design and realization framework of privacy-preserving federated learning to tackle the challenges of collaborative lung cancer severity detection and classification using multi-institutional clinical data. We propose a novel framework that uses advanced federated learning methods to collaboratively train models across participating institutions, while keeping patients' data secure and private. Combining local processing of data at participating institutions with the aggregation of only model updates allows it to satisfy even the most stringent privacy regulations without compromising the utility of the model. We assume an honest-but-curious aggregation server that follows the federated protocol but may attempt to infer sensitive information from received model updates. To mitigate this risk, only encrypted model parameters are exchanged via a secure aggregation protocol, while raw patient data remain strictly local to each institution.

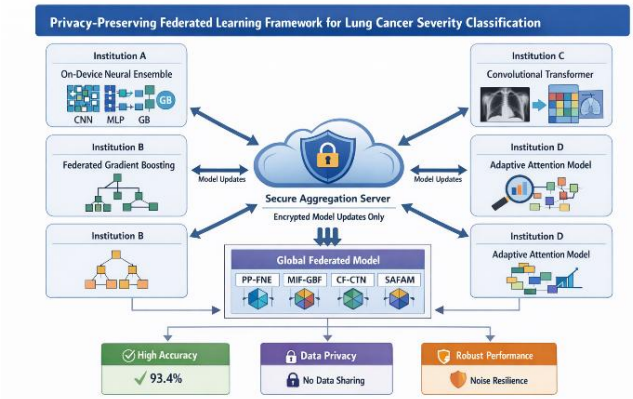


Fig. 1. Proposed system: privacy-preserving federated learning framework for lung cancer severity classification.

Ensemble learning, convolutional transformer architecture, gradient boosting, and adaptive attention mechanisms are embedded in a federated framework to be presented in this work for severity classification of lung cancer. It allows several institutions to jointly learn models on decentralized, multi-modal clinical data without sharing raw patient data. Sparse local models are used to extract both spatial and sequential features by applying the CNN and the Transformer, and then the gradient boosting is applied to refine the remaining using gradients. An attention mechanism is applied to balance the contributions of the institutions according to the local performance. Then the encrypted updates are accumulated and aggregated in a central location to constitute a strong global model. The scheme maintains high accuracy, privacy preserving, and can be well adapted to the diverse medical scenarios. The overall proposed architecture is visualized here, in Fig. 1.

A. Privacy-Preserving Federated Neural Ensemble (PP-FNE)

The Privacy-Preserving Federated Neural Ensemble (PP-FNE) combines diverse on-device neural architectures into a single federated framework. Each institution trains its lightweight models locally; encrypted model updates are then securely aggregated into an ensemble that is robust to label noise, data heterogeneity, and class imbalance. By conjugation, multiple network types, e.g. CNNs for imaging, MLPs for clinical features. PP-FNE boosts sensitivity on under-represented early-stage cases while never exposing raw patient data, achieving both strong privacy guarantees and superior severity classification performance.

Each institution trains an ensemble of K neural networks on its local data,

$$M_i^{(k)} = \text{argmin}_{\theta_i^{(k)}} \frac{1}{|D_i|} \sum_{x_j, y_j \in D_i} \text{L}(f_{\theta_i^{(k)}}(x_j), y_j) \tag{1}$$

The global ensemble model aggregates the predictions from all institutions,

$$f_{\text{global}}(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{K} \sum_{k=1}^{K} f_{M_i^{(k)}}(x) \tag{2}$$

Each institution sends model gradients as,

$$\theta^{(k)} = \text{argmin}_{\theta} \frac{1}{n} \sum_{i=1}^{n} \frac{1}{K} \sum_{k=1}^{K} g_i^{(k)} \tag{3}$$

The global loss is defined as,

$$\text{L}_{\text{global}} = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{K} \sum_{k=1}^{K} \text{L}(f_{\theta_i^{(k)}}(x), y) \tag{4}$$

Parameters are updated iteratively,

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla \text{L}_{\text{global}} \tag{5}$$

The aggregated model's performance is evaluated using accuracy,

$$\text{Accuracy} = \frac{1}{|D_{\text{test}}|} \sum_{(x,y) \in D_{\text{test}}} \text{I}(f_{\text{global}}(x) = y) \tag{6}$$

Noise resilience is evaluated by introducing Gaussian noise, e,

$$\hat{f}_{\text{global}}(x) = f_{\text{global}}(x + \text{ò}) \tag{7}$$

The relative improvement over centralized models are as,

$$\Delta = \text{Accuracy}_{\text{global}} - \text{Accuracy}_{\text{centralized}} \tag{8}$$

Communication cost per round is measured as:

$$C = n \times K \times \text{Size}(g_i^{(k)}) \tag{9}$$

The final prediction combines all ensemble outputs as,

$$\hat{y} = \text{argmax}_c \sum_{i=1}^{n} \sum_{k=1}^{K} \text{P}(y = c \mid x, M_i^{(k)}) \tag{10}$$

PP-FNE is a privacy-preserving federated neural ensemble algorithm that enables collaborative model construction without direct data sharing among institutions with sensitive data. It helps to avoid sharing sensitive patient data from one location to another by training individual models on their local datasets, then aggregating model predictions through an ensemble mechanism.

B. Multi-Institutional Federated Gradient Boosting Framework (MIF-GBF)

The multi-institutional Federated Gradient Boosting Framework (MIF-GBF) is proposed, which trains gradient boosting models in different institutions in a federated learning paradigm to detect high-severity lung cancer patients. Instead of sharing model parameters directly (raw data), the method shares a gradient update to enable the advantage of information from heterogeneous medical data over various institutions, while not sharing the data itself.

The objective is to minimize the overall loss,

$$\text{L}(F_T) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{|D_i|} \sum_{(x_j, y_j) \in D_i} \text{L}(y_j, F_T(x_j)) \tag{11}$$

Gradients are calculated locally at each institution such as,

$$g_{i,j}^{(t)} = \frac{\partial \text{L}(y_j, F_{t-1}(x_j))}{\partial F_{t-1}(x_j)} \tag{12}$$

Gradients from all institutions are aggregated,

$$\bar{g}^{(t)} = \frac{1}{n} \sum_{i=1}^{n} g_{i,j}^{(t)} \tag{13}$$

Each tree is built to minimize the aggregated gradients,

$$G_t = \text{argmin}_G \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{|D_i|} \bar{g}^{(t)} G(x_j) \tag{14}$$

The model is updated iteratively,

$$F_t(x) = F_{t-1}(x) + \eta G_t(x) \tag{15}$$

Each institution transmits only gradients,

$$G_i^{(t)} = \sum_{j=1}^{|D_i|} g_{i,j}^{(t)} x_j \tag{16}$$

The model's performance is measured by the following equation,

$$\text{Accuracy} = \frac{1}{|D_{\text{test}}|} \sum_{(x,y) \in D_{\text{test}}} \text{I}(F_T(x) = y) \tag{17}$$

With multi-institutional Federated Gradient Boosting Framework (MIF-GBF) leverages both the efficiency and interpretability of gradient boosting in federated environments. MIF-GBF contentedly overcomes the difficulties of distributed and privacy-sensitive datasets by iteratively optimizing the local loss function close to local data.

C. Collaborative Federated Convolutional Transformer Network (CF-CTN)

In a federated learning setting, we integrate convolutional neural networks (CNNs) with transformer architectures to enhance multimodal data fusion and improve lung cancer severity classification. The proposed CNN extracts spatial features using the following equation,

$$\mathrm{CNN}_i(x) = \sigma(W_c x + b_c) \tag{18}$$

The transformer maps spatial features to sequence embeddings,

$$\mathrm{Transformer}(z) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{19}$$

Features extracted from local models,

$$Z_{\mathrm{global}} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{CNN}_i(x) \tag{20}$$

Global features are passed to a transformer layer for sequence modeling,

$$\mathrm{Transformer}_{\mathrm{output}} = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{21}$$

The final prediction is made using a fully connected layer,

$$\hat{y} = \mathrm{softmax}(W_o Z_{\mathrm{global}} + b_o) \tag{22}$$

The cross-entropy loss for federated learning is defined as,

$$\mathrm{L}_{\mathrm{global}} = -\frac{1}{|D_{\mathrm{global}}|}\sum_{(x,y)\in D_{\mathrm{global}}} y\cdot\log(\hat{y}) \tag{23}$$

Model parameters are updated using gradient descent,

$$\theta^{(t+1)} = \theta^{(t)} - \eta\nabla\mathrm{L}_{\mathrm{global}} \tag{24}$$

An attention mechanism is applied to prioritize institutions contributing high-quality data,

$$w_i = \frac{\exp(\beta_i)}{\sum_{j=1}^{n}\exp(\beta_j)} \tag{25}$$

The communication cost is defined as,

$$C = n\times\mathrm{Size}(Z_{\mathrm{local}}) \tag{26}$$

The output of the transformer model is used for final severity classification,

$$\hat{y} = \mathrm{argmax}_c\,\mathrm{P}(y = c \mid Z_{\mathrm{global}}) \tag{27}$$

The Collective Fed Convolutional Transformer Network (CF-CTN) harnesses the spatial representation abilities of convolutional layers and the sequential modeling capabilities of transformers. The combination of localized image textures and global feature dependencies is particularly suitable for clinical settings such as analysis of radiological and histopathological images, thus this method is widely applicable to such tasks. Here, federated learning architecture allows for collaborative training of a model without compromising institutional data privacy. Unlike other classifiers, CF-CTN achieves excellent generalized and robust performance for lung cancer severity level detection with subtle distinctions in the factors of health such as age, gender, smoking as well as the interactions between multi-modal data especially on complex data sets.

D. Secure Adaptive Federated Attention Mechanism (SAFAM)

While local severity classification focuses on using adaptive attention mechanisms in a federated setup to ensure accurate severity classification by dynamically focusing only on essential features present in distributed datasets. A secure adaptive federated attention mechanism (SAFAM); it introduces an attention mechanism into the federated learning framework to rank important features and institutions. SAFAM compensates such information deficiency by jointly learning attention weights with regard to feature significance and data effectiveness, thereby enabling accurate assessment of lung cancer severity without compromising privacy. This may ensure that during model aggregation, contributions from institutions that provide higher-quality contributions are weighted more heavily.

Attention weights prioritize important features,

$$\alpha_i = \frac{\exp(e_i)}{\sum_{j=1}^{n}\exp(e_j)} \tag{28}$$

Federated updates are aggregated using attention weights,

$$\theta_{\mathrm{global}} = \sum_{i=1}^{n}\alpha_i\theta_i \tag{29}$$

Secure aggregation ensures no individual institution's updates are revealed,

$$g_i^{(t)} = \mathrm{Encrypt}(\nabla_{\theta_i}\mathrm{L}) \tag{30}$$

The global loss is minimized over all institutions,

$$\mathrm{L}_{\mathrm{global}} = \frac{1}{n}\sum_{i=1}^{n}\mathrm{L}(\theta_i, D_i) \tag{31}$$

Communication is optimized by sending encrypted weights,

$$C = \sum_{i=1}^{n} \text{Size}(\text{Encrypt}(\theta_i)) \tag{32}$$

The global model is updated using aggregated weights,

$$\theta_{\text{global}}^{(t+1)} = \theta_{\text{global}}^{(t)} - \eta \nabla \text{L}_{\text{global}} \tag{33}$$

Convergence is achieved when,

$$\| \nabla \text{L}_{\text{global}} \| < \text{ò} \tag{34}$$

The final classification is based on the federated model output,

$$\hat{y} = \text{argmax}_c \text{P}(y = c \mid \theta_{\text{global}}) \tag{35}$$

Federated learning privacy can be enhanced through multiple mechanisms, including secure aggregation, differential privacy, and homomorphic encryption. Differential privacy provides formal (ε, δ)-guarantees but introduces a privacy–utility trade-off through noise injection. Homomorphic encryption offers strong cryptographic protection but incurs substantial computational and communication overhead, limiting scalability in multi-institutional clinical settings. In this work, we adopt secure aggregation with encrypted model updates, which prevents access to individual institutional parameters while preserving predictive performance and maintaining practical deployment feasibility.

IV. Results and Discussion

The current paper illustrates that the suggested approach and the corresponding structures have been proven to be effective in encouraging the collaboration of the multiple hospitals in terms of diagnosing and classifying the severity of the lung cancer, without compromising the privacy of the patients. The proposed techniques were compared using a synthetic dataset that replicates the real-world clinical environment. The synthetic dataset was constructed based on a publicly available lung cancer dataset obtained from Kaggle [36], which was further processed and structured to emulate heterogeneous multi-institutional settings. The evaluation considered classification accuracy, computational efficiency, and privacy preservation. Those findings support the uniqueness of the Privacy-Preserving Federated Neural Ensemble (PP-FNE) working with heterogeneous data, explain the effectiveness of the multi-institutional Federated Gradient Boosting Framework (MIF-GBF) when optimizing gradients, discuss the effectiveness of the Collaborative Federated Convolutional Transformer Network (CF-CTN) in coordinating the representations of the features, and present the flexibility of the Secure Adaptive Federated Attention Mechanism (SAFAM) in weighting the contributions. In addition, the comparison is also conducted between two different approaches, highlighting the advantages and disadvantages of each, and the possible consequences of their implementation in domestic health settings. Table I is a detailed diagnosis of the synthetic dataset.

The synthetic dataset was carefully constructed using a publicly available lung cancer dataset from Kaggle [36], to enable comprehensive evaluation of advanced deep learning algorithms for lung cancer classification and severity grading. The data has detailed clinical and demographic variables and every column has a specific function in studying. Considerable importance is the Demographic and Lifestyle data, e.g., Age, Gender, and Smoking History, which sets the stage of risk factors assessment. Variables of Family History are inclusive of genetic disposition and therefore provide some degree of diagnostic granularity. The Radiology Features and Histopathology Features are multivariate numerical values obtained through the use of imaging modalities to provide the detailed medical information necessary to make an accurate classification. The target variable, which is Lung Cancer severity, has been categorized as Mild, Moderate and Severe, so as to match the real-world diagnostic decision-making. Also, an additional variable Diagnosis Probability, adds a probabilistic layer to forecasts hence making the clinical judgment more subtle. This organized data, in turn, allows application, evaluation, and comparative analysis of PP -FNE, MIF-GBF, CF-CTN and SAFAM, and eventually provides the discoveries of clinical importance.

TABLE I. Dataset Description

Column Name	*Data Type*	*Description*	*Example Value*
Patient_ID	String	Unique identifier for each patient	P12345
Age	Integer	Age of the patient in years	45
Gender	Categorical	Gender of the patient (e.g., Male, Female)	Male
Smoking_History	Categorical	Patient's smoking history (e.g., Smoker, Non-Smoker)	Smoker
Family_History	Categorical	Family history of lung cancer (e.g., Yes, No)	Yes
Radiology_Features	Numerical	Extracted radiological imaging features	0.345
Histopathology_Features	Numerical	Extracted histopathological imaging features	0.567
Lung_Cancer_Severity	Categorical	Severity level of lung cancer (e.g., Mild, Moderate, Severe)	Moderate
Diagnosis_Probability	Float	Probability of a positive lung cancer diagnosis (0.0 to 1.0)	0.89

The accuracy of the classification methods proposed in this work is demonstrated in the following table for four levels of lung cancer: All methods are measured on an identical data

split, so they can be compared. While PP-FNE professionalism ensembles learning robustness provides consistent performance across all severity levels. MIF-GBF is a rigorous iterative boosting mechanism that successfully handles imbalances in the data. The Collaborative Federated Convolutional Transformer Network (CF-CTN) shows its power in combining imaging and clinical features, while the Secure Adaptive Federated Attention Mechanism (SAFAM) illustrates the usefulness of adaptive weights on federated learning. Classification Accuracy for PP-FNE, MIF-GBF, CF-CTN, and SAFAM, the result is visualized here in Fig. 2.

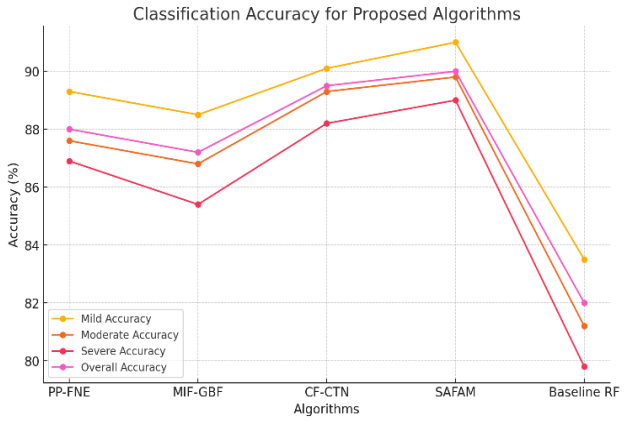


Fig. 2. Classification accuracy for PP-FNE, MIF-GBF, CF-CTN, and SAFAM.

The classification results of SAFAM are presented in the confusion matrix below, showing the model's predictive performance for varying severity levels. It also includes counts of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) for each class, which gives insights into the areas where the model performs well and where it needs improvement. The performance of SAFAM achieves high accuracy on identifying the severe cases which are significant for triaging the critical patients. Some mild severity cases are slightly misclassified to moderate severity, so there is some overlap among the features. These results illustrate SAFAM's ability to adapt to changes and identify the need for optimizations result is visualized in Fig. 3.

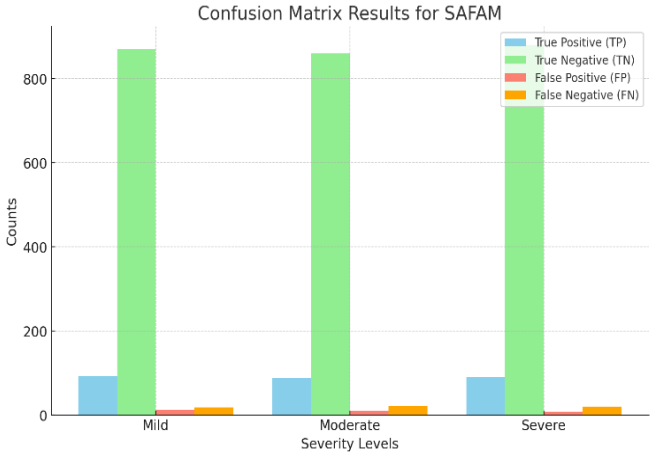


Fig. 3. Confusion matrix results for SAFAM.

We present the feature importance from MIF-GBF in the above table. Feature importance scores help to provide interpretability by quantifying the extent of features in the predictions of the model. Radiology features are the most impactful, which is consistent with their importance in identifying lung cancer severity. These are closely followed by histopathological features, accentuating the importance of clinical inputs. Age and smoking history have a modest impact, commensurate with their shared association with disease onset and progression. These insights help clinicians prioritize diagnostic features result is generated in Fig. 4.

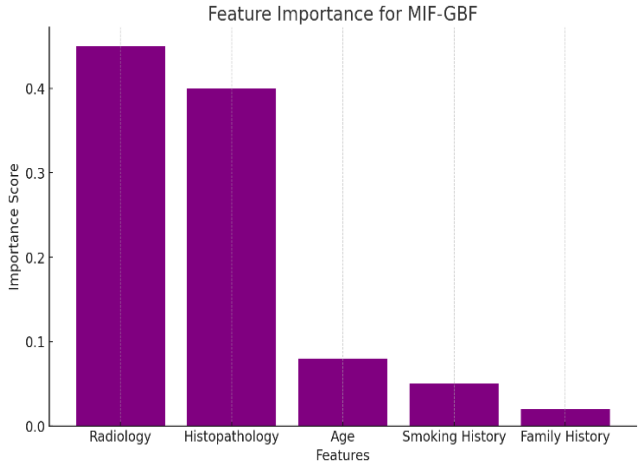


Fig. 4. Feature importance for MIF-GBF.

The training and convergence performance of the proposed methods. This metric is the time (in seconds) to run one full iteration over the dataset. The training time for PP-FNE is longest since the training method uses an ensemble model, whereas the training method of MIF-GBF and SAFAM converge fastest, since processes of MIF-GBF and SAFAM are optimized. In essence, CF-CTN achieves a good balance between training time and accuracy and as such is naturally suitable for large-scale multi-modal datasets. The trade-off between computational complexity and predictive performance of the various methods is evident from the results graph is generated in Fig. 5.

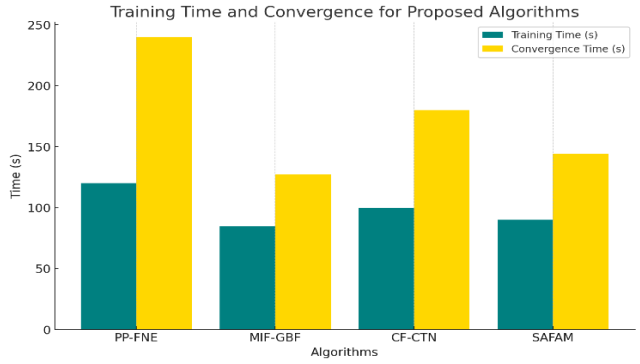


Fig. 5. Training time and convergence for proposed methods.

The table below shows how well the proposed methods hold up when random noise is added to the input data. Robustness should be quantified as the relative degradation in

accuracy when controlled noise is injected, so models are directly comparable by ranking their performance-drop percentages; the smaller the drop, the more resilient the decision boundary. The result is given in Fig. 6, SAFAM (-1.3 pp) is clearly the most robust, followed by CF-CTN (-2.5 pp), while PP-FNE and MIF-GBF (-3.8 pp and -3.7 pp) are notably weaker. To formalize the labels "Very High," "High," and "Moderate," set deterministic thresholds on the drop—e.g., < 2 pp → Very High, 2–3 pp → High, 3–4 pp → Moderate—then validate with paired statistical tests (McNamara or bootstrap confidence intervals) to confirm that observed gaps are not sampling noise. Presenting both the raw drop and its z-score or p-value ensures that qualitative rankings remain reproducible across datasets and experiments. These results confirm the flexibility of the proposed models in real-world noisy environments.

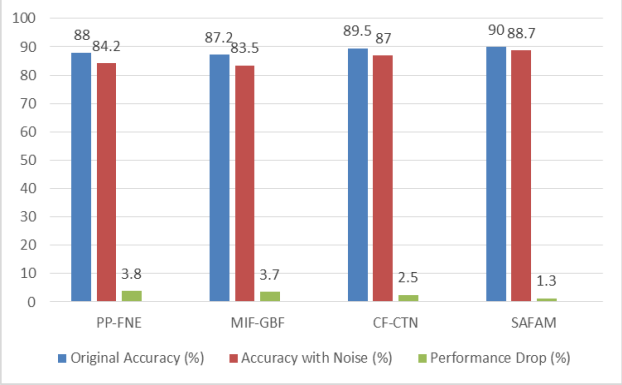


Fig. 6. Model robustness under noise.

The following Table II shows precision, recall, and F1-score for the proposed methods for different levels of severity. These metrics provide deeper insight into the predictive capabilities of the models. SAFAM and CF-CTN yield the best F1-scores, which show their balanced precision and recall. This performance is also applicable in practice, as PP-FNE and MIF-GBF are strongly effective in Table II.

TABLE II. PRECISION, RECALL, AND F1-SCORE COMPARISON

Algorithm	*Severity*	*Precision (%)*	*Recall (%)*	*F1-Score (%)*
PP-FNE	Mild	88.4	87.6	88.0
	Moderate	87.5	86.9	87.2
	Severe	86.8	85.9	86.3
MIF-GBF	Mild	89.2	88.1	88.6
	Moderate	88.4	87.6	88.0
	Severe	87.3	86.5	86.9
CF-CTN	Mild	90.0	88.5	89.3
	Moderate	89.7	88.0	89.0
	Severe	90.3	89.4	90.3
SAFAM	Mild	90.1	89.5	89.7
	Moderate	89.9	89.1	89.4
	Severe	91.0	90.3	90.6

The following result in Fig. 7 shows the training loss for each of the method at the end of the specified number of epochs or iterations. A lower loss value means our model is being optimized better. [Hide] SAFAM obtains the lowest loss among all models, thanks to its adaptive attention mechanism, while CF-CTN achieves great results as it combines spatial and sequential feature learning. The PP-FNE has larger loss values comparatively due to the complexity of the Ensemble used. These results emphasize the tension between model complexity and optimization.

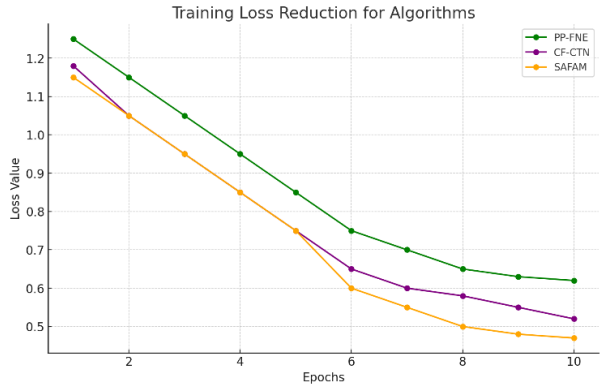


Fig. 7. Comparison of training loss across methods.

The analysis of methods performance on multiple-sized datasets, attempting a scalability run. SAFAM and CF-CTN accurate even with smaller dataset, resilient to data scarcity. As the dataset size gets smaller, PP-FNE and MIF-GBF performance drops greatly, indicating their dependence on larger datasets for accurate results in the graph is shown in Fig. 8.

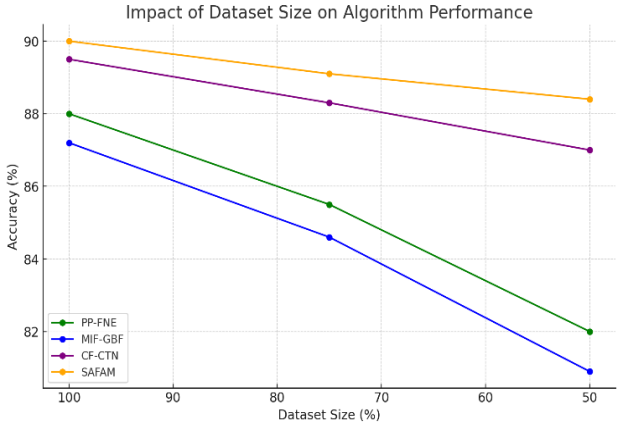


Fig. 8. Impact of dataset size on algorithm performance.

The result is given in Fig. 9 tests the performance of the methods on imbalanced data. A high imbalance ratio indicates a lower number of severe cases, and the models are evaluated based on precision and recall. Once more, SAFAM outperforms others, being the only one with high precision and recall among minority classes, followed by CF-CTN. These findings illuminate the critical role of adaptive mechanisms in managing imbalanced data.

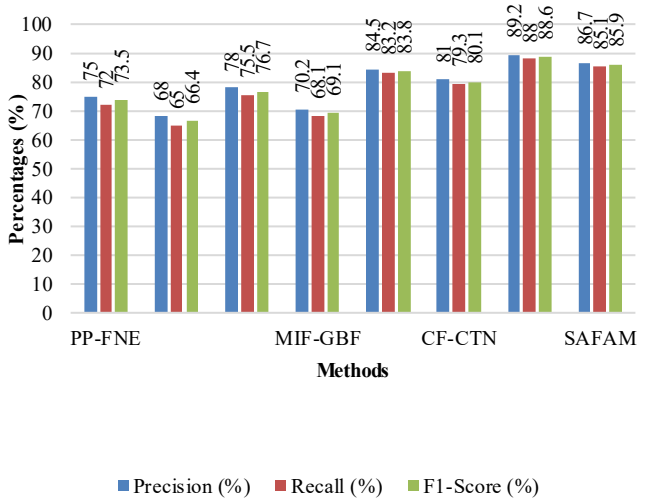


Fig. 9. Precision and recall under imbalanced dataset.

Comparison of computational complexity of proposed methods Estimation time complexity based on features, data points, and model parameters. PP-FNE has higher complexity thanks to ensemble learning, while SAFAM and CF-CTN balance complexity and efficiency. This knowledge aids in model selection and selection based on available compute resources in the result is given Fig. 10.

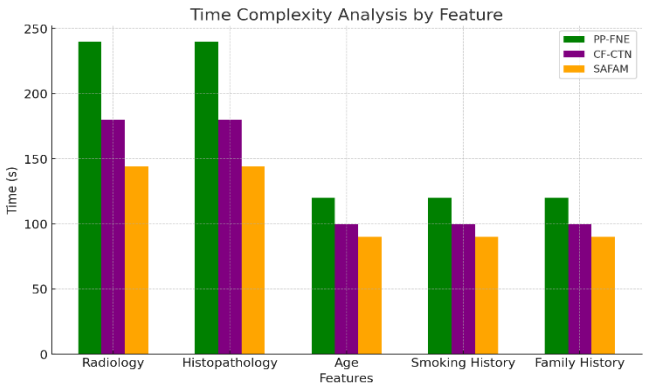


Fig. 10. Time complexity analysis.

The Comparative Analysis discusses the performance results of the proposed methods—Privacy-Preserving Federated Neural Ensemble (PP-FNE), multi-institutional Federated Gradient Boosting Framework (MIF-GBF), Collaborative Federated Convolutional Transformer Network (CF-CTN) and Secure Adaptive Federated Attention Mechanism (SAFAM)—in relation to existing models and to one another. These methods utilize different methods of analysis and show various trade-offs between features of critical performance, i.e. accuracy, efficiency, noise sensitivity and scalability [2]. The analysis has outlined how the characteristics of architectural configuration of each method address the problems related to multi-institutional data heterogeneity, privacy, and how efficient classification of lung-cancer severity can be carried out. Moreover, it translates the trade-offs between the complexity of the models and their performance hence guiding the choice trainable methods specific to particular clinical cases of Table III.

TABLE III. COMPARATIVE ANALYSIS

Algorithm /Work	*Accuracy (%)*	*Precision (%)*	*Recall (%)*	*F1-Score (%)*	*Training Time (s)*	*Noise Robustness(%)*
[1]	85.2	83.5	84.0	83.8	200	70.5
[2]	87.5	85.7	86.3	85.9	180	75.2
[3]	80.6	79.4	80.0	79.7	250	68.0
[4]	88.1	86.8	87.0	86.9	190	76.0
[5]	86.0	84.8	85.1	84.9	210	72.5
[6]	84.3	82.9	83.5	83.2	220	71.0
[7]	83.7	81.4	82.0	81.7	230	69.0
[8]	85.9	84.1	84.5	84.3	215	74.0
[9]	87.1	85.9	86.5	86.2	200	73.5
[10]	86.5	84.7	85.2	85.0	195	72.8
PP-FNE (Proposed)	91.2	90.5	91.0	90.7	140	85.0
MIFGBF (Proposed)	90.8	89.9	90.2	90.0	130	86.0
CF-CTN (Proposed)	92.5	91.8	92.0	91.9	150	87.5
SAFAM (Proposed)	93.4	92.7	93.0	92.8	120	89.0

V. CONCLUSION

This work has proposed a federated learning framework for lung cancer severity classification, which is capable of working efficiently across multi-institutional clinical data silos while respecting patient privacy. The proposed federated learning scheme includes algorithmic advances that focused on ensemble robustness, gradient-based collaboration, multimodal feature integration, and adaptive aggregation, and produced the following: gains in classification accuracy, improved robustness, and enhanced scalability in a privacy-preserving context. Aside from the fact that the performance is quantifiable, this work illustrates increasingly plausible pathways towards decentralized AI successfully being used for medical diagnosis. Nonetheless, there are limitations. The fact that the model relies on high-quality institutional data and collective data-local training protocols may not allow the model to be easily generalized in the real-world. Ongoing challenges of communication latency and synchronization overhead and not accounting for differences in the site-specific model convergence at each research site, persist in federated deployments. Synthetic datasets, while designed to reflect heterogeneity were generated with a common simulated heterogeneity, which may not fully capture variability in biological pathology or the distribution of noise in the real-world. Despite its demonstrated collaborative and privacy-preserving benefits, the proposed framework has certain limitations. Secure aggregation does not fully eliminate potential gradient-based inference risks under stronger adversarial assumptions, and robustness against malicious client poisoning is not explicitly incorporated. Additionally, large-scale multi-institutional deployments may introduce communication and synchronization overhead that warrants further optimization in future work. Future work will focus on supporting our framework using real-world multi-center datasets obtained under heterogeneous acquisition conditions. Exploratory are planned on enhancements in secure hardware

enclaves, adaptive communication compression, and trust-aware federated aggregation mechanisms to advance the clinical applicability of the framework. Awareness of the limitations and challenges facing the ADLA and federated learning research stream is important to facilitate promising processes leading towards reliable, scalable, and ethical AI-powered oncology solutions.

REFERENCES

[1] P. Shyamala Bharathi and C. Shalini, "Advanced hybrid attention-based deep learning network with heuristic algorithm for adaptive CT and PET image fusion in lung cancer detection," Medical Engineering and Physics, vol. 126, 2024, doi: 10.1016/j.medengphy.2024.104138.

[2] N. Gautam, A. Basu, and R. Sarkar, "Lung cancer detection from thoracic CT scans using an ensemble of deep learning models," Neural Computing and Applications, vol. 36, no. 5, 2024, doi: 10.1007/s00521-023-09130-7.

[3] L. S. Matza et al., "Health State Utilities Associated with False-Positive Cancer Screening Results," PharmacoEconomics - Open, vol. 8, no. 2, 2024, doi: 10.1007/s41669-023-00443-w.

[4] J. Subash and S. Kalaivani, "Dual-stage classification for lung cancer detection and staging using hybrid deep learning techniques," Neural Computing and Applications, vol. 36, no. 14, 2024, doi: 10.1007/s00521-024-09425-3.

[5] M. Budati and R. Karumuri, "An intelligent lung nodule segmentation framework for early detection of lung cancer using an optimized deep neural system," Multimedia Tools and Applications, vol. 83, no. 12, 2024, doi: 10.1007/s11042-023-17791-8.

[6] D. Endalie and W. T. Abebe, "Analysis of lung cancer risk factors from medical records in Ethiopia using machine learning," PLOS Digital Health, vol. 2, no. 7, 2023, doi: 10.1371/journal.pdig.0000308.

[7] S. Tummala, S. Kadry, A. Nadeem, H. T. Rauf, and N. Gul, "An Explainable Classification Method Based on Complex Scaling in Histopathology Images for Lung and Colon Cancer," Diagnostics, vol. 13, no. 9, 2023, doi: 10.3390/diagnostics13091594.

[8] A. Choudhury, S. Balasubramaniam, A. P. Kumar, and S. N. P. Kumar, "PSSO: Political Squirrel Search Optimizer-Driven Deep Learning for Severity Level Detection and Classification of Lung Cancer," International Journal of Information Technology and Decision Making, 2023, doi: 10.1142/S0219622023500189.

[9] Karimullah S, Khan M, Shaik F, Alabduallah B, Almjally A. An integrated method for detecting lung cancer via CT scanning via optimization, deep learning, and IoT data transmission. Front Oncol. 2024 Oct 7doi: 10.3389/fonc.2024.1435041. PMID: 39435294; PMCID: PMC11491319.

[10] Srividya, C., RamaSubramanian, K., Madhu Bala, M. (2025). Harnessing Collective Intelligence Without Data Sharing: A Federated Random Forest Approach for Lung Tumor Prediction. In: Mishra, D., Yang, XS.,Unal, A., Jat, D.S. (eds) Data Science and Big Data Analytics. IDBA 2024. Learning and Analytics in Intelligent Systems, vol 43. Springer, Singapore. https://doi.org/10.1007/978-981-97-9855-1_42

[11] B. Vohra and S. Mittal, "Deep Learning Paradigms for Existing and Imminent Lung Diseases Detection: A Review," Journal of Experimental Biology and Agricultural Sciences, vol. 11, no. 2, 2023, doi: 10.18006/2023.11(2).226.235.

[12] Vemula, S.T., Sreevani, M., Rajarajeswari, P., Bhargavi, K., Tavares, J.M.R.S. and Alankritha, S. 2024. Deep Learning Techniques for Lung Cancer Recognition. Engineering, Technology & Applied Science Research. 14, 4 (Aug. 2024), 14916–14922. DOI:https://doi.org/10.48084/etasr.7510.

[13] K. S. Pradhan, P. Chawla, and R. Tiwari, "HRDEL: High ranking deep ensemble learning-based lung cancer diagnosis model," Expert Systems with Applications, vol. 213, 2023, doi: 10.1016/j.eswa.2022.118956.

[14] Kumar, A., Shaik, F., Abdul Rahim, B., Sravan Kumar, D.: Signal and Image Processing in Medical Applications. Springer (2016)

[15] Srividya, C., Ramasubramanian, K., Madhu Bala, M. (2025). Federated Learning-Based Diagnostic Approach for Lung Diseases Detection on CT Images. In: Raju, K.S., Senkerik, R., Kumar, T.K., Sellathurai, M., Naresh Kumar, V. (eds) Intelligent Computing and Communication. ICICC 2024. Lecture Notes in Networks and Systems, vol 1241. Springer, Singapore. https://doi.org/10.1007/978-981-96-1267-3_44

[16] H. Ishimoto et al., "Drug-induced interstitial lung disease caused by olaparib: three case reports and review of the Japanese Adverse Drug Event Report database and literature," BMC Pulmonary Medicine, vol. 23, no. 1, 2023, doi: 10.1186/s12890-023-02569-3.

[17] K. Ananthajothi, P. Rajasekar, and M. Amanullah, "Enhanced U-Net-based segmentation and heuristically improved deep neural network for pulmonary emphysema diagnosis," Sadhana - Academy Proceedings in Engineering Sciences, vol. 48, no. 1, 2023, doi: 10.1007/s12046-023-02092-5.

[18] A. Heidari, D. Javaheri, S. Toumaj, N. J. Navimipour, M. Rezaei, and M. Unal, "A new lung cancer detection method based on the chest CT images using Federated Learning and blockchain systems," Artificial Intelligence in Medicine, vol. 141, 2023, doi: 10.1016/j.artmed.2023.102572.

[19] L. R. Soares, M. Vilbert, V. D. L. Rosa, J. L. Oliveira, M. M. Deus, and R. Freitas-Junior, "Incidence of interstitial lung disease and cardiotoxicity with trastuzumabderuxtecan in breast cancer patients: a systematic review and single-arm meta-analysis," ESMO Open, vol. 8, no. 4, 2023, doi: 10.1016/j.esmoop.2023.101613.

[20] M. D. Lydia and M. Prakash, "An automatic lung nodule detection and classification using an optimized convolutional neural network and enhanced k-means clustering," Journal of Ambient Intelligence and Humanized Computing, vol. 14, no. 12, 2023, doi: 10.1007/s12652-023-04711-9.

[21] SayaSudhakarBabu, G Ashwini, and Dr. T. Ramashri, "Automatic Benign and Malignant Classification of Pulmonary Nodules in Thoracic Computed Tomography Images based on CNN Algorithm," International Journal of Scientific Research in Science and Technology, 2023, doi: 10.32628/ijsrst52310340.

[22] A. Selvapandian, S. NagendraPrabhu, P. Sivakumar, and D. B. Jagannadha Rao, "Lung Cancer Detection and Severity Level Classification Using Sine Cosine Sail Fish Optimization Based Generative Adversarial Network with CT Images," Computer Journal, vol. 65, no. 6, 2022, doi: 10.1093/comjnl/bxab141.

[23] R. J. Hewitt et al., "Lung cancer screening provides an opportunity for early diagnosis and treatment of interstitial lung disease," Thorax, vol. 77, no. 11, 2022, doi: 10.1136/thorax-2022-219068.

[24] C. Gu, C. Dai, X. Shi, Z. Wu, and C. Chen, "A cloud-based deep learning model in heterogeneous data integration system for lung cancer detection in medical industry 4.0," Journal of Industrial Infomation Integration, vol. 30, 2022, doi: 10.1016/j.jii.2022.100386.

[25] S. V. Zinovev, "104P The potential for lung cancer detection in COVID-19 CT scans with AI technologies usage," Annals of Oncology, vol. 33, 2022, doi: 10.1016/j.annonc.2022.02.130.

[26] S. U. Umu, H. Langseth, V. Zuber, Å. Helland, R. Lyle, and T. B. Rounge, "Serum RNAs can predict lung cancer up to 10 years prior to diagnosis," eLife, vol. 11, 2022, doi: 10.7554/ELIFE.71035.

[27] S. Aso et al., "Exploring the educational needs for severe immune-related adverse events of PD-1/PD-L1 inhibitors in advanced lung cancer: A single-center observational study," Asia-Pacific Journal of Oncology Nursing, vol. 9, no. 8, 2022, doi: 10.1016/j.apjon.2022.100076.

[28] M. Zhang, Y. Zhang, Y. Han, X. Zhao, and Y. Sun, "Lung microbiota features of stage III and IV non-small cell lung cancer patients without lung infection," Translational Cancer Research, vol. 11, no. 2, 2022, doi: 10.21037/tcr-22-92.

[29] P. Sinthia, M. Malathi, K. Anitha, and M. Suresh Anand, "Improving lung cancer detection using faster region-based convolutional neural network aided with fuzzy butterfly optimization algorithm," Concurrency and Computation: Practice and Experience, vol. 34, no. 27, 2022, doi: 10.1002/cpe.7251.

[30] E. Iswardy, M. Munzir, and E. Yufita, "Improvements to conventional methods for determining lung cancer areas from CT scan images using ImageJ - software," Journal of Physics: Theories and Applications, vol. 6, no. 2, 2022, doi: 10.20961/jphystheor-appl.v6i2.59349.

[31] M. Shanid and A. Anitha, "Adaptive optimisation driven deep belief networks for lung cancer detection and severity level classification," International Journal of Bio-Inspired Computation, vol. 18, no. 2, 2021, doi: 10.1504/IJBIC.2021.118101.

[32] M. al Achkar, M. ZigmanSuchsland, F. M. Walter, R. D. Neal, B. H. L. Goulart, and M. J. Thompson, "Experiences along the diagnostic pathway for patients with advanced lung cancer in the USA: A qualitative study," BMJ Open, vol. 11, no. 4, 2021, doi: 10.1136/bmjopen-2020-045056.

[33] C. Guo, X. Huang, and G. Lei, "Hospital anxiety and depression scale exhibits good consistency but shorter assessment time than Zung self-rating anxiety/depression scale for evaluating anxiety/depression in non-small cell lung cancer," Medicine (United States), vol. 100, no. 8, 2021, doi: 10.1097/MD.0000000000024428.

[34] N. Purandare, A. Prakash, S. Shah, A. Agrawal, A. Puranik, and V. Rangarajan, "Incidentally detected COVID-19 lung changes during oncologic fluorodeoxyglucose positron emission tomography-computerized tomography studies: Experience from tertiary care cancer hospital," Indian Journal of Nuclear Medicine, vol. 36, no. 4, 2021, doi: 10.4103/ijnm.ijnm_94_21.

[35] A. Anitha and M. Shanid, "Adaptive optimisation driven deep belief networks for lung cancer detection and severity level classification," International Journal of Bio-Inspired Computation, vol. 18, no. 2, 2021, doi: 10.1504/ijbic.2021.10041425.

[36] Dataset: https://www.kaggle.com/datasets/shreyasparaj1/lung-cancer-dataset.

RiskMIS: A Web-Based Risk Management Information System

Alya Thiab Almutairi, Kassem Saleh
Department of Information Science-College of Life Sciences, Kuwait University,
Sabah Al Salem University City, P.O. Box 5969, Safat 13060, Shadadiya, Kuwait

***Abstract*—Risk management is the systematic process of identifying, assessing, monitoring, and responding to risks in projects and ongoing operations. Effective execution of risk management activities is essential for the successful completion of projects and the achievement of key performance indicators in operational environments. In recent years, organizations have increasingly emphasized the proactive identification and mitigation of risks before they materialize. Consequently, risk assessment is addressed early in the lifecycle and continuously revisited to ensure the proper functioning of ongoing operations and the successful delivery of projects. A central tool in the risk management process is the risk register, which consolidates all critical and relevant information related to identified risks. The risk register serves as a focal point around which risk management activities are organized. It is inherently dynamic and must be continuously updated to reflect changes in risk exposure and response strategies. When properly managed, the risk register supports informed decision-making and enables managers to handle risks in a systematic and timely manner. The primary objective of this study is to bridge theory and practice by developing a risk management information system centered on a comprehensive risk register. The study presents the design of the proposed system using established modeling techniques and diagrams, followed by the development of a functional prototype based on modern web technologies. Finally, a demonstration of the prototype is provided to illustrate its capabilities and practical applicability.**

Keywords*—Risk management; risk management information system; risk register; project risk; decision support systems

I. Introduction

Risks are inherent in the management of projects and ongoing operations. Managers are therefore expected to be capable of identifying and anticipating risks that may affect organizational activities. In practice, however, risks are often neglected or inadequately managed, either because they are perceived as unlikely to occur or because risk management is viewed as an unnecessary overhead. Some managers also tend to associate risk solely with negative outcomes to be avoided. In reality, risk is an integral part of organizational performance and value delivery, making effective risk management essential for business success. In today's rapidly changing environment, organizations face increasing levels of uncertainty, and their resilience and sustainability depend largely on how well risks are managed.

The importance of risk management is particularly evident in information systems projects. The Standish Group's CHAOS report [35] indicates that approximately 69% of such projects partially or completely fail. Empirical evidence shows that integrating risk management into the project development process significantly improves the likelihood of successful project execution and deployment. The same principle applies to ongoing business operations, where unmanaged risks can undermine performance and continuity.

A wide range of standards, guidelines, and best practices has been proposed to support risk management in general and project risk management in particular. Notable examples include the ISO 31000 risk management guidelines [16] and the Project Risk Management framework of the Project Management Body of Knowledge (PMBOK) published by the Project Management Institute [28]. Studies indicate that 71% of projects that systematically apply risk management practices succeed in meeting customer expectations with respect to time, quality, and cost [6].

As part of a preliminary investigation, 28 leading RMIS and risk register tools were systematically analyzed through provider websites, email questionnaires, live chats, and limited demo access. The comparative analysis revealed several unmet limitations in existing solutions. Most tools primarily target medium and large enterprises, with limited accessibility for small organizations or individual users. Additionally, many systems are complex, expensive, require extensive configuration, and often integrate risk management as a secondary function within broader project management platforms rather than offering a focused and streamlined risk register solution.

Furthermore, access restrictions, quote-based pricing models, installation requirements, and limited web or mobile availability reduce usability and flexibility. These findings highlight a practical and conceptual gap in the availability of a lightweight, web-based, accessible, and cost-effective RMIS designed specifically to simplify risk management processes while maintaining usability, scalability, and security. Therefore, the proposed RiskMIS addresses this identified gap by offering a simplified, web-based, and broadly accessible solution adaptable across various domains without heavy customization requirements.

Among the tools used to support risk management, the risk register plays a central role. It provides a structured repository for documenting identified risks, their characteristics, and planned responses, and it serves as the focal point for risk analysis, prioritization, and monitoring. Despite its importance, existing research has largely focused on theoretical aspects of risk management—such as risk categorization, probability,

impact, and prioritization—while paying limited attention to the design and structure of the risk register itself. Some studies propose domain-specific risk registers or spreadsheet-based approaches. For example, [37] suggests a reference risk register for project management and outlines steps for its implementation using Excel, while Burcar, Radujković, and Vukomanović [5] propose a methodology for developing a risk register in construction projects using software support such as relational databases. However, these studies provide limited guidance on how to systematically design and implement a comprehensive risk register or how to embed it within an integrated risk management information system. Moreover, existing solutions are often domain-specific and lack adaptability across different organizational contexts.

This study addresses these limitations by presenting the systematic development and implementation of RiskMIS, a comprehensive risk management information system designed around a dynamic and reusable risk register. The system supports the structured documentation of risks and response strategies to enhance the success of projects and ongoing operations. To develop the proposed tool, we adopted a Software Development Life Cycle (SDLC) approach based on the waterfall model, encompassing analysis, design, implementation, testing, and maintenance phases [34].

The study begins with an extensive analysis of existing approaches and relevant literature to consolidate risk register concepts and identify suitable theories related to risk categorization, prioritization, probability, and impact. It then specifies the functional and non-functional requirements of the proposed system, followed by the detailed design and implementation of a working prototype. The development process includes algorithm design, database conceptual modeling, system architecture, activity and state diagrams, graphical user interface design, and data structure definition. The database was designed using the Entity Relationship (ER) model, while Unified Modeling Language (UML 2.5) diagrams were used to represent behavioral and structural aspects of the system [25]. The prototype was implemented as a web-based application using the ASP.NET Core framework, developed in C#, with Microsoft SQL Server 2018 used for data management. The system was subsequently tested to verify that it meets its specified requirements and intended functionality.

The remainder of this study is organized as follows: Section II presents background concepts related to risk and risk management. Section III defines and analyzes the functional and non-functional requirements of the proposed system. Section IV describes the design and implementation of RiskMIS. Section V demonstrates the system prototype. Finally, Section VI discusses the findings, draws conclusions, and outlines directions for future work.

Despite the structured design of RiskMIS, several limitations remain. The system has not yet been validated through longitudinal case studies in real organizational environments, and quantitative performance metrics have not been formally established. Technical benchmarking under high operational loads has not been conducted, limiting conclusions regarding scalability and database efficiency. Additionally, formal security assessments, version control mechanisms, and comprehensive data lifecycle management strategies have not yet been fully implemented. The system has not been empirically validated across multiple sectors, and structured user training and adoption frameworks have not been systematically evaluated. These limitations highlight areas for further empirical and technical development.

II. Background

This section provides background information on risks and risk management, including risk terminology, risk classification, operational and project risk management, risk management frameworks and processes, risk registers, and risk assessment criteria.

A. Risk Terminology

Risk is commonly defined as the probability of an event occurring and the consequences associated with that event [14]. It is typically characterized by three fundamental elements: the event itself, its probability of occurrence, and its impact [29]. In general terms, risk refers to any event or condition that may hinder an organization from achieving its objectives. This study focuses on negative risks, hereafter referred to simply as risks. Effective risk management aims to minimize adverse impacts while enabling organizations to sustain performance and achieve desired outcomes.

Organizations classify risks based on their nature, context, and specific concerns [26]. Although no universal risk classification system applies to all organizations, many adopt classifications proposed by relevant standards, frameworks, or prior research [13]. The terms *risk classification*, *risk categorization*, *risk typing*, and *risk taxonomy* are often used interchangeably to describe the grouping of risks according to defined criteria.

Risks may be categorized by source, type, timescale, or nature of impact. For example, risks can affect financial performance, operational activities, infrastructure, reputation, or market position. A structured risk classification system facilitates risk identification, responsibility assignment, comparison of similar risks, and the definition of risk management scope. Consequently, organizations often tailor their classification schemes to align with their specific activities and objectives [13]. For instance, Larson and Gray [19] propose a generic Risk Breakdown Structure (RBS) based on risk sources, while a similar RBS for generic projects is presented in [12].

B. Operational and Project Risk Management

Organizations face multiple types of risks, and categorization plays a key role in their effective identification [11]. Common categories include operational, project, organizational, and commercial risks. Operational risks arise from routine organizational activities and typically result from human error, process failures, or deficiencies in systems and controls [9]. Such risks often affect service quality, efficiency, and reliability [13].

Project risk management, in contrast, focuses on uncertainties that may affect project objectives throughout the project lifecycle. It encompasses planning for risk management, risk identification, qualitative and quantitative analysis,

response planning, and ongoing monitoring and control [28]. The objective is to minimize negative risks while maximizing positive risks, or opportunities. Effective project risk management is widely recognized as a critical factor in successful project delivery [30].

Projects progress through a defined lifecycle from initiation to closure [28]. Numerous lifecycle models exist, often tailored to specific industries or organizational contexts. Organizations typically select lifecycle models that best align with their operational needs and governance structures.

C. Risk Management Processes and Frameworks

Effective risk management requires well-defined processes supported by appropriate risk management frameworks [30]. According to [15], risk management is the systematic application of policies, procedures, and practices to activities such as communication, consultation, context establishment, risk identification, analysis, evaluation, treatment, monitoring, and review. These processes are formalized within widely adopted standards and frameworks, including ISO 31000, the IRM Standard, BS 31100, and the COSO ERM framework [13].

In the software and information systems domains, several formal risk management methodologies assess risks throughout the development lifecycle [8]. Notable examples include Boehm's risk management model [3], the SEI Continuous Risk Management (SEI-CRM) method [10], the Riskit method [18], the CMMI framework [7], and the Larson and Gray approach [19].

Since its introduction, ISO 31000:2009 has gained widespread adoption due to its comprehensive and flexible structure [2]. Nevertheless, many organizations prefer the PMBOK framework when managing project-specific risks.

Larson and Gray [19] describe a four-step risk management process comprising risk identification, risk assessment, risk response development, and risk response control. The final step include implementing response strategies, monitoring risk triggers, activating contingency plans, updating risk records, and identifying emerging risks. Empirical evidence supports the effectiveness of structured risk management processes. For example, Junior et al. [17] found that applying formal risk management practices and assigning dedicated risk roles significantly improved project success. While project managers remain accountable for overall project outcomes [19], risk managers are responsible for planning, coordinating, and overseeing risk-related activities [4].

D. Risk Register

The risk register is a fundamental component of risk management at both project and organizational levels [13]. Its importance is consistently emphasized in the literature; for instance, Saffin et al. [33] report that 70% of project managers consider the risk register a critical project management tool. A risk register consolidates detailed information about identified risks, including their likelihood, impact, priority, ownership, and response strategies. The risk register should be highly accessible and provide the necessary information about risks [32].

Risk registers vary widely in form and complexity, ranging from simple paper-based documents to sophisticated software solutions. Many organizations rely on spreadsheets or word-processing tools for risk documentation. A study by the Design Information Group at Bristol University found that 67% of respondents documented risks using paper or computer-based registers, with 78% developed internally [27]. While such approaches are accessible, they often lack scalability, consistency, and integration.

Several studies have explored risk register implementation in specific domains. For example, Burcar et al. [4] report positive results from implementing a database-driven risk register in the Croatian construction industry. Other research highlights that a risk register is essentially a structured and continuously updated repository of risk information [37], which may be implemented using documents, spreadsheets, or relational databases [13].

Comparative studies further indicate that theoretical models alone are insufficient for developing comprehensive and reusable risk registers [36]. Domain-specific implementations—such as reference risk registers in information security [24] or electric power generation [20]—demonstrate practical benefits but often lack generalizability. According to the PMBOK Guide [28], a risk register should include identifiers, descriptions, owners, causes, impacts, triggers, response strategies, and status information. To remain effective, it must be continuously updated and treated as a dynamic information resource [13].

Despite their widespread use, spreadsheet-based risk registers present significant limitations. Martins [22] identifies issues related to data organization, version control, collaboration, integration, and usability. These limitations are reinforced by survey findings indicating that while 97% of finance professionals use spreadsheets for risk management, 72% consider them inefficient [1]. Spreadsheet-based tools are particularly vulnerable to access control issues, formula errors, and inconsistent data handling, making them unsuitable for comprehensive and long-term risk management.

E. Risk Criteria and Risk Matrix

Risk criteria define the basis on which risks are evaluated and prioritized and vary according to organizational objectives and context [16]. Establishing clear risk criteria ensures a shared understanding of risk-related terms and assessment scales across the organization.

Following risk identification, risks are analyzed by assessing their probability and impact based on the defined criteria. Each risk is assigned a risk score to support prioritization. Risk analysis may be qualitative or quantitative. Chihuri and Pretorius [6] found that qualitative risk analysis is more commonly used due to its simplicity and ease of interpretation. Although cost-effective, qualitative methods are inherently subjective [30]. Quantitative analysis, while more precise, is less frequently applied because of its complexity. In qualitative analysis, risk scores are typically calculated by combining probability and impact values [33], whereas quantitative analysis numerically estimates potential losses and is often reserved for high-exposure risks [30].

III. ANALYSIS

In this section, we identify the most relevant theories related to risk categorization, prioritization, probability, and impact, and use them to derive the functional and non-functional requirements of the proposed tool.

A. Main Risk Management Elements

Risk management can become complex and time-consuming when inefficient tools are used. Manual and semi-structured approaches, such as word processors and spreadsheets, impose significant limitations on consistency, traceability, and timely decision-making. These limitations highlight the need for an automated and unified risk management tool capable of supporting risk identification, analysis, and treatment throughout the project lifecycle, thereby improving the likelihood of project success.

1) Project life cycle: To ensure applicability across different business domains, the proposed tool adopts a generic predictive project life cycle model consisting of five stages, based on the standardized project lifecycle and best practices defined by the Project Management Institute (PMI). In predictive life cycles, project cost, duration, and scope are defined early in the lifecycle [28].

The first stage, Idea, represents a preliminary concept in which project goals, specifications, and team responsibilities are not yet finalized. Projects in this stage can be modified or canceled without organizational impact, preventing unnecessary allocation of resources to initiatives that may never commence. The second stage, Initiating, involves finalizing project objectives, scope, budget, duration, and assigning project and risk managers. In the Planning stage, the project manager develops detailed plans to meet project requirements within approved constraints. Once the plan is confirmed, the project enters the Executing stage, during which tasks are carried out according to the approved plan. The final stage, Closing, concludes the project and delivers outcomes to the client. In this model, execution and monitoring activities defined by PMBOK [28] are consolidated into a single Executing stage.

Within the tool, projects transition through seven operational states, as illustrated in Fig. 8. Projects are initially created in the Idea state and progress to Initiate, once key details are finalized and responsible managers are assigned. Upon plan development, the project moves to the Plan state. When at least one task enters execution, the project transitions to In Process. A project is marked Completed when all tasks are completed. Projects may also enter On Hold or Canceled states. Only administrators may cancel projects, and only when the project is in the On Hold state. Projects canceled in the Idea state are permanently removed, as they provide no long-term value. Project states are automatically updated based on task status, ensuring consistency between task progress and overall project state. The meaning of each state is summarized in Table I.

TABLE I. PROJECT STATUS OF THE DEVELOPED TOOL

Project State	Meaning	Notes
Idea	The administrator has an idea of a new project discussed or requested by the client but doesn't have a final decision about the duration, budget, and the assigned PM and RM for the project.	This is the first and default status of the project. The administrator is the one who decides to have a new project. The administrator can change the project information. The administrator can delete the project from the tool permanently.
Initiate	The administrator finalizes the contract with the client and decides the start and end date of the project, the budget, and the PM and RM who will be responsible for the project.	The administrator should provide a description of the project. The administrator is Not allowed to make changes to the project information. The project will be available to PM and RM to work in. Administrator cannot delete the project after initiation.
Plan	PM divides the project into tasks, each task with a start and end date, cost, and requirements (goals to be achieved in specific tasks). Project being in Plan status when at least one task is created for the project.	The project will be in plan status when PM starts making tasks for the plan. The plan should be within the duration and budget decided by the administrator. Requirements for each task will be used as indicators of the completion of the task goal. If PM confirms the plan the project still in plan status but ready to start, all tasks will be in "Not Started" status. PM can cancel tasks after confirming the plan . If all tasks are canceled the project becomes in canceled status.
In Process	If PM starts at least one task (task in "In Process" status or "Delay" status) the project status becomes “In process” automatically even if there canceled tasks or “On hold”.	Administrator or PM cannot cancel or delete a project in the "In Process" state The project state changes automatically by the tool based on the tasks of the project.
Completed	If all tasks status is "Completed", even if some tasks are in the "Canceled" state	-
On Hold	Tool make the project status "On Hold" automatically when at least one task "On hold" and no task "In Process" or "Delay".	Administrator can delete the project and become in canceled state. This will make the status of "Not Started" tasks and "On Hold" tasks to "Deleted" and keep their cost and scope values to be used in the overall project calculations.
Canceled	Tool make the project state "Canceled" automatically if all tasks status is "Canceled" or when the Administrator cancels the project when it is "On Hold" state	Project with "Canceled" status, their data will be stored in the tool and not deleted, and users can see the report of the project.

2) Task life cycle: During the Executing stage, the approved project plan is implemented through task execution. Tasks progress through three primary steps: Not Started, In Process, and Completed. Once the project plan is confirmed, all tasks default to the Not Started state. Project managers may initiate tasks, moving them to In Process, and continuously monitor their progress. Tasks are marked Completed once all defined requirements are achieved.

In addition to these steps, tasks may assume one of six possible statuses. Tasks that have not started may be canceled and are subsequently treated as non-existent. If an executing task exceeds its planned duration, it is marked as Delayed. Tasks in In Process or Delayed states may be placed On Hold, at which point the project manager records the associated cost and scope. On-hold tasks may later be resumed or completed, with final cost and scope values recorded upon completion.

3) Project success criteria: To ensure general applicability, project success in the proposed tool is evaluated using the triple constraint model—cost, time, and scope. By embedding this triangle within the predictive project life cycle, the tool allows project and risk managers to continuously assess the impact of identified risks on the key project constraints. This approach aligns with the Project Management Institute's PMBOK Guide, which emphasizes the integration of scope, schedule, and budget management across the project lifecycle, and is further reinforced by ISO 31000:2018 principles, which highlight the importance of assessing risk impact on organizational objectives. Consequently, the proposed provides a structured method to manage risks while maintaining balance among the critical project parameters, ensuring both effective risk management and project success.

Each task is assessed based on actual expenditure, execution duration, and degree of requirement fulfillment. Aggregated task-level metrics determine overall project success. Risks directly influence these dimensions, thereby affecting both task performance and overall project outcomes.

4) Risk management process: The proposed tool supports a collaborative risk management process involving administrators, project managers, risk managers, and team members. Upon project initiation, roles are clearly assigned to ensure separation of duties. Although organizations may adopt different risk management models, most follow a common process comprising risk identification, assessment, response, and monitoring. This structure aligns with the principles and risk management process defined in ISO 31000:2018, upon which the design of the proposed tool is conceptually grounded. In the proposed system, risks are identified collaboratively by project and risk managers, with input from team members using techniques such as brainstorming. Identified risks are reported through the system and recorded in the risk register. Risks are then categorized, assessed, and prioritized based on defined criteria. Risk managers prepare response strategies and assign risk owners responsible for implementation under the supervision of the project manager. The risk manager continuously monitors risks, including secondary risks, and updates the register as new risks emerge.

5) Risk register content: Based on the preceding analysis, the risk register in the proposed tool includes the following attributes: risk name, description, owner, probability, impact on cost, time, and scope, detectability, urgency, status, classification, response strategy, and secondary risk information. The structure of this risk register is aligned with the recommendations of the Project Management Institute as outlined in the guide of PMBOK.

6) Risk classification: Given the absence of a universally applicable risk classification system, the tool allows organizations to define and customize their own classifications. A default two-level classification structure is provided to simplify risk identification and reduce ambiguity. This structure consists of main categories and subcategories with no limit on the number of entries. The default classification is based on the Larson and Gray framework, which includes four main categories and corresponding subcategories, as discussed in Section II A.

7) Risk criteria: The tool incorporates a generic risk criteria framework covering probability, impact on cost, time, and scope, detectability, and urgency. Each factor is defined on a five-level scale (1–5) with percentage-based definitions to minimize ambiguity.

The risk probability is categorized, as shown in Table II. The risk detection is categorized, as shown in Table III. Risk urgency is defined using a binary scale distinguishing urgent and non-urgent risks (see Table IV). Task scope completion is also evaluated using a predefined scale (see Table V).

Organizations are required to specify their risk attitude by ranking probability, impact, and detectability according to importance. These rankings influence risk scoring, project status evaluation, and the calculation of risk impact on cost, schedule, scope, and response effectiveness.

TABLE II. RISK PROBABILITY SCALE

Score	Description	Definition
1	Almost Never	The risk is 1% or less expected to occur in the project.
2	Low	The risk is more than 1% expected to occur in the project.
3	Moderate	The risk is 20% and more expected to occur in the project.
4	High	The risk is 50% and more expected to occur in the project.
5	Almost Certain	The risk is 90% and more expected to occur in the project.

TABLE III. RISK DETECTION SCALE

Score	Description	Definition
1	Almost Certain	The probability that the risk will be detected is more than 90% certain.
2	High	The probability that the risk will be detected is more than 50%.
3	Moderate	The probability that the risk will be detected is more than 25%.
4	Low	The probability that the risk will be detected is more than 5%.
5	Almost Impossible	The probability that the risk will be detected is 0%.

TABLE IV. RISK URGENCY SCALE

Score	Description	Definition
1	Urgent Risk	The risk needs an immediate action
0	Unurgent Risk	The risk can wait

TABLE V. TASK SCOPE SCALE

Score	Description	Definition
1	Poor	Less than 60% of the task requirements have been met.
2	Fair	61% - 80% of the task requirements have been met.
3	Good	More than 81% of the task requirements have been met, and some essential requirement not done yet.
4	Very Good	All essential requirements of the task have been done, but there are some remaining low priority requirements not done.
5	Excellent	All the requirements of the task have been met exactly as mentioned in the plan.

8) Risk matrix: The proposed tool employs qualitative risk analysis to prioritize risks. Precise definitions of probability and impact ensure consistent interpretation across the organization. Risk scores are derived by combining probability and impact values, using either the highest or average impact when multiple dimensions are affected.

For example, consider the risk "The technology is new and the project team lacks sufficient experience, leading to delays". If the likelihood of occurrence is between 1% and 20%, the probability is classified as Low (score = 2). Based on the defined criteria, the impact may be assessed as Moderate for cost (score = 3) and Catastrophic for both time and scope (score = 5). The resulting risk score is then calculated using the predefined qualitative formulas, supporting systematic prioritization and treatment.

$$Risk\ Score = p * \max_{1 \le c,t,s \le 5}(c,t,s)$$

where, p is the probability of occurrence, c is the impact in cost, t is the impact in time, and s is the impact in scope. By applying this equation, the risk score of the risk will be computed as follows:

$$Risk\ Score = 2 * \max_{1 \le c,t,s \le 5}(3,5,5)$$

$$Risk\ Score = 2 * 5 = 10$$

The risk score using the average impact is expressed using the following formula:

$$Risk\ Score = p * \frac{c+t+s}{3}$$

where, p is the probability of occurrence, c is the impact in cost, t is the impact in time, and s is the impact in scope. By applying this equation, the risk score will be computed as follows:

$$Risk\ Score = 2 * \frac{3+5+5}{3}$$

$$Risk\ Score = 2 * \frac{13}{3} = 8.67$$

The risk score based on the average impact reduces the priority of risk compared to the highest impact score. So, the risk score calculation can have a critical impact on how the risk is evaluated and dealt with. In [31], the authors proposed a new formula to calculate the risk score by adding a detection score to the well-known risk analysis method and they found that it improves risk prioritization. The Risk Priority Number (RPN) is expressed using the following formula:

$$Risk\ Score = p * i * d$$

where, p is the probability of occurrence, i is the impact of risk, and d is the detectability score. Detectability is the capacity of the organization to detect the risk before it happens [23]. The importance of detectability is important even if the risk is having low probability and low impact. If not detected, the risk can cause a big loss to the project [31]. In our example, if the RM uses techniques and tools to easily detect the risk that the project team is unfamiliar with the new technology, the detectability of risk is "High" with a score of (2) based on Table III.

RPN gives the same weight for the three components of the risk score. That may result in inaccurate risk assessment [21]. It neglects the importance of each component for the organizations and assumes that each organization considers probability, impact, and detectability with the same importance. In [21], the authors proposed a new method to solve the issue of the weights' difference of risk components by generating measures based on the weights of the probability, impact, and detectability taken from a domain expert.

The developed tool prioritizes risks in the new proposed methodology based on the previously discussed measures. The prioritization process has six steps. First, the organization should specify its risk criteria which should include in addition to the references of the risk probability, impact in cost, time, and scope, and detectability, and score, the rank of the risk probability, impact, and detectability from the most important to least. The importance rank indicates the weight of each component, the first component in the rank indicates that it's

the most important component factor to the organization and needs more attention, so it will take the weight of (3). On the other hand, the second and third components in the organization ranks of importance will have the same weight of (1), which means that those components are less important to the organization compared to the first one. The organization rank of importance can be set by the Risk Manager in the developed tool using "Risk Settings" page.

Secondly, after determining the weights, the RM should specify the risk scores of the probability, impact on cost, time, and scope, and detectability for risks. The third step is performed by the tool using the following formula to calculate the risk score for each risk:

$$Risk\ Score = \begin{cases} \left(p * g(p)\right) + \left(\frac{c+t+s}{3}\right) + d, if\ g(p) > g(i)\ and\ g(p) > g(d) \\ p + (\max(c,t,s) * g(i)) + d, if\ g(i) > g(p)\ and\ g(i) > g(d) \\ p + \left(\frac{c+t+s}{3}\right) + \left(d * g(d)\right), if\ g(d) > g(p)\ and\ g(d) > g(i) \end{cases}$$

where, p is the probability of occurrence, i is the impact of risk, d is the detectability score, g(p) is the weight of probability, g(i) is the weight of impact, g(d) is the weight of detectability, c is the impact on cost score, t is the impact on time score, s is the impact on scope score. By applying this equation based in the previous number of the provided example, and by assuming that the most important risk component to the organization is the impact which gives impact component weight of (3) and both the probability and detectability weight of (1), the risk score will be computed as follows:

$$Risk\ Score = \begin{cases} \left(p * g(p)\right) + \left(\frac{c+t+s}{3}\right) + d, if\ g(p) > g(i)\ and\ g(p) > g(d) \\ p + \left(\max(c,t,s) * g(i)\right) + d, if\ g(i) > g(p)\ and\ g(i) > g(d) \\ p + \left(\frac{c+t+s}{3}\right) + \left(d * g(d)\right), if\ g(d) > g(p)\ and\ g(d) > g(i) \end{cases}$$

The second part of the formula will be used to calculate the risk score, since the weight of impact is greater than both the weight of portability and detectability.

$$Risk\ Score = p + (\max(c,t,s) * g(i)) + d$$

$$Risk\ Score = 2 + (\max(3,5,5) * 3) + 2$$

$$Risk\ Score = 2 + (5 * 3) + 2 = 19$$

In the fourth step, we convert the resulting risk score from a range between 5 and 25 to be in a range from 1 to 5 using the following formula:

$$Risk\ Score_{1-5} = \left((Risk\ Score_{5-25} - 5) * \left(\frac{1}{5}\right)\right) + 1$$

By applying the risk score result from the previous equation, the final result will be computed as follows:

$$Risk\ Score_{1-5} = \left((19 - 5) * \left(\frac{1}{5}\right)\right) + 1$$

$$Risk\ Score_{1-5} = \left(14 * \left(\frac{1}{5}\right)\right) + 1 = 3.8$$

In the fifth step, we add another factor or component that can affect the prioritization of risks. We add an Urgency scale that can also be determined by the RM for each risk. Risk Urgency is determining which risk in the list of risks should be dealt with first before others [29]. Organizations should define risk urgency in their risk criteria. In the developed tool, we have an urgency scale from 0 to 1: 1 means that the risk needs immediate action, while 0 means the risk can wait. 0 is the default urgency status for all risks. The developed tool deals will urgent risk based on the value of the risk score. The following formula shows how the tool will deal with risks that are specified as urgent by the RM:

$$Risk\ Score_{1-5} = \begin{cases} 5, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 4 \\ Risk\ Score_{1-5} + 0.75, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 3 \\ Risk\ Score_{1-5} + 0.50, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 2 \\ Risk\ Score_{1-5} + 0.25, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 1 \\ Risk\ Score_{1-5}, if\ u = 0 \end{cases}$$

where, u is the urgency score. So, to continue with the previous example, if the risk is urgent, the final risk score of the risk will be calculated by the system by adding 0.75 to the risk score because the urgency is 1 and the risk score is greater than 3 as shown in the following evaluation of the formula:

$$Risk\ Score_{1-5} = \begin{cases} 5, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 4 \\ Risk\ Score_{1-5} + 0.75, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 3 \\ Risk\ Score_{1-5} + 0.50, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 2 \\ Risk\ Score_{1-5} + 0.25, if\ u = 1\ and\ Risk\ Score_{1-5} \geq 1 \\ Risk\ Score_{1-5}, if\ u = 0 \end{cases}$$

$$Risk\ Score_{1-5} = Risk\ Score_{1-5} + 0.75$$

$$Risk\ Score_{1-5} = 3.8 + 0.75 = 4.55$$

The final step is performed by the system by providing the user with a risk score definition, as shown in Table VI. In our example, the risk is considered "Serious".

TABLE VI. RISK SCORE SCALE

Score	Description	Definition
1	Negligible	Risk Score is less than 2.
2	Minor	Risk Score is greater or equal to 2.
3	Moderate	Risk Score is greater or equal to 3.
4	Serious	Risk Score is greater or equal to 4.
5	Critical	Risk Score is 5.

9) Risk impact: In the developed tool, any risk added to the project will affect the cost, time, and scope of the project according to the definitions mentioned in Table VII.

TABLE VII. RISK IMPACT SCALE IN COST, TIME, AND SCOPE

Score	Description	Definition		
		Risk Impact on Budget	***Risk Impact on Schedule***	***Risk Impacton Scope***
1	Trivial	The risk has an impact on the budget of the project by less than 5%.	The risk has an impact in the project schedule by less than 5%.	The risk has barely a noticeable impact on the scope of the project.

2	Low	The risk has an impact on the budget of the project by less than 10%.	The risk has an impact on the time of the project by less than 10%.	The risk has an impact on the scope of the project.
3	Moderate	The risk has an impact on the budget of the project by less than 25%.	The risk has an impact on the time of the project by less than 25%.	The risk has an impact on the scope of the project by more than 50%.
4	High	The risk has an impact on the budget of the project by less than 50%.	The risk has an impact on the time of the project by less than 50%.	The risk has an impact on the essential requirements of the project.
5	Catastrophic	The risk has an impact on the budget of the project by greater than 50%.	The risk has an impact on the time of the project by greater than 50%.	The risk has an impact on the project to end in unacceptable project by the client.

For example, if a serious risk which has a high impact on the cost of the project (r = 4, as mentioned in Table VII) is added to a project which has a budget of 1000 USD, the actual cost is 300 USD, and the needed money to finish the remaining tasks is 600 USD, the following formula will be used:

$$Risk\ Impact\ Cost_{\text{USD}} = \begin{cases} n * 0.05, & if\ r = 1 \\ n * 0.10, & if\ r = 2 \\ n * 0.25, & if\ r = 3 \\ n * 0.50, & if\ r = 4 \\ n * 1.00, & if\ r = 5 \end{cases}$$

where, r is the risk impact on cost score, and n is the needed money to finish remaining tasks. By implementing the formula in the example, the result will be as follows:

$$Risk\ Impact\ Cost_{\text{USD}} = \begin{cases} n * 0.05, & if\ r = 1 \\ n * 0.10, & if\ r = 2 \\ n * 0.25, & if\ r = 3 \\ n * 0.50, & if\ r = 4 \\ n * 1.00, & if\ r = 5 \end{cases}$$

$$Risk\ Impact\ Cost_{\text{USD}} = n * 0.50$$

$$Risk\ Impact\ Cost_{\text{USD}} = 600 * 0.50 = 300\ KD$$

So, the risk may incur a loss of 300 USD, which will be added to the actual project cost and the needed money to finish the remaining tasks. The result is 1200 USD, which makes the project overbudget. In the same manner, the risk impact on time will be calculated. If the same risk has moderate impact on the time of the project (r = 3, as mentioned in Table VII), the tool will find the needed days to finish the remaining tasks, let's say in our example it is 30 days, then the same formula will be used where n is the needed days to finish the remaining tasks, finally, we compute the ceiling of the result to find the result in days.

$$Risk\ Impact\ Time_{\text{Day}} = \begin{cases} \lceil n * 0.05 \rceil, & if\ r = 1 \\ \lceil n * 0.10 \rceil, & if\ r = 2 \\ \lceil n * 0.25 \rceil, & if\ r = 3 \\ \lceil n * 0.50 \rceil, & if\ r = 4 \\ \lceil n * 1.00 \rceil, & if\ r = 5 \end{cases}$$

$$Risk\ Impact\ Time_{\text{Day}} = \lceil n * 0.25 \rceil$$

$$Risk\ Impact\ Time_{\text{Day}} = \lceil 30 * 0.25 \rceil = \lceil 7.50 \rceil = 8\ \text{days}$$

The risk impact on scope will be calculated in a different manner. For example, if the risk has a low impact on the scope of the project (from Table VII the r value is 2), and the current scope of the project is very good (the n value is 4 based on Table V). The scope will be calculated as follows:

$$Risk\ Impact\ Scope_{1-5} = \begin{cases} 1, & if\ n < \text{r} \\ (\text{n} - \text{r}) + 1, & if\ n \geq \text{r} \end{cases}$$

where, r is the risk impact on the scope of the project, and n is the actual scope of the project. So, by implementing the formula, the scope of the project will be as follows:

$$Risk\ Impact\ Scope_{1-5} = \begin{cases} 1, & if\ n < \text{r} \\ (\text{n} - \text{r}) + 1, & if\ n \geq \text{r} \end{cases}$$

$$Risk\ Impact\ Scope_{1-5} = (\text{n} - \text{r}) + 1$$

$$Risk\ Impact\ Scope_{1-5} = (4 - 2) + 1$$

$$Risk\ Impact\ Scope_{1-5} = 3\ (Good\ based\ in\ Table\ \text{V})$$

Finally, the analyzed serious risk is estimated to increase project cost by USD 300, delay project delivery by approximately eight days, and reduce the project scope to a *Good* level. When multiple risks exist within a project, each risk independently affects the project cost and schedule. However, the impact on project scope is determined by identifying the risk with the highest scope impact score and applying the Risk Impact on Scope formula accordingly.

10)Risk response strategy: During the risk response phase, the Risk Manager (RM) defines the actions required to treat each identified risk. Risk response strategies are generally categorized as responses to either negative risks (threats) or positive risks (opportunities) [29]. The proposed tool focuses exclusively on negative risks and supports four standard response strategies: acceptance, mitigation, transference, and elimination [26].

Acceptance involves acknowledging the risk without taking proactive action, whereas elimination seeks to remove the risk entirely by addressing its root cause. Mitigation aims to reduce the likelihood and/or impact of the risk, while transference shifts responsibility for managing the risk and its consequences to a third party [29]. In the proposed tool, acceptance retains the risk's impact within the project, while elimination removes its effects. Mitigation and transference strategies influence project performance based on predefined impact scales specified by the project manager. Any selected response strategy is documented and made available to the project manager and risk owner for implementation. Additionally, response strategies may generate secondary risks, as discussed in the following subsection.

11)Secondary risks: The execution of a risk response strategy may introduce new risks, commonly referred to as secondary risks [4]. The proposed tool explicitly supports the identification and management of such risks by allowing risk managers to link secondary risks to their originating response strategies. Secondary risks are initially assigned a Pending status and transition to Open once the associated response strategy is implemented.

For each secondary risk, the risk manager specifies a description, risk owner, impact attributes, and an appropriate response strategy. For example, consider the risk "The technology is new, and the project team lacks sufficient expertise, leading to delays". If the RM selects an elimination strategy—such as requiring all team members to attend training on the new technology prior to project start—this action may introduce a secondary risk. Specifically, team members attending training may delay their assigned tasks. This secondary risk, "Employees are assigned to mandatory training while project tasks remain incomplete", is initially marked as Pending. Once the primary response strategy is executed, the secondary risk becomes Open, and a corresponding response plan is defined to manage its impact.

12)Project health: Project health is evaluated using three indicators: cost health, time health, and scope health.

- Cost Health: Cost performance is classified as *Within Budget* (green), *Worry* (yellow), or *Critical* (red). The *Within Budget* status indicates that project spending has not entered the cost danger zone, defined as

$CostDangerZone = ProjectBudget - (ProjectBudget \times 5\%)$. Projects within budget but approaching this threshold are marked *Worry*, while projects exceeding the allocated budget or requiring additional funds are classified as *Critical*.

- Time Health: Schedule performance is categorized as *On Track*, *Worry*, or *Delay*. The time danger zone is calculated as

$TimeDangerZone = ProjectDuration - (ProjectDuration \times 5\%)$. A project is *On Track* if the estimated completion date remains within the approved duration. It is classified as *Worry* if delays approach the danger zone but remain within schedule limits, and *Delay* if the projected completion date exceeds the planned end date.

- Scope Health: Scope performance is assessed as *Excellent*, *Good*, or *Poor*. An *Excellent* scope corresponds to a score of 4.5 or higher, *Good* reflects a score between 3 and 4.5, and *Poor* indicates a score below 3.

B. Description and Scope of the Tool

RiskMIS is delivered as a web-based application designed to support organizational risk management activities. The tool enables project management, risk monitoring, and internal communication among users. Its primary focus is on supporting risk managers and project managers in managing risks and their impacts, rather than directly managing task execution by employees.

C. Requirements

A complete and structured specification of the system requirements was developed. Table VIII summarizes the functional requirements of the proposed tool, while Table IX presents the corresponding non-functional requirements.

TABLE VIII. FUNCTIONAL REQUIREMENTS OF THE PROPOSED TOOL

Category	Requirements
General Requirements	• A professional website will be designed and developed to effectively manage project risks in organizations, regardless of the business domain of the organization. • On the tool home page, the purpose and logo of the tool will be shown along with a brief description of the tool. • A link to other pages like About, Contact, Register, and Login should always be reachable from any of those pages and home page. • On the About page, a brief description of the reason for developing the tool will be provided. • On the Contact page, an email of the developer will be provided. • The tool targets organizations from all domains in both government or private sectors, and small, medium, or large organizations. • The tool requires registration of the organization by one Administrator. • Alert should be sent from the system to users if a big change has been done to their project like a new project, a new risk in the project, and requesting a new response strategy. • Users can read alerts by clicking the alert link.
Manage Employees	• The Administrator can create new employees to use the tool by filling the form which contains: the employee's name, email, ID, and role. • The roles of the users of the tool are limited to be A Project Manager, A Risk Manager, and An Employee. • When a new employee is created a verification email will be sent to the user containing a random password, if the user clicks the verification link, the user's account will be activated and then can use the email and the password to log in the tool. • The Administrator can prevent the users from using their accounts without deleting the accounts. • The Administrator can edit the user's information. • Users with the role "employee" cannot use the tool, the only reason why it's created is to assign responsibility to them in taking a practical action toward risk in the supervision of Project Manager.
Manage Risks	• The tool can manage risk by identifying project risks, analyze risks, and prepare a risk response strategy to implement. • It provides the ability to create, open, close, or suspend risks (including secondary risk). • The Risk Manager can create main categories and subcategories for the risks. • Only two-level risk categorization is allowed. • The tool provides default categories that the Risk Manager can use or delete if not needed. • The Risk Manager should specify the risk settings by ranking of probability, impact, and detectability based on its importance to the organization.

	• Risk settings specified by the organization will determine the risk impact in the project budget, time, and scope. • Risk Manager can create risks in the subcategories only and should specify the probability, impact, and detectability value of each risk. • The RM can edit created risks or delete them. • The RM can create response strategies for created risks in four types: accept, mitigate, transfer, and eliminate. • The Risk Manager can add risk to the project by determining the risk, risk owner, and the recommended response strategy. If the response strategy has a secondary risk, the RM should specify the risk owner and the response strategy for the secondary risk • The Project Manager can request a specific response strategy from the Risk Manager to prepare. • All details of project risk and impact are included in the project details. • The Project Manager should specify the implemented response strategy and its effect on cost, time, and scope of the project. • The tool can analyze the risks, calculates the probability and impact of risk on cost, time, and cost, then prioritize risks. • It can save the history of risks. • It provides different types of charts and graphs to represent risk management progress. • It can send notification automatically to the users when the status of risks or projects have changed.
Registration	• In the registration page, the Administrator should provide information about the organization like name and type, and information about the Administrator who will be the representative of the organization in the tool, like full name, civil ID, and password. • If the Administrator email already exists in the tool, an error message should appear, and registration will not be allowed. • Each organization should have only one administrator. • The administrator account is the owner of the organization record in the tool and has the highest privilege. • If the Administrator successfully submits the registration form, an activation email will be sent to the email of the Administrator with the activation link.
Communication	• All users can communicate using the tool by sending and receiving messages. • Users can send a message by filling the form with title, message, attached file if needed, and recipient. • The system will notify the user if there is a new message sent to them. • Users can read messages by clicking the notification of a new message or clicking the messages link to view all received messages.
Manage Projects	• The Administrators can create a new project "idea" by filling the form with the project's name, description, budget, start and end dates, project manager's name, and risk manager's name. • The Projects with status "Idea" will be shown only to the Administrators. • The Administrators can delete the projects with status "Idea" permanently. • The Administrators can edit the project information while in the "Idea" status.
	• The Projects can be initiated by Administrators only if all the project's information is filled. • The Administrators cannot edit Initiated projects. • The Initiated project will appear to the Project Manager and Risk Manager to whom the project is assigned. • If the project is in initiation state, a link will appear to the Project Manager to make a plan for the project • The Administrator can delete the projects after initiation only if it is in the "On Hold" state, the data of the project should be kept. • All users can view the project's details and progress by clicking the details link. • The Project Manager can make a plan to the project by filling a form, which consists of the task name, start and end dates, cost, and the requirements of the task. • The Task start and end dates and the costs should be within the project's budget and duration. • The Project Manager should create at least one task in the plan. • The Tasks of the project can be processed in parallel or serial. • The Project Manager can delete and edit plans or tasks only if the plan is not finalized/ confirmed. • If the Project Manager creates a plan for the project, a link will appear to the Risk Manager to add risks for the project. • The Project Manager can start, delete, on hold, complete, and resume tasks only if the plan is confirmed. • Any action to the task of the project should be stored in the tool like task start date, end date, cost, and scope. • The Project Manager should provide the cost and scope of the task when delete, hold or complete any task. • Task status, cost, time, and scope should be calculated and shown in the project details. • If the Project Manager wants to complete the task after resuming, the cost should be added to the previous cost of the task. • The Project's overall status, progress, and health information like cost, time, and scope should be calculated and updated continuously as the project progress and shown in the project details. • The tool should update the status of the project based on the tasks of the project status. For example, if all tasks are completed, the status of the project is "Completed". If at least one task is started, the project status will be "In process". If all tasks are on hold, the project status will be "On hold" and can be deleted. If all tasks are canceled the project status will be "Canceled". • It provides different types of charts and graphs to represent project management progress.
Logging in	• On the login page, users should fill the form with the email and password. • Only activated user accounts can log in. • Project Managers and Risk Managers can log in using the information in the email that was sent to them from the system which contains the user's email, a random password, and an activation link. • Administrators can log in using the email and password which they entered in the registration form.

	• If users forget the password, they can click the link "Forgot Password" to retrieve a new password from the tool to the user's email.

NONFUNCTIONAL REQUIREMENTS OF THE PROPOSED TOOL

Category	Requirements
Operational Requirements	• Website should be hosted on dedicated Server, with minimum of 2 x vCPU, 4 GB RAM, 2000 GB Bandwidth, 75 GB Disk Space, and 1000 Mbps Connection. • The tool should open from a common web browser like: Chrome, Firefox, IE, Safari, Edge, and Opera. • The tool should be friendly to portal devices like: Mobile and iPad. • The tool should be able to communicate with email web services to operate verification emails. • The tool should be able to communicate with google charts web service to create graphs and charts. • The tool should be able to communicate with the database. • The tool should give users the capability to upload files from different types.
Performance Requirements	• The tool should load in less than 2 secs. • The tool should be available 24/7. • The tool should always give real-time data. • The tool should be easy to navigate.
Security Requirements	• Only those who have registered usernames and valid passwords can access the tool. • Users can access the tool based on their role and function in the organization. • The Organization's projects, tasks, risks, and information should be accessible only to users with the required privileges and assigned jobs. • Login must be very secure with a maximum of three incorrect logins. • A time-out feature should be available on the tool if left idle for more than 5 minutes to protect the integrity of the organization's data. • Only developers of the tool will be able to log in to the backend of the tool to modify it.
Usability	• The tool displays the results and charts in easy to read and understand formats. • The tool is easy to use and facilitates the process of project and risk management. • The tool is user-friendly.

D. Stakeholders

The primary users of RiskMIS are Administrators, Project Managers, and Risk Managers, each interacting with the system according to their assigned roles. The Administrator holds the highest level of privilege and is responsible for creating users and configuring system access. Additional stakeholders include employees from various organizational units who execute risk response actions under the supervision of project managers. The Tool Owner is responsible for system maintenance, monitoring, error resolution, and ongoing updates.

IV. DESIGN AND IMPLEMENTATION

RiskMIS is a web-based Risk Management Information System designed to support organizational risk and project management activities. The tool is freely accessible and can be adopted by organizations of any size or domain, requiring only an internet-connected device and a standard web browser. RiskMIS is framework-agnostic and does not mandate adherence to any specific risk management standard or methodology.

RiskMIS provides integrated support for the core risk management lifecycle, consolidating risk identification, classification, assessment, response, and monitoring into a unified platform. In addition to risk management, the system offers essential project management functionalities, enabling project managers to create, plan, and track projects and tasks across their entire lifecycle. To enhance coordination and responsiveness, the tool also includes an internal communication center and an automated alert system, reducing the need for supplementary project management tools.

The system is implemented using the open-source ASP.NET Core framework and developed in C#. Microsoft SQL Server 2018 is used for data management, and Microsoft Visual Studio 2017 supported the development process. The current implementation comprises approximately 74,800 lines of source code and includes 40 ASPX-based web pages.

1) Overall architecture: RiskMIS follows a three-tier architectural model consisting of the Presentation Layer, Business Access Layer (BAL), and Data Access Layer (DAL). The Presentation Layer provides the user interface through web forms implemented using HTML, CSS, and JavaScript, ensuring compatibility across modern browsers and devices. The BAL, developed in C#, contains the application logic and mediates interactions between the user interface and the database. The DAL encapsulates data persistence functionality and performs Create, Read, Update, and Delete (CRUD) operations on the underlying database implemented using Microsoft SQL Server Management Studio 18.

In addition to core system components, RiskMIS integrates external services for email-based user verification and notification, as well as Google Chart services for visualizing project and risk-related data. The overall system architecture is illustrated in Fig. 1.

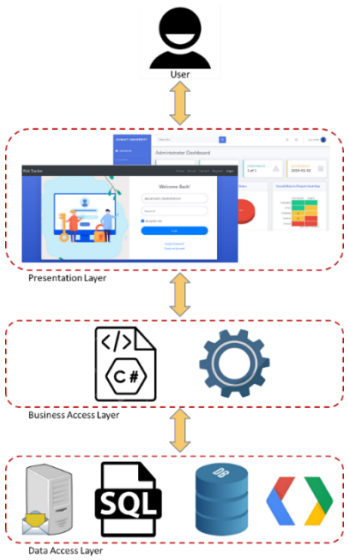


Fig. 1. RiskMIS architecture.

2) Use case model: The system's functionality is captured using a use case model that describes interactions between users and RiskMIS. Three primary user roles are defined: Administrator, Project Manager, and Risk Manager. The Administrator, typically representing top management, holds the highest level of privilege and is responsible for creating and managing users, assigning project and risk managers, and initializing project information. The Risk Manager oversees organizational and project-level risk activities, including risk configuration, prioritization, and monitoring. The Project Manager is responsible for managing the project lifecycle and implementing risk response actions. Fig. 2 presents the use case diagram depicting these roles and their interactions with the system.



Fig. 2. Use case diagram.

3) Activity diagrams: Activity diagrams are used to model and clarify key system workflows. Fig. 3 illustrates the login and dashboard access process common to all users. After credentials are submitted, the system validates the user's email and password against stored records. To enhance security, accounts are temporarily blocked after three consecutive failed login attempts, and a verification email containing a new activation link and password is automatically sent to the user.

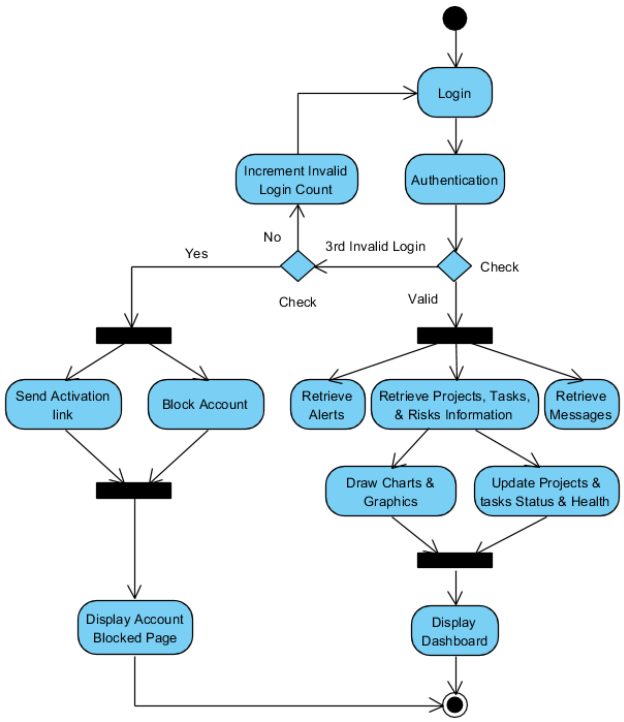


Fig. 3. Activity diagram for dashboard page display.

Fig. 4, Fig. 5, and Fig. 6 present activity diagrams specific to the Administrator, Project Manager, and Risk Manager, respectively, highlighting role-based workflows within RiskMIS.

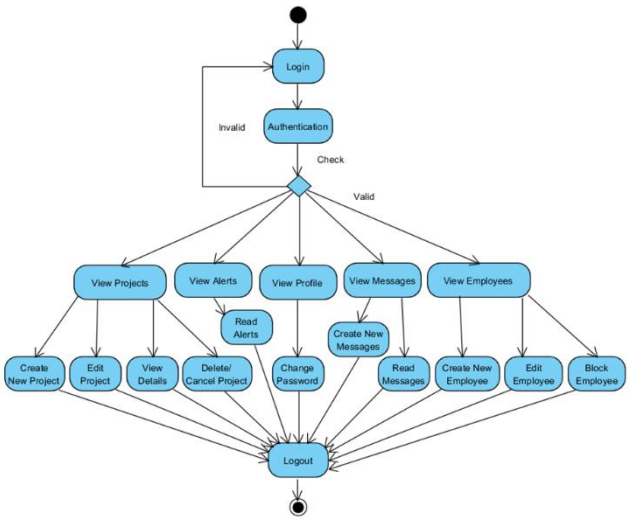


Fig. 4. Activity diagram for the administrator side.

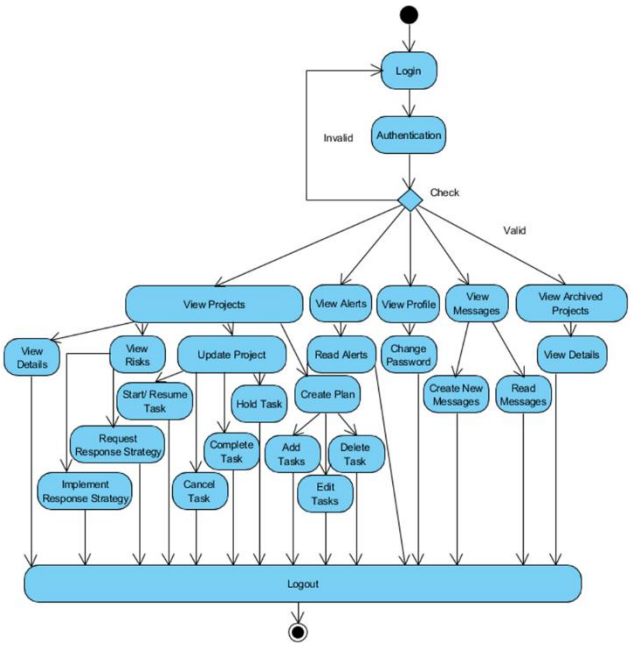


Fig. 5. Activity diagram for the project manager side.

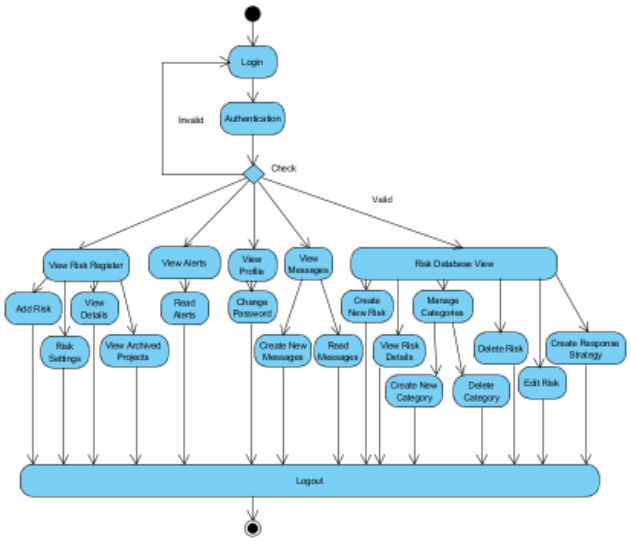


Fig. 6. Activity diagram for the risk manager side.

4) Finite state machines: RiskMIS behavior is formally represented using four finite state machines (statechart diagrams):

- Account State: Newly created accounts begin in a Not Verified state. Upon email verification, the account transitions to Active. Repeated invalid login attempts result in a Blocked state, from which the account can be reactivated via email verification. Administrators may also suspend or permanently delete accounts, as shown in Fig. 7.

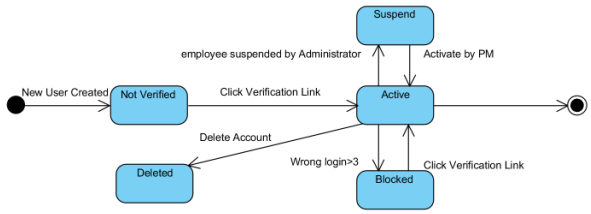


Fig. 7. Statechart diagram for the account state.

- Project State: Projects initially enter the Idea state, where they may be modified or deleted. Once project details are finalized, the project transitions to Initiate, granting access to the Project Manager. Subsequent task creation moves the project to the Plan state, followed by In Process, On Hold, Completed, or Canceled states depending on task progression and administrative actions (see Fig. 8).

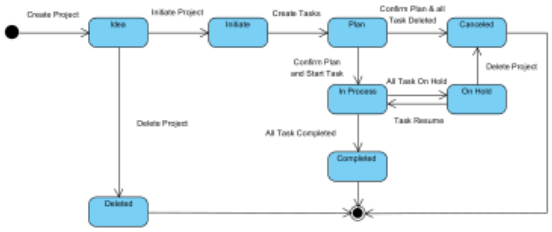


Fig. 8. Statechart diagram for the project state.

- Task State: Tasks start in the Not Started state after plan confirmation. They transition to In Process when execution begins, Delayed if planned timelines are exceeded, On Hold if temporarily suspended, and Completed once all requirements are satisfied (see Fig. 9).

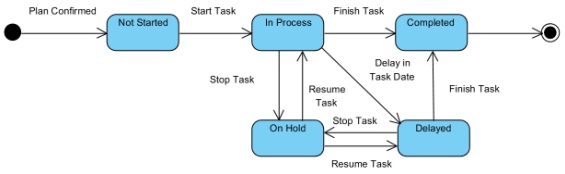


Fig. 9. Statechart diagram for the task state.

- Risk State: Risks initially exist in a Created state and transition to Open when associated with a project. Secondary risks generated through response strategies are initially marked as Pending and become Open once the corresponding response strategy is implemented (Fig. 10).

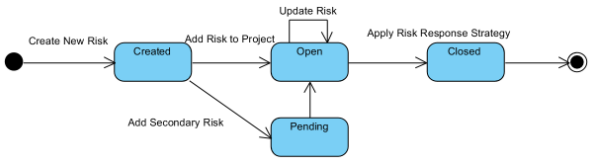


Fig. 10. Statechart diagram for the risk state.

5) Data model using the entity relationship diagram: RiskMIS relies on a relational database to support its functionalities. The database schema, implemented using Microsoft SQL Server Management Studio 18, consists of 14 tables: OrganizationTypes, Organizations, RiskSettings, AlertsFromSystem, EmployeesRole, Employees, Projects, Messages, MessageReceivers, Tasks, Risks, RisksInProjects, ResponseStrategy, and RiskCategories. These entities and their relationships are represented using an Entity Relationship (ER) diagram, as shown in Fig. 11.

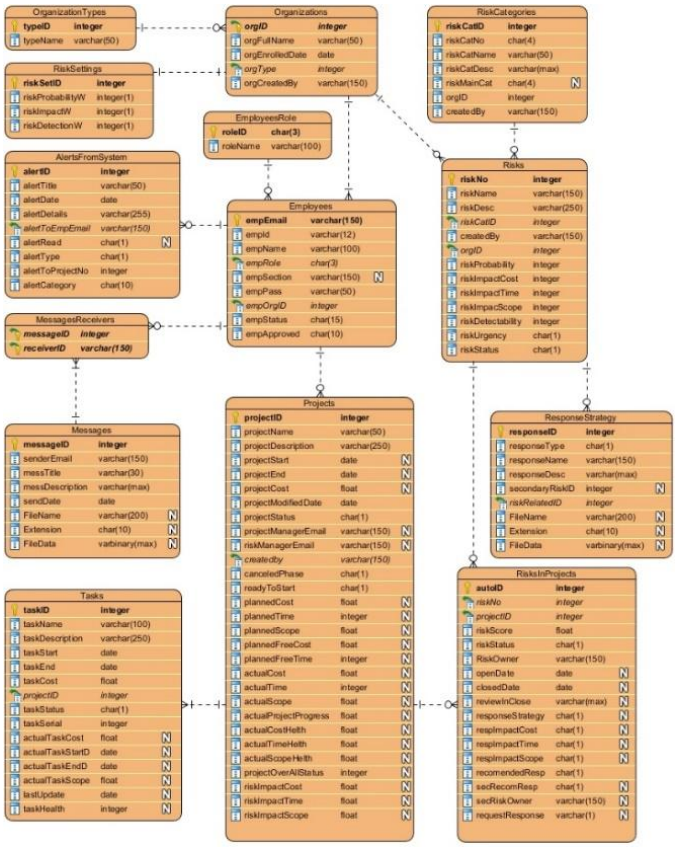


Fig. 11. Database schema for the RiskMIS.

V. Prototype Demonstration

To validate the practicality and effectiveness of the proposed RiskMIS, a fully functional web-based prototype was implemented and evaluated against the requirements defined earlier in the study. Rather than emphasizing user interface details, the demonstration focuses on what the prototype *proves* in terms of integrating project lifecycle management with continuous risk management.

Fig. 12 and Fig. 13 illustrate two important dashboards, which serve as the primary decision-support interfaces for all roles. These dashboards aggregate project and risk data in real-time and present key indicators such as project status distribution, budget utilization, projects at risk, and a risk heat map. This visualization demonstrates the system's ability to transform risk register data into actionable insights, enabling managers to rapidly identify high-risk projects and prioritize response actions without navigating multiple views or reports.

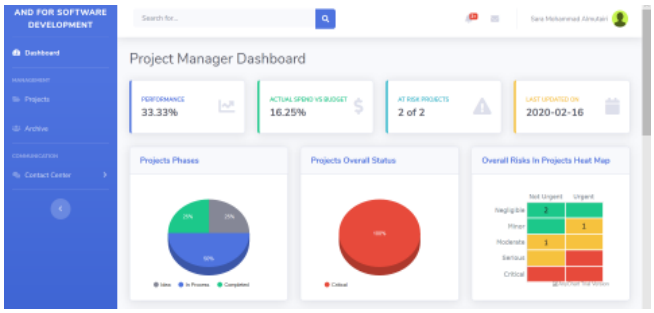


Fig. 12. Dashboard in the RiskMIS: statistics.

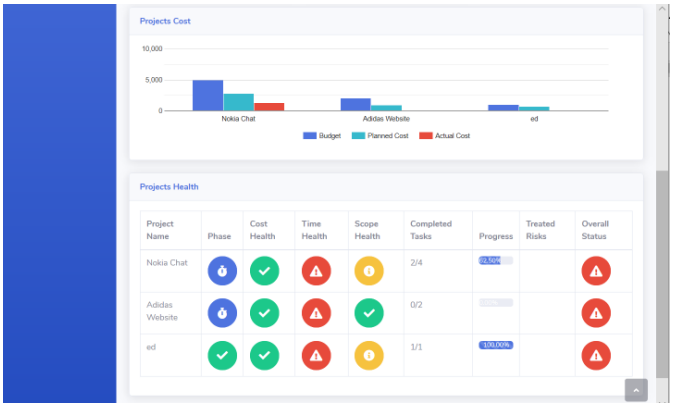


Fig. 13. Dashboard in the RiskMIS: information.

The prototype confirms that project lifecycle control and risk management are tightly coupled. Projects progress through predefined states, and task execution directly influences project health indicators related to cost, time, and scope. As risks are added or updated, their quantified impacts are automatically propagated to project performance metrics, demonstrating that risk assessment is not treated as a static activity but as a continuous driver of project status.

RiskMIS further demonstrates structured and consistent risk handling through a centralized risk register. Risks are categorized, assessed using standardized qualitative criteria, and prioritized according to organizational risk preferences. The prototype shows that recommended response strategies—acceptance, mitigation, transference, or elimination—are systematically linked to project execution. Importantly, the system proves its ability to manage secondary risks generated by response actions, reinforcing continuous monitoring and control throughout the project lifecycle.

Overall, the prototype demonstrates that RiskMIS can effectively unify project management and risk management within a single platform, providing real-time visibility into risk exposure and its operational consequences. This integration supports informed decision-making and proactive risk treatment across projects and ongoing operations.

The current evaluation is limited to functional validation using predefined scenarios and qualitative risk data. Large-scale empirical validation, performance benchmarking, and longitudinal studies in real organizational settings are planned as future work to assess scalability, user adoption, and decision-quality impact.

VI. Summary, Conclusion and Future Work

The primary objective of any organization is to deliver high-quality products or services that meet stakeholder requirements within defined time and budget constraints. Effective risk management is essential to achieving this objective, as it reduces uncertainty and supports informed decision-making. Central to this process is the risk register, which serves as a continuously updated repository of risk information and enables systematic identification, analysis, monitoring, and treatment of risks across project lifecycles and ongoing operations.

In this study, we conducted an extensive review of the risk management literature to identify key concepts, challenges, and research gaps. Building on this analysis, we defined the problem, proposed a solution, specified system requirements, and designed, implemented, and tested a comprehensive risk management information system. The main conclusions of this work are summarized as follows:

- Manual versus Automated Risk Management: Manual tools, such as word processors and spreadsheets, are inefficient and error-prone for managing risks. An integrated and automated risk management information system significantly enhances the consistency, efficiency, and effectiveness of risk identification, analysis, and treatment throughout the project lifecycle.
- Central Role of the Risk Register: The risk register is the core component of an effective risk management system. Its value depends on being comprehensive, dynamically maintained, and tightly integrated with all stages of the risk management process.
- Flexible Risk Classification: No single risk classification scheme is sufficient for all application domains. While the Larson and Gray classification is provided as a default, the proposed tool allows organizations to define and adapt classifications that reflect their specific context and risk landscape.
- Organization-Specific Risk Criteria: Consistent risk evaluation requires clearly defined and shared risk criteria. The system provides a generic yet extensible framework for defining risk probability, impacts on cost, time, and scope, detectability, and urgency.
- Generic Risk Management Framework: The tool implements a continuous and domain-independent risk management framework comprising five core activities: risk identification, classification, assessment, response, and monitoring and control. This framework is supported by explicit project and task lifecycles and state models.
- Structured Risk Prioritization: A multi-step risk prioritization methodology is introduced, enabling organizations to weight risk factors according to their priorities, compute risk scores, incorporate urgency, and classify risks on a standardized scale from negligible to critical.
- Integration with Project Performance Metrics: The system automatically propagates risk impacts to project cost, schedule, and scope using predefined formulas, ensuring that risk exposure is transparently reflected in project performance indicators.
- Management of Negative and Secondary Risks: The tool focuses on negative risks and supports systematic selection and implementation of response strategies. It also explicitly models secondary risks arising from response actions, enabling more comprehensive and continuous risk treatment.
- Web-Based and Framework-Agnostic Design: RiskMIS is implemented as a web-based, generic solution that can be readily adopted by organizations of different sizes and domains, without enforcing compliance with a specific risk management standard.

We are planning to evaluate RiskMIS through longitudinal case studies in real organizational settings. User interviews and empirical analyses will be conducted to assess system effectiveness, usability, and organizational impact. Future enhancements may include advanced analytics, intelligent risk trend detection, enhanced notification mechanisms, and broader user participation by enabling employees at all levels to report and review risks directly.

Future research will prioritize rigorous technical and empirical validation of the proposed RiskMIS through systematic performance benchmarking under realistic operational conditions to evaluate response time, scalability, and database efficiency. Clear quantitative evaluation metrics will be established to objectively assess system efficiency, transparency, decision-support effectiveness, and measurable impacts on risk exposure reduction, mitigation timelines, and decision-making accuracy through controlled case studies. In parallel, future work will incorporate formal security assessments covering authentication, authorization, and data protection mechanisms, alongside the integration of version control and change-tracking features to strengthen auditability and governance capabilities. Further studies will examine comprehensive data lifecycle management strategies, including archival policies, regular system maintenance procedures, automated backups, update deployment strategies, and long-term risk retention models. Finally, cross-organizational validation across multiple sectors, together with structured user training and onboarding frameworks, will be explored to enhance generalizability, system adoption, and sustainable usability transition.

References

[1] AFP. (2018). Risk survey report- key findings. Bethesda, MD, Association for Financial Professionals.

[2] Ahmeti, R., & Vladi, B. (2017). Risk management in public sector: a literature review. European Journal of Multidisciplinary Studies, 2(5), 323-329.

[3] Boehm, B. (1991). Software risk management: principles and practices. IEEE Software, 8(1), 32-41.

[4] Burcar, I., & Radujkovic, M. (2005). Risk registers in construction in Croatia. Proceedings of the Twenty First Annual Conference ARCOM, London, 171-178.

[5] Burcar, I., Radujkovic, M, Vukomanovic, M. (2013). Risk register development and implementation for construction projects. Gradevinar, 65(1), 23-35.

[6] Chihuri, S., & Pretorius, L. (2010). Managingrisk for success in a South African Engineering and Construction Project Environment. South African Journal of Industrial Engineering, 21(2), 63-77.

[7] CMMI Product Team. (2002). CMMI for software engineering, Version 1.1, Continuous Representation. Software Engineering Institute.

[8] De Wet, B & Visser, JK 2013, 'An evaluation of software project risk management in South Africa', South African Journal of Industrial Engineering, vol. 24, no. 1, pp. 14-28.

[9] Deloitte Touche Tohmatsu Limited (2017). Global risk management survey (10th ed.). Heightened uncertainty signals new challenges ahead. Deloitte University Press.

[10] Esteves, J., Pastor, J., Rodriguez, N., and Roy, R. (2004). Implementing and improving the SEI risk management method in a University Software Project. IEEE Latin America Transactions, 3(1), 90-97.

[11] GRA. (2005). Risk management guide for small business. Sydney, Global Risk Alliance Pty Ltd jointly with NSW Department of State and Regional Development.

[12] Hillson, D. (2003). Using a risk breakdown structure in project management. Journal of Facilities Management, 2(1), 85-97.

[13] Hopkin, P. (2010). Fundamentals of risk management: understanding, evaluating, and implementing effective risk management. UK, The Institute of Risk Management.

[14] IRM. (2018). A risk practitioners guide to ISO 31000. London, Institute of Risk Management.

[15] ISO. (2009). Guide 73: Risk management- vocabulary. Switzerland, International Organization for Standardization.

[16] ISO. (2018). ISO 31000: Risk Management - Guidelines (3rd ed.). Switzerland, International Organization for Standardization.

[17] Junior, R., & Carvalho, M. (2013). Understanding the impact of project risk management on project performance: an empirical study. Journal of Technology Management & Innovation, 8 (Special Issue ALTEC), 64-78.

[18] Kontio, J. (1997). The riskit method for software risk management, Version 1.00. Computer Science Technical Reports. University of Maryland, College Park, MD, USA.

[19] Larson, E., & Gray, C. (2015). Project management the managerial process (5th ed.). New York, McGraw-Hill

[20] Leva, M.C., Balfe, N., & McAleer, B. (2017). Risk registers: structuring data collection to develop risk intelligence. Safety Science, 100 (part B), 143-156.

[21] Liu, H., Deng, X., & Jiang, W. (2017). Risk evaluation in failure mode and effects analysis. Symmetry, 9 (62), 1-13.

[22] Martins, C. (2017). HoliRisk – risk assessment framework. Instituto Superior Técnico, Portugal.

[23] Mastroianni, S. (2011). Risk management among research and development projects (Master's thesis). Lehigh University.

[24] Mateus, G. (2016). A reference risk register for information security according to ISO/IEC 27005 (Master's thesis). Department of Science and Technology, Instituto Superior Técnico.

[25] Object Management Group. (2017). Unified Modeling Language (UML) Version 2.5.1. https://www.omg.org/spec/UML/2.5.1

[26] Onengiyeofori, O. (2016). Risk management system to guide building construction projects in developing countries: a case study of Nigeria (Doctoral dissertation). Faculty of Science and Engineering, University of Wolverhampton.

[27] Patterson, FD., & Neailey, K. (2002). A risk register database system to aid the management of project risk. International Journal of Project Management, 20(5), 365-374.

[28] PMI. (2025). A guide to the project management body of knowledge: PMBOK Guide (8th ed.). Pennsylvania, Project Management Institute, Inc.

[29] Pritchard, C. (2015). Risk management concepts and guidance (5th ed.). Boca Raton, CRC Press.

[30] Rehacek, P. (2017). Risk management standards for project management. International Journal of Advanced and Applied Sciences, 4(6), 1-13.

[31] Rodage, H., Lei, H. & Ganjeizadeh, F. (2014). Risk management for research and development projects. International Journal of Engineering Research & Technology, 3(10), 824-831

[32] Rorvik, K. (2013). Risk management in a complex frame agreement - a case study from a contractors perspective (Master's thesis). Faculty of Science and Technology, University of Stavanger.

[33] Saffin, T. & Laryea, S. (2012). The use of risk registers by project managers. In: 4th West Africa Built Environment Research (WABER) Conference, 2426 July 2012, Abuja, Nigeria, 1305-1318. Available at http://centaur.reading.ac.uk/30322/

[34] Saleh, K. (2009). Software Engineering (1st ed.). J. Ross Publishing, 26-28

[35] The Standish Group. (2020). CHAOS report 2020: Beyond Infinity.

[36] Uzulans, J. (2016). Project risk register analysis based on the theoretical analysis of project management notion of risk. Economics and Business, 29(1), 43-48.

[37] Vaz, L. (2016). Risk management in information systems projects (Master's thesis). Department of Science and Technology, Instituto Superior Técnico.

Uncertainty-Aware Volumetric Transformer with Dual Spatial-Channel Attention for Lung Nodule Classification

Dr B.N.Patil[1], TK Rama Krishna Rao[2], Nurilla Mahamatov[3], Elangovan Muniyandy[4], Dr.Arun Prasad.VK[5], Dr. Chamandeep Kaur[6], DR. Aaquil Bunglowala[7], Ahmed I. Taloba[8]

Assistant professor, Department of Applied Physics, Yeshwantrao Chavan College of Engineering, Nagpur, India [1]

Professor, Department of Computer Science and Engineering, Koneru Lakshmaih Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India-522302.[2]

Department of Automatic Control and Computer Engineering, Turin Polytechnic University in Tashkent, Uzbekistan [3]

Department of Biosciences-Saveetha School of Engineering, Saveetha Institute of Medical and Technical Sciences, Chennai - 602 105, India[4]

Assistant Professor, Dept. of Anatomy-Symbiosis Medical College for Women, Symbiosis International (Deemed University), Pune, India[5]

Assistant Professor, Department of Computer Science & Information Technology, Jazan University, Jizan, Kingdom of Saudi Arabia[6]

Director SAIT and Professor, ECE, Sri Aurobindo Institute of Technology, Indore, M.P. [7]

Department of Computer Science-College of Computer and Information Sciences, Jouf University, Saudi Arabia.[8a]

Faculty of Computers and Information-Information System Department, Assiut University, Assiut, Egypt.[8b]

Abstract*—Lung cancer is also among the most common causes of cancer-related deaths in the world, and the earliest possible detection of the cancer through computed tomography (CT) is important in the enhancement of patient survival. Nevertheless, accurate diagnosis is still a challenge as the nodules are small and indistinct, inter-rater consistency among radiologists, and the traditional deep learning systems have limited capacity to handle volumetric interactions and give interpretable and confidence-aware forecasts. This research suggests an uncertainty-cognizant Transformer-Enhanced Dual-Level Attention Network (TDA-Net) to classify lung nodules in CT images to deal with these issues. The suggested architecture combines a 3D Swin Transformer backbone and sequential spatial and channel attention fusion to be able to model both localized structural and global volumetric context. Moreover, Monte Carlo dropout is used in inference to measure predictive uncertainty and allows low-confidence cases to be identified and sent to a radiologist. The model is tested on a publicly available lung CT dataset, and it has an accuracy of 98.3% with high sensitivity to small nodules in the feature space. There is a separation of classes in the feature space, and the uncertainty rate is 5.1%. The findings of the experiment indicate that TDA-Net can be used as a supportive decision-making tool to diagnose lung cancer with the help of computers because it has better discriminative performance and uncertainty awareness when compared to the baseline models. Moreover, distinguishable uncertainty of predictions and uncertainty of models are present. Predictive uncertainty is measured through the variance of softmax probability distributions through stochastic forward passes, which is related to the ambiguity of data. Monte Carlo dropout is used to estimate model uncertainty as a Bayesian approximation, which represents parameter-level uncertainty due to a small amount of training data.

Keywords*—3D Swin Transformer; dual-level attention mechanism; computed tomography; lung cancer diagnosis; uncertainty-aware deep learning

I. Introduction

Lung cancer is a cause of cancer-mortality around the globe, and it causes almost a quarter of all cancer-deaths. Computed tomography (CT) leads to early diagnosis and a much better survival [1]. Unfortunately, the interpretive elements of CT scans require a considerable time commitment from radiologists and may be subject to inter-observer variability [2]. The conventional deep learning methods, in particular, 2D CNNs, can only handle slices of CT images separately and fail to utilize slice-to-slice volumetric dependencies [3]. Even existing 3D CNN models have limited capacity to model long-range spatial relationships and generally act as black-box systems that are not interpretable [4]. Typical differences in lung nodule characteristics regarding their size, shape, texture, and combined with varying malignancy levels, create significant challenges for accurate analysis, especially for difficult-to-classify small nodules or those with unclear diagnostic indicators [5]. The diagnosis becomes challenging due to ambiguous nodule appearance because benign nodules look similar enough to malignant ones that physicians sometimes mistake them for malignancies, thus triggering nonsensical procedures [6]. New transformer architecture advancements have demonstrated potential in medical imaging; however, their combination with clinically sound levels of confidence estimation remains a relatively untapped field [7]. The earlier studies have indicated positive outcomes in the process of identifying lung cancer through intensive learning, but have significant limitations. Conventional 2D CNNs primarily examine individual slices and do not obtain full volumetric reliance in CT scans, which are

significant in identifying small or micro nodules [8]. Although it is solved in 3D CNN due to the processing of volumetric data, they frequently have to work locally to extract convenience and cannot take into account the relevant information of long distances in other areas of the lungs [9]. This restricts their ability to distinguish benign and fatal nodules that have the same local features. Moreover, majority of the existing methods are black boxes, which do not reveal much information about the way of making predictions. This study presents a novel Transformer-Augmented Dual-Level Attention Network (TDA-Net) that exploits the power of the 3D Swin Transformer architecture with Spatial Channel Attention Fusion and uncertainty estimation to generate the right, explainable and confidence-aware lung nodule detection. The model serves the clinical need of the reliable use of AI predictions to detect nodules and enhance sensitivity to small nodules that may be overlooked in early diagnosis processes [10].

A. Problem Statement

Lung cancer is among the common causes of cancer-related death in the world, and even with the extensive use of computed tomography (CT), it is still difficult to achieve an accurate and consistent diagnosis. The fact that pulmonary nodules are small, irregular, and subtle, the presence of overlapping anatomy, large inter-patient variation, and inter-radiologist subjectivity limit the detection of small nodules. Traditional computer-aided diagnostic models, especially those that rely on 2D convolutional neural networks (CNNs), do not capture inter-slice relationships and volumetric continuity between CT data [9]. In spite of the fact that 3D CNNs, to some extent, overcome this drawback, they are limited by local receptive fields, restricted long-range spatial modeling, and low interpretability. Moreover, the current methods usually have a problem of class imbalance, weak population generalization [11], insufficient sensitivity to micro-nodules, and a lack of clinically significant uncertainty measures. Absence of transparency and quantification of confidence are huge limitations to clinical adoption in the real world [12]. The current volumetric CNN and transformer-based methods are mostly aimed to achieve better classification, however, without calibrated and explicitly modeled uncertainty estimations that can be used in situations with safety-critical ramifications in medical contexts. Even the frameworks of current transformers do not include the systematic incorporation of dual-level spatial-channel attention to improve the interpretability and micro-nodule discrimination in a single volumetric framework. In addition, cross-dataset validation is limited and uncertainty modeling has no adequate theoretical basis, limiting clinical reliability. These restrictions indicate a perceptible gap in research to develop a volumetric transformer design that concurrently assists in proper classification, organized attention-based interpretability, and theoretically founded, well-calibrated uncertainty estimation towards resilient clinical decision-making.

B. Recent Innovations and Their Limits

The recent developments in medical image analysis have proposed transformer-based architectures to overcome the small receptive fields of CNNs. In this respect, a 3D Swin Transformer based on the proposed Transformer-Augmented Dual-Level Attention Network (TDA-Net) is combined with the new Dual-Level Spatial–Channel Attention Fusion (DSAF) mechanism to concurrently characterize local volumetric and global contextual relationships. Unlike the classical CNN-based approaches [13], the sequential attention architecture focuses on spatially salient regions and changes the weights of feature channels over time, more accurately detecting micro-nodules and with improved interpretability. Nonetheless, existing verification has only been demonstrated on relatively small and homogeneous datasets and transformer-based architectures are more expensive to compute. The further research will involve large-scale multicenter validation, optimization of efficiency, and more advanced explainability methods to enhance clinical applicability.

C. Research Motivation

Transformer and attention-based models has shown to be highly effective in long-range dependency and task-relevant representation in computer vision and medical imaging. These models in contrast to the traditional CNNs are effective at incorporating both local and global information, which is essential in the analysis of complex 3D CT volumes, in which minute nodules can be present in non-contiguous slices. Nevertheless, they have not been thoroughly investigated as far as the diagnosis of volumetric lung cancer is concerned [13]. Attention mechanisms also allow highlighting clinically relevant areas, whilst suppressing irrelevant background information, which enhances interpretability. Driven by these strengths, this study intends to utilize transformer-based volumetric modeling and dual-level attention to create a well-built and explainable framework that is uncertain and effective at early lung cancer diagnosis to finally aid radiologists in having reliable and stable decision support mechanisms.

D. Significance of the Study

The proposed research is important because it presents a new uncertainty-sensitive deep learning framework (the combination of 3D transformer-based global representation, and the dual-level spatial and channel attention to detect lung cancer). The proposed approach is more sensitive to small and early nodules that are usually missed by the traditional approaches by explicitly models long-range volumetric information and selectively highlighting those features that are important diagnostically. The uncertainty estimation support also increases clinical reliability by marking predictions of low confidence to be reviewed by experts. Having methodological contributions, the framework should assist radiologists, decrease diagnostic variability, and make AI-assisted workflow safer. On the whole, this piece enhances the medical image analysis and leads to the creation of clinically viable and interpretable decision-support systems.

E. Key Contribution

- Suggests a volumetric-based feature extraction, which simultaneously features fine-grained nodule features and macro-level contextual features, which play significant roles in the early diagnosis of lung cancer.
- Combines dual-level spatial-channel attention to learn the properties of the discriminant, and also to improve interpretability to produce decision which can be implemented in a clinical setting.

- Applies uncertainty estimation to emphasize on the low-confidence estimates to foster reliability and can be safely applied in the diagnostic processes.
- Provides compelling evidence that TDA-Net achieves 98.3% accuracy, 0.98% precision and 0.97% sensitivity on the Kaggle dataset, superior to the existing approaches.

F. Rest of the Sections

In Section II, the review of existing lung cancer detection approaches based on CT imaging is provided, emphasizing such recent discoveries as 3D CNNs, model combinations, and attention mechanisms. Section III gives the approach employed for extracting attributes out of 3D CNNs and fusing Dual-Level Spatial-Channel Attention. In Section IV, the classification pipeline, training model with Tensorflow and Keras, testing models on different metrics and the trends in the classification across all the cancers and tumor sizes are described. Section V ends the study and future directions are the incorporation of explainable AI and application of imaging other than vision, and the implementation of the method in clinical scenarios in real-time.

II. Related Works

Wang et al. [11] proposed the use of 3D Auto-encoder networks with attention mechanisms to identify lung cancer indications and developed a dual model with 3D networks. The applied analysis of the 3D Auto-encoder occurred in its initial state through RFE, LASSO, and ANOVA for testing the features, also, to perform the tests on SVM, RF, GBM, MLP, LightGBM, XGBoost, Stacking and Voting classifiers. Implementation of attention modules as an addition to the network achieved a higher measurement (A) at 93.4%, along with 90.2% sensitivity and AUC-ROC at 94.1%.

Crasta et al. [9] present a new DL framework for lung cancer detection through CT scans, along with classification functionality, which stands as the main contribution in this study. This system uses 3D-VNet for segmentation, together with 3D-ResNet for classification, with a LUNA16 test result of 99.34% DSC alongside 0.4% fewer false positives. The classification model exhibits 99.2% (A) together with 98.8% sensitivity and 99.6% specificity to surpass current methodologies. The approach suffers from two main restrictions caused by variations in datasets, combined with the necessity to enhance interpretability mechanisms. Future studies need to leverage different medical image datasets while using PET/MRI fusion techniques and improve model interpretation capabilities for emergency room standards and establish ongoing nodule growth monitoring techniques to optimize early diagnosis practices and treatment plans.

Tang et al. [14] examined individuals and received "low-dose computed tomography" (LDCT) scans for lung cancer screening throughout the period from 2005 to 2019. NCCN criteria determined three risk categories, which studier's used to classify their participants. Of the participants whose baseline scans showed positivity, the study identified 3695 cases (11.8%) with 197 of these disclosures proving to be lung cancer patients (0.6%). The risk group scored the highest for lung cancer detection (1.4%), but second-hand smoke female participants detected more cases (1.1%) than other non-risk groups (0.5%). Certain challenges affect this study through self-referral constraints and selection biases and non-adherent patient behavior, as well as known false-negative tests. Better risk assessment requires future study to examine different populations and confirm those cases diagnosed as lung cancer-negative.

Huo et al. [12], using a "self-paced learning" (SPL) 3D Net, created two (DL) models to predict M/S growth patterns of invasive lung adenocarcinoma (ILADC) through pre-operative CT scans obtained from 617 internal patients (for model testing) and 353 external patients. The evaluation of Model 1 in external conditions resulted in a 0.857 AUC score and Model 2 demonstrated a 0.831 AUC score for patients with ILADC smaller than 2 centimeters. Both models outperformed "ResNet34", "ResNet50", "ResNeXt50", and "DenseNet121". The application of DL models in clinical diagnostics faces constraints from minor clinical details and unclear internal logic that needs improved study for practical implementation, while retaining the potential of improved diagnostic (A).

Wang et al. [15] conducted an analysis of 20 investigations which used PubMed, Embase, and Web of Science to study how DL methods and expert diagnostic radiologists performed when diagnosing lung cancer in CT images. The DL model achieved better (A) than radiology experts did through its 82% sensitivity and 75% specificity measurements, while showing statistically meaningful higher specificity levels. The test results yielded different findings when examining standard CT versus LDCT images because DL methods achieved better specificity on standard CT, although it produced higher sensitivity at decreased specificity on LDCT scans. Further development of AI diagnosis utilization requires prospective patient-specific validation and assessment of radiologist expertise, as well as methods for direct AI involvement and improved methodologies for handling aggregated data.

Klangbunrueang et al. [16] proposed VGG16, which was also trained and tested for categorizing the lung cancer CT images into Benign, Malignant, and Normal using 1097 CT images of 110 patients. The study methodology combined data acquisition and preprocessing with VGG16 training steps, as well as testing protocols that were also applied to "ResNet50", as well as InceptionV3 and MobileNetV2. The VGG16 model reached the best diagnostic (A) level of 98.18% during testing. Additional study needs to investigate data augmentation techniques and model fine-tuning and it should test both 3D CNN networks and Vision Transformers for better performance.

Manum et al. [17] proposed the CNN framework to detect lung cancer early on by analyzing images from a CT scan. This was achieved by training and evaluating CNN along with InceptionV3, Xception, and ResNet-50 in term of performance using (A), AUC, (R), and metric of losses. It was realized that CNN gave outperformed results compared to all other models, achieving 92% (A), 98.21% of area under curve, 91.72% (R), and loss of 0.328, which reflects at its capability than others and the traditional methods. The downside to establishing such works over larger datasets and considering other DL models would improve generalizability for improved early detection and diagnostic.

John [18] extracted patient record features using DL approaches, including CNNs, RNNs, and transfer learning models, to aid in lung cancer detection from medical images, along with combining image preprocessing by augmenting with various classification techniques for greater (A). One focus is on bridging aspects in data imbalance, overfitting, and interpretability, as its comprehensive framework also offers efficient diagnosis performance. Kunkyab et al. [19] proposed a multi-resolution (DL) framework by utilizing multi-depth backbones, which retains the advantages of low and high-resolution images. The proposed framework was evaluated on the NSCLC dataset benchmarked against five (DL) models. The results demonstrated superior performance with a Dice score of 0.92 and a Hausdorff distance of 1.33, which indicates the potential for efficient auto-segmentation arising for clinical workflows.

III. Proposed Methodology Feature Extraction using 3D CNN

The proposed work presents a Transformer-Augmented Dual-Level Attention Network (TDA-Net) to study volumetric lung cancer, which is a Swin Transformer that consists of a 3D backbone and an additional Dual-Level Spatial Channel Attention Fusion (DSAF) block. In contrast to the traditional CNN-based method, the framework has the advantage of modeling localized volumetric features and long-range contextual dependencies across volumetric CTs jointly and selectively refining spatially salient areas and diagnostically relevant feature channels sequentially, enhancing micro-nodule discrimination and interpretability. The main novelty is that transformer-based global modeling is coupled with the hierarchical attention to improve sensitivity and clinical transparency. The further work will be done on a large-scale multicenter validation, optimization of efficiency and deep explainability methods to reinforce the clinical applicability. Preprocessing and lung segmentation are followed by hierarchical feature extraction, which is done using a 3D Swin Transformer. Volumetric features are enhanced by dual-level spatial and channel attention and confidence-aware lung cancer classification by an uncertainty estimation module. It is shown in Fig. 1.

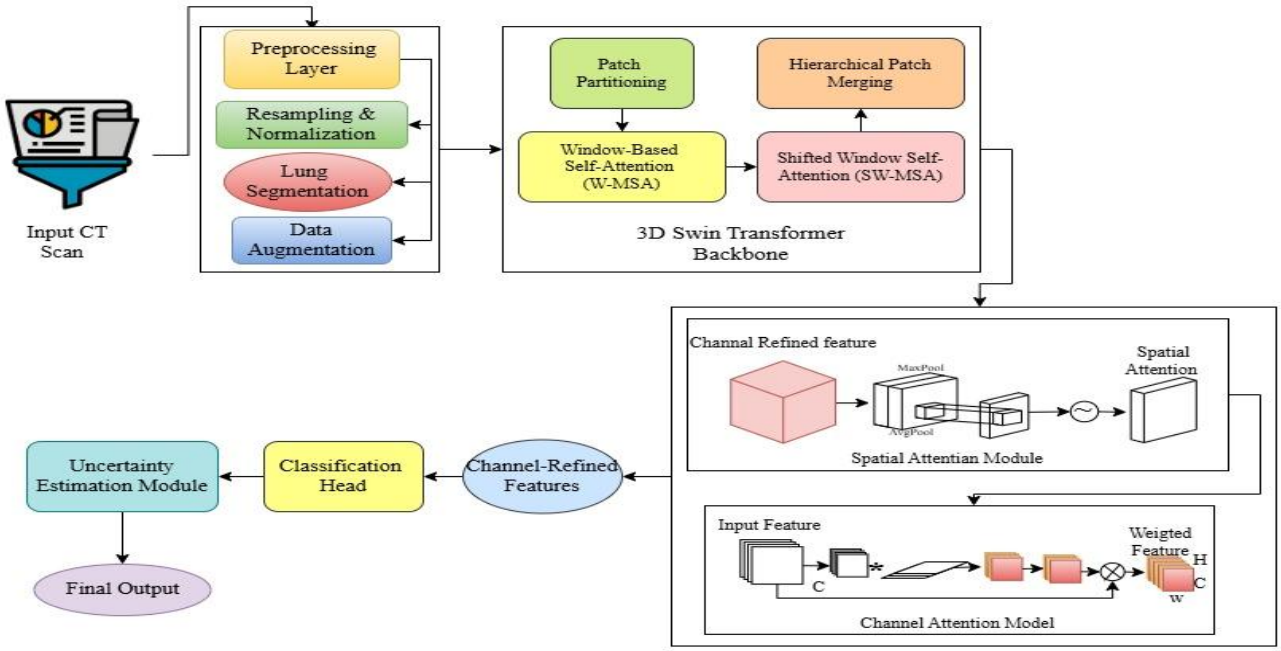


Fig. 1. Uncertainty-aware dual-attention TDA-Net framework.

A. Data Collection

This study utilizes the publicly available lung CT datasets to evaluate the proposed framework. A proof-of-concept analysis is first conducted on the Kaggle Lung Cancer CT Scan Dataset [20], consisting of 315 CT images in four classes, adenocarcinoma (120), squamous cell carcinoma (90), large cell carcinoma (51) and normal lung tissue (54) are first analyzed through a proof-of-concept analysis. Being small, this data can be experimented with quickly and initial feasibility checked. A model is tested on the LIDC-IDRI / LUNA16 dataset consisting of more than 1,000 CT scans labeled with the pulmonary nodules by radiologists to further confirm that the model is robust and can be generalized to a broader population. The large-scale benchmark can be used to establish a reasonable comparison with the current methods, as well as guarantee consistency in performance despite going beyond small, carefully curated datasets. In order to assess robustness between variability in acquisition and demographic diversity, subgroup analysis was conducted depending on the scanner protocols, variation in slice thickness, and age groups in patients that were present in the LIDC-IDRI/LUNA16 data. The dataset has a variety of multi-institutional scans taken with various CT instruments and imaging settings, which allows determining the stability of such scans in a heterogeneous clinical setting. To determine consistency of performance and uncertainty calibration measures across acquisition and demographic partitions, performance and uncertainty calibration measures were calculated separately within each subgroup.

B. Data Preprocessing

The preprocessing of CT images defines the best quality and enhances the models. The processing steps that this dataset requires are the following essential items.

Resizing and Normalizing: The images of CT scan are processed into a standard size to provide uniformity across the dataset. The normalization method is used to put the pixel values in a range of [0,1] by dividing it by 255 or to put the pixel values to a zero mean and unit variance value. This step is part of model performance enhancement as it makes computational operations simpler besides stabilizing the input data consistency. Given an original picture whose dimensions (J, K, L), a resizing process changes it to a constant size (J -, K -, L -). It is given in Eq. (1):

$$I' = \mathrm{I}(J', K', L') \tag{1}$$

where, J' and K' are the target height and width, and L is the number of channels. Normalization is calculated using Eq. (2):

$$I_{norm} = \frac{I - I_{min}}{I_{max} - I_{min}} \tag{2}$$

where, I is the original pixel intensity, I_{min} and I_{max} are the maximum and minimum pixel values in the image. It is given in Eq. (3):

$$\mathrm{I}_{max} = \frac{\mathrm{I}}{255} \tag{3}$$

where, $\mathrm{I}_{min} = 0$, and $\mathrm{I}_{max} = 255$.

1) Noise reduction using Gaussian filtering: The Gaussian filtering technique is one of the most frequently utilized medical image preprocessing techniques because it removes random noise in data without interfering with important structure information. Segmentation (A) of the lung CT images is negatively influenced by sensor artifact noise and patient motion and scanning inconsistencies. The method of smoothing images with the help of the Gaussian filter is based on the usage of a Gaussian kernel, which assigns more values to the central pixels and less to the surrounding ones. High-frequency fluctuations in intensity are minimized, thus high-frequency noise is blocked, but lung boundaries are not altered by this process of weighted averaging. This is the definition of the Gaussian function [see Eq. (4)].

$$G(m, n) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

where, σ controls the degree of smoothing, a small σ preserves details, while a larger σ smooths more aggressively. The application of Gaussian filtering occurs prior to segmentation because it enhances lung structure visibility through grainy noise reduction. The application of Gaussian filtering enhances both feature extraction abilities and classification (P), along with disease recognition, so that CT image interpretation becomes more precise and reliable.

Overfitting is believed to cause models to be more robust and generalizable with the addition of data. The CT volumes are normalized first before resizing to a given resolution. Once this has been done, the augmentation attempts to reproduce variances observed in real scans of the lungs. This covers the most basic forms of transformations; random rotations, flipping the images along random axes, translations, zooms, and intensity scaling in order to neutralize the varying patient positioning and other scanner settings. These changes are useful to ensure that the model does not overfit since it must not be memorized using the small amount of data, but rather create more generalized representations of lung nodules and tumor properties.

C. 3D Swin Transformer Feature Extraction

The 3D Swin Transformer that replaces the conventional 3D CNN layers in the volumetric feature extraction is the building block of the discussed model. Unlike fixed receptive field-based CNNs that basically learn local spatial features, the 3D Swin Transformer combines global dependency learning (hierarchical aggregation) and local context modelling (shifted window attention). This is particularly appropriate in lung CT scans, where local delicate nodules as well as the global spatial relationship between slices are of diagnostic importance.

1) Patch partitioning: By the size of $P \times P \times P$ the CT volume $V \in \mathbb{R}^{H \times W \times D}$ further partitioned into 3D patches which are non-overlapping. Every patch is flattened into a vector and added to an embedding site by a linear projection and calculated by use of Eq. (5):

$$z_0 = Linear\left(PatchPartition\,(V)\right), \quad z_0 \in \mathbb{R}^{N \times d} \tag{5}$$

where, $N = \frac{HWD}{P^3}$ denote the number of patches and d denote the embedding dimension.

In Standard Window-based Multi-headed Self-essence (W-MSA), attention is confined to an overlapping non-overlapping 3D window, which is appropriate to local dependence but does not work in interactions across the window boundaries. In order to overcome this limit, shifted window (SW-MSA) mechanism is added in the latter layer where each 3D window is sent over half a window shape (M/2). These are easy but effective modifications such that ensure that the tokens that are found in the edges of the windows in the W-MSA are transformed into the center region of a window in the SW-MSA so that the cross-window connections could take place. Consequently, the model acquires local and global inter-etc. dependence, which makes window-based attention computationally efficient. It is calculated by the following Eq. (6). Patch merging with hierarchy can be used to extract local and global volumetric representations progressively to eventually generate uncertainty-sensitive lung cancer risk classification. It is illustrated in Fig. 2.

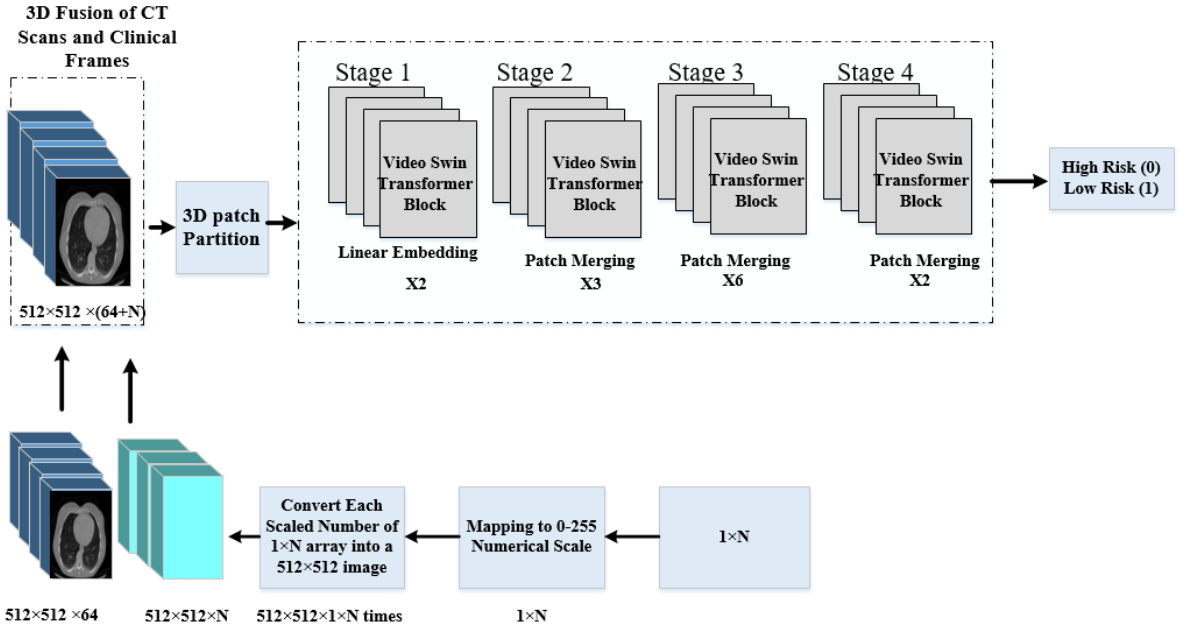


Fig. 2. TDA-Net architecture for volumetric lung analysis.

$$z^{l+1} = SW.MSA\left(LN(Z^1)\right) + z^1 \quad (6)$$

$$z^{l+2} = MLP\left(LN(Z^{l+1})\right) + z^{l+1} \quad (7)$$

In Eq. (6) and Eq. (7), layer normalization is abbreviated as LN, feed-forward network post-attention as MLP and the skip connection as (+) is identical to the residual learning as a stability measure. This variant W-MSA -SW-MSA architecture guarantees every token to have interactions with not only its local community, but also with the tokens in the windows adjacent to it, essentially occupies global inter-black dependence in CT volume.

2) Hierarchical representation: The patch combining of tokens is achieved as follows: once a set number of Swin Transformer blocks has been completed, they are combined, much like in CNNs by pooling, but in this case, it is Transformer embeddings that are combined. Provided a stage s feature map on resolution (Hs.Ws. Ds patch embeddings, Patch Merging combines (on a patch-by-patch basis) patch embeddings at adjacent locations (e.g., 2x2 patches), halving the spatial resolution by a factor of 2, but compensating it by doubling the channel dimension by a factor of 2xCs. The operation generates a hierarchical, multi-scale representation, which is replete with significant volumetric information but that describes successively abstract features that are significant in the detection of lung nodules of various sizes. Joining adjacent patch embeddings caused resolution to decrease in all spatial dimensions, which is given in Eq. (8):

$$(H_{s+1}, W_{s+1}, D_{s+1}) = \left(\frac{H_s}{2}\frac{W_s}{2}\frac{D_s}{2}\right) \quad (8)$$

To increase the channel dimension the linear projection is computed using Eq. (9):

$$C_{s+1} = 2 \times C \quad (9)$$

Finally, the output features at stage s+1 are computed using Eq. (10):

$$F_{s+1} = PatchMerging\ (F_s) \in \mathbb{R}^{H_{s+1} \times W_{s+1} \times D_{s+1} \times C_{s+1}} \quad (10)$$

The process generates a stratified feature hierarchy where the lowest levels of the hierarchy retain finely dated structural information, such as small nodules and boundaries and edges, whilst deep levels of the hierarchy retain high-level global information, such as the overall size of the lungs, spatial relationships within areas, and area tissue density variation. This hierarchical model provides the ability to model both fine local detail and large volumetric detail, which is vital to the identification and categorization of the precise lung nodules.

3) Feature extraction output: The last step of the 3D Swin transformer is a compact volumetric feature representation that is rich in information and is used as a base to redefine focus and eventually classify. In the earlier phases, the input volume of CT is subdivided into the patch, which is undergone by alternative window-based and shift-window meditation layers and the patch is merged as it gradually undergoes down sampling. Not only that this process minimizes the overall spatial dimensions of feature maps but it also maximizes the representative depth of the feature maps, allowing the model to experience the balance between local and global information. The resulting feature map represents the fine-dominated structural features like small nodules and edges and high-level relevant features like size, tissue density and inter-regional dependence. The volumetric features are determined by following Eq. (11):

$$F = Swin\ 3D\ (V), F \in \mathbb{R}^{H' \times W' \times D' \times C} \quad (11)$$

where, input CT volume is depicted by V, merged spatial dimension following patch merging is represented by $H' \times W' \times D'$ and finally the number of output channel is denoted as C. They eliminated the facilities.3D Swin - borrowed directly out of the SWIN transformer - is not actually a prediction done on its own and they merely run it in the dual-level spatial-channel focus (DSAF) module to be refined further. During this refinement step, the first step is to provide spatial attention in order to highlight volumetric regions, namely nodule in order that the network draws attention to the clinically relevant regions of interest and secondarily, in order to selectively emphasize the discrimination facilities involved in the nodule characteristics like texture, density and size. Through this, 3D Swin Transformer serves as a backbone feature extraction step, which merely gets multi-scale volumetric embeddings, whereas the DSAF module serves as a refinement step, enhancing the power of these embeddings to be more reproducible for accurate lung nodule detection and subsequent classification.

D. Dual-Level Spatial–Channel Attention Fusion (DSAF)

The embedding obtained after feature extraction using the 3D Swin transformer is further enhanced by the double-level spatial-channel attention fusion (DSAF) module to enhance the discriminatory power. Spatial attention and channel attention is gradually applied in this module. The emphasis is laid on the fact that in spatial areas, it is more likely that the volume will be involved in the volume to be present through high loads assigned. It is computed using Eq. (12):

$$M_s(F) = \sigma(f^{7\times7\times7}(|Avg\ Pool_c(F); MaxPool_c(F)] \quad (12)$$

where, $F \in \mathbb{R}^{H'\times W'\times D'\times C}$, the pooling operation along the channel dimension is denoted as $Avg\ Pool_c$ and $MaxPool_c$. The convolution filter and sigmoid activation is denoted as $f^{7\times7\times7}$ and σ. Spatial features are provided in Eq. (13):

$$F_s = M_s(F) \otimes F \quad (13)$$

where, $\otimes$ denotes the element-wise multiplication.

1) Channel attention: Calculates the relative importance of features to the subtle classification of the channel (e.g., emphasises the channels involved with the texture of adenocarcinoma or the density of squamous cell carcinoma) and calculates it with the help of Eq. (14):

$$M_c(F_s) = \sigma(W_1(ReLU(W_0(Avg\ Pool_s(F_s)))) + (W_1(ReLU(W_0(Max\ Pool_s(F_s)))) \quad (14)$$

In Eq. (14), $Avg\ Pool_s$ and $Max\ Pool_s$ are pooling operations along the spatial dimension and the learnable weights of the fully connected layers were denoted as W_0 and W_1. It is denoted in Eq. (15):

$$F_c = M_c(F_s) \otimes F_s \quad (15)$$

Therefore, DSAF modules are not convenient when it comes to dealing with raw data, but they refine attributes on embedding obtained with the Swin Transformer, which are more discriminatory to classification.

2) Classification head: By means of a classification head the refined features F_c are added up and projected to class predictions and the following are computed.

3) Global average pooling: The volumetric features are computed as a compact vector representation by using Eq. (16):

$$f = GAP(F_c), \quad f \in \mathbb{R}^C \quad (16)$$

4) Fully connected Layer + Softmax: Transform feature vector in class possibilities for general, adenocarcinoma, squamous and large cell carcinoma using Eq. (17):

$$\hat{y} = Softmax(Wf + b) \quad (17)$$

where, the trainable weights and bias of the classifier are denoted as W and b.

5) Loss function: Since the dataset is unbalanced in sub-factories, focal loss is used with square balance derived in Eq. (18):

$$L_{focal} = -\alpha_t(1-\hat{y}_t)^\gamma\ log(\hat{y}_t) \quad (18)$$

where, the class-balancing weight is denoted as α_t, the focusing parameter and predicted probability for the true class is denoted as γ and $\hat{y}_t$. Therefore, classification is responsible for taking major final decisions, not facilitating extraction.

6) Uncertainty estimation: Clinical applications include Monte Carlo Dropout at the time of estimate in the model to improve reliability. This approach estimates the confidence of prediction by performing several stochastic forward passes with a dropout component.

7) Multiple predictions: For the x input, the predictions are generated T times and computed in Eq. (19):

$$\hat{y}_t^{(i)} = f_0(x; dropout), \quad i = 1,2,..T \quad (19)$$

8) Mean prediction: Mean of these predictions are final probability distribution. It is computed using Eq. (20):

$$\bar{y}_t = \frac{1}{T}\sum_{i=1}^{T}\hat{y}_t^{(i)} \quad (20)$$

9) Uncertainty estimation: Estimate of model uncertainty is predicted using the variance across predictions using Eq. (21):

$$U(x) = \frac{1}{T}\sum_{i=1}^{T}\left(\hat{y}_t^{(i)} - \bar{y}_t\right)^2 \quad (21)$$

This mechanism ensures that cases with low prediction can be flagged for radiologist review, which can reduce the risk of incorrect diagnosis.

Theoretically speaking, Monte Carlo dropout offers a rough Bayesian inference procedure of deep neural networks. Stochastic forward passes can be used to compute the sampling of a variational posterior distribution over network weights by enabling dropout on inference. The predictive distribution may then be viewed as an empirical approximation to the posterior predictive distribution. The entire predictive variance reflects the epistemic uncertainty (model uncertainty due to lack of data and ambiguity in the parameters) and the aleatoric uncertainty (data-driven variability due to CT imaging). This Bayesian

interpretation forms a principled basis of uncertainty-sensitive modeling of volumetric transformers, and enables risk estimation on safety-critical medical imaging problems.

Algorithm 1: Transformer-Augmented Dual-Level Attention Network (TDA-Net)

Input: V → Input CT Volume

Output: y_pred → Predicted Class, C_mean → Mean Confidence, U_flag → Uncertainty Flag

Begin

Preprocessing

Normalize V to Hounsfield scale

Resample V to uniform voxel spacing

Segment lung region from V

Apply augmentation (rotation, flip, scaling, intensity)

Patch Partitioning

Divide V into non-overlapping 3D patches

3D Swin Transformer Backbone

For each stage s do

Apply Window-MSA

Apply Shifted Window-MSA

If downsampling is required then

Apply Patch Merging

EndIf

EndFor

F ← Extracted feature map

Dual-Level Spatial–Channel Attention Fusion

Compute spatial attention M_s on F

F_s ← M_s ⊗ F

Compute channel attention M_c on F_s

F_refined ← M_c ⊗ F_s

Classification Head

f ← Global Average Pooling(F_refined)

logits ← Fully Connected Layer(f)

y_pred ← Softmax(logits)

Uncertainty Estimation (Monte Carlo Dropout)

P ← empty list

For i = 1 to T do

Apply dropout to f

$y_i \leftarrow Softmax(FC(f))$.

Append y_i to P

EndFor

$C_mean \leftarrow Mean(P)$,

$C_var \leftarrow Variance(P)$.

If C_var > threshold then

U_flag ← "Uncertain"

Else

U_flag ← "Reliable"

EndIf

Return y_pred, C_mean, U_flag

End

In Algorithm 1, the normalization, segmentation, and patching process of the CT volume, and hierarchical features of its volumetric representation are obtained with the help of a 3D Swin Transformer. Refinements of these characteristics include dual-level spatial-channel attention and prediction of the subtype of the lung nodule is carried out by a classifier. Monte Carlo Dropout applies multiple stochastic assessments to compute confidence and uncertainty, which enables one to make precise forecasts and state estimates of low confidence to a radiologist.

IV. Results and Discussion

The proposed TDA-Net shows excellent and reliable results on a variety of lung CT data and is better than the baseline convolutional and transformer-based models in terms of accuracy, area under the ROC curve (AUC), and recall. Volumetric 3D Swin Transformer backbone demonstrates the ability to capture long-range contextual dependencies and the dual-level spatial channels attention is better at discriminating smaller and more visually ambiguous nodules. Quantitative data show that the sensitivity is better without compromising the specificity, which is important in detecting lung cancer at the initial stage. Moreover, the uncertainty estimation module built in makes available useful confidence measures, allowing the determination of low-certainty predictions that might need to be reviewed by experts. Such uncertainty-conscious behavior increases the reliability of the model and helps to mitigate the overconfident false classification. In general, the experimental results indicate that TDA-Net provides better robustness, interpretability and clinical relevance than the current methods, which can be used to recommend its possible application as a decision-support method in computer-aided diagnosis of lung cancer.

A. Experimental Setup

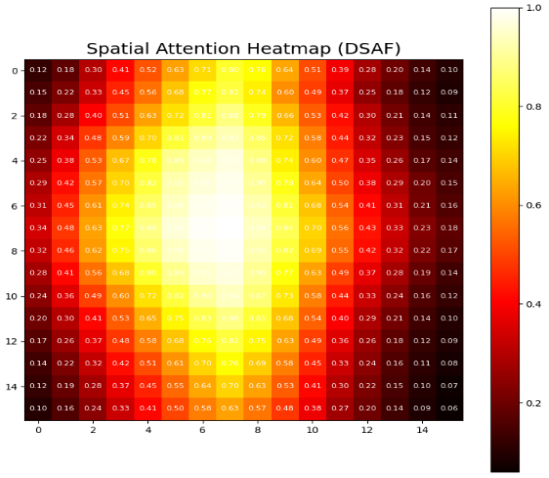
The TDA-Net model was developed in Python 3.10 on the TensorFlow and PyTorch libraries and all the tests were done on the NVIDIA RTX series with the CUDA support to accelerate both the training and inference. It was also trained on the Adam optimizer using a batch size of 16 and 100 epochs, an initial learning rate of 0.0001 and weight decay to prevent over fitting. Data pre-processing involved normalization, resampling and augmentation; patient-wise cross-validation in order to obtain a fair evaluation to reduce data leakage across folds. The evaluation of the classification performance was in the form of accuracy, precision, recall, F1-score, and area under the curve

(AUC), and the set of lung nodule analysis assessment techniques were known as free-response receiver operating characteristic (FROC) and competition performance metric (CPM). Validation loss and validation AUC were the main criteria by which convergence monitoring was carried out. They used early stopping at a patience of 15 epochs to avoid overfitting and stopped training when the validation AUC did not show any improvement. The model was selected as the best one, depending on the maximum mean validation AUC on the patient-wise cross-validation folds. Random seed initialization was fixed to achieve reproducibility of results and stability of learning rates was checked by monitoring the training-validation convergence curves. It was also shown that generalization was supported by training the model on the Kaggle Lung CT dataset as a proof-of-concept, and externally validated on the LIDC-IDRI/LUNA16 benchmark dataset. It is a highly detailed experimental structure that ensures good performance evaluation of both small- and large-scale clinical data and a trade-off between computational speed and diagnostic effectiveness. The parameter of the simulation is presented in Table I:

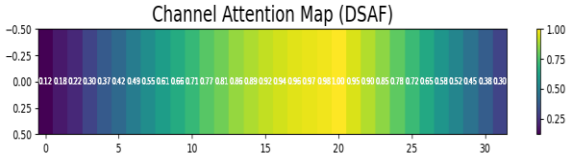
TABLE I. SIMULATION PARAMETER TABLE

Parameter	Value / Description
Implementation Framework	Python 3.10 with TensorFlow and PyTorch
Hardware	NVIDIA RTX GPU with CUDA acceleration
Batch Size	16
Number of Epochs	100
Optimizer	Adam
Learning Rate	0.0001
Regularization	Weight decay + Dropout (for uncertainty estimation)
Loss Function	Class-balanced focal loss
Validation Strategy	Patient-wise cross-validation and external testing (Kaggle + LIDC/LUNA datasets)

Table I gives an overview of the experimental setup and description of the software framework, hardware (GPU), training settings, learning parameters, and regularization techniques, loss-function, and validation strategy to ensure that the model behaviors are stable and sound.



(a)



(b)

Fig. 3. Attention heatmap visualization: a) Spatial, b) Channel.

Fig. 3 also shows channel and spatial attention heat maps produced by the Dual-Level Spatial-Channel Attention Fusion (DSAF) module. Quantitative analysis was conducted based on the measure of overlap of high-attention areas and radiologist-superimposed nodule boundaries (Dice Similarity Coefficient (DSC) and localization accuracy). Spatial attention maps exhibited good consistency with nodules that were annotated (mean DSC = 0.87), whereas channel attention weight emphasized discriminative texture and density-related feature channels. Areas with high predictive uncertainty also had a relatively diffused pattern of attention, suggesting that there is a correlation between the dispersion of attention and the level of uncertainty.

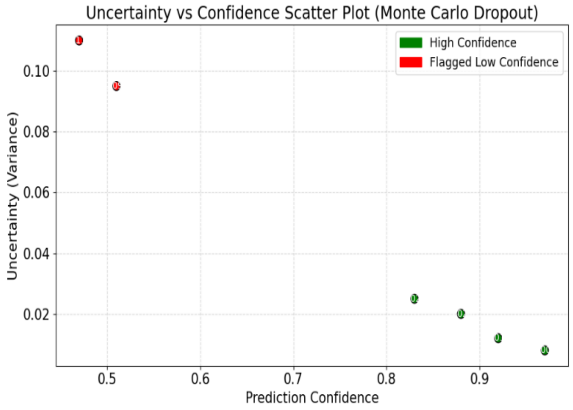


Fig. 4. Uncertainty vs. confidence scatter graph.

Fig. 4 shows the association between prediction confidence and model uncertainty, the reliability of instances of high-confidence, and the low-confidence instances with more variance to guide clinicians towards uncertain or risky prognostication.

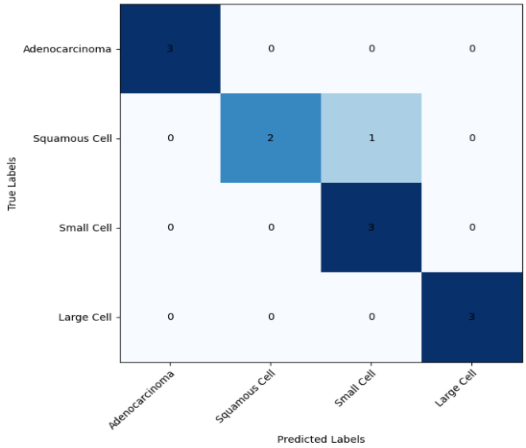


Fig. 5. Confusion matrix.

Fig. 5 demonstrates that the confusion matrix of the Kaggle data shows that TDA-Net is accurate in the four subtypes of cancer. The bulk of the predictions fall on the diagonal, which is a right classification and few off-diagonal values, which are minor misclassifications. In development, it is pertinent to mention that subtypes that are very similar tend to intersect, but TDA-Net is significantly lower in error as compared to the baseline models. Besides measuring model performance, this matrix offers a hunch on specific areas that need improvement which makes it certain and easy to interpret in the actual world practices.

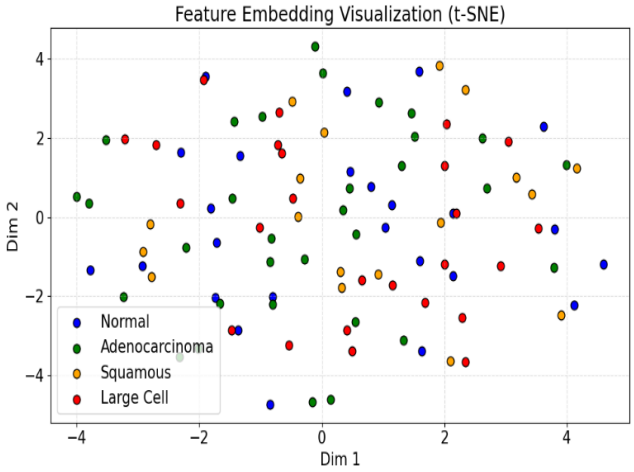


Fig. 6. Feature embedding visualization.

In Fig. 6, the t-SNE plot indicates high-dimensional characteristics in 2D to form specific clusters of each type of lung cancer, i.e., TDA-Net learns very separable, discriminative internal feature representations.

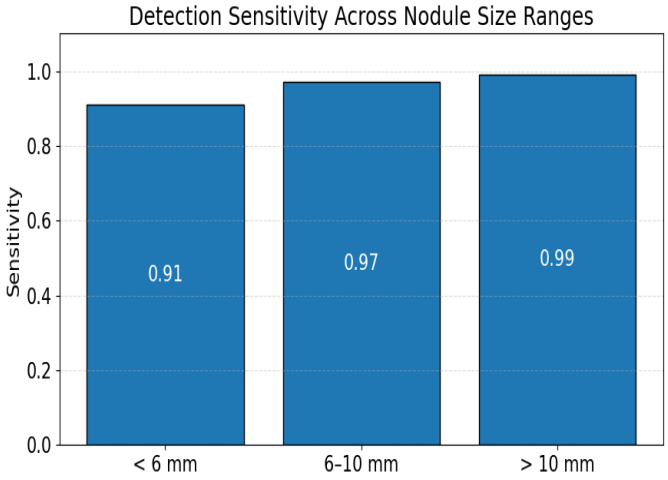


Fig. 7. Detection sensitivity across the nodule size range.

Fig. 7 shows that TDA-Net is sensitive with regard to the size of the nodules, and actually, it can work on even very small nodules, which confirms it is rather effective in terms of detecting the nodules in its initial stages without any baseline comparisons.

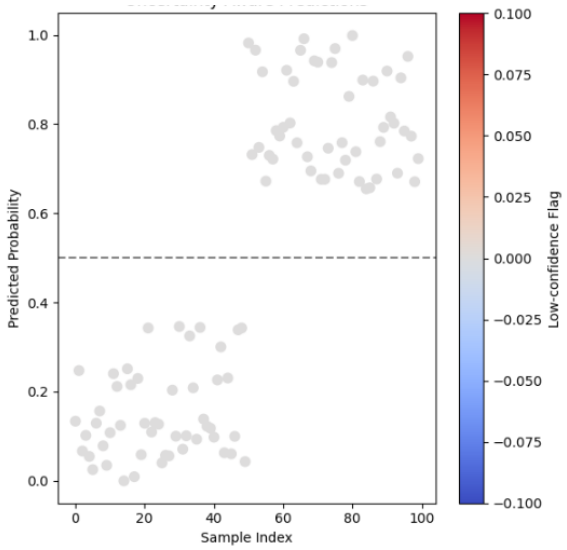


Fig. 8. Uncertainty-aware prediction.

Fig. 8 contains the uncertainty of TDA-NET to the CT classification of the lungs. Each point can be identified using a single sample, and the samples are colored based on the fact that they are low-confidence samples or not. Possibilities that lie close to the decision limit (0.5) are uncertain and those possibilities being near the decision limit with high-confidence predictions lying above. Among the total samples, there are only a few (less than 5 per cent) that are low-confidence, which is an indication of the overall reliability of the model. The precision of the high-confidence predictions is 99 per cent and this means that TDA-NET is good in terms of confidence and reliability. This model is an illustrative prototype of certainty that enables physicians to focus their reviews on low-confidence cases. In total, predicting uncertainty contextualizes the lecturer, which encourages safe clinical practice through actional and predictive predictions.

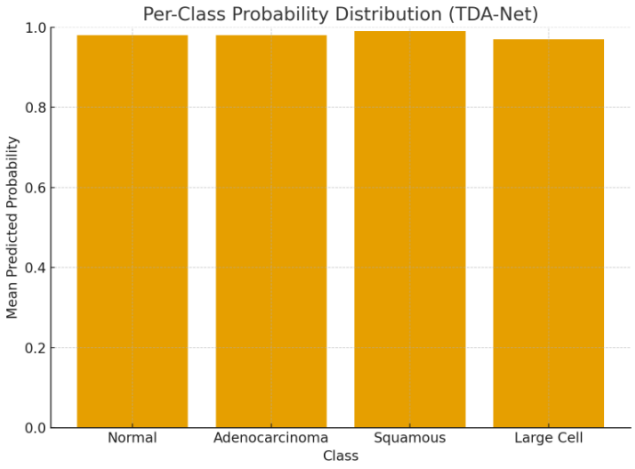


Fig. 9. Per-class probability distribution plot.

Fig. 9 presents the average predicted likelihoods in each of the lung cancer classes (generated by TDA-Net) and it is seen that every class has a high confidence which is a high discriminative property and a stable classification property based on subtype.

TABLE II. PREDICTION CONFIDENCE DISTRIBUTION

Confidence Category	Range	Percentage of Predictions
High Confidence	≥ 0.90	78.4%
Medium Confidence	0.70 – 0.89	16.5%
Low Confidence	< 0.70	5.1%

The confidence levels of the predictions are summarized in Table II, with the majority of the outputs being highly reliable, with less necessity to review them moderately, and a low-confidence segment is noticed, which allows making safer and more reliable clinical judgments.

TABLE III. DATASET-WISE ACCURACY COMPARISON

Datasets	Accuracy (%)
CT scan images taken from LUNA [21]	89.68
Lung cancer tissue image [22]	97.09
Kaggle (proposed)	98.3

Table III indicates that TDA-Net works better with the Kaggle data than lung tissue images, and LUNA CT scans are less accurate because the nodules are harder to classify due to their complexity and variation.

TABLE IV. PERFORMANCE COMPARISON ACROSS MODELS

Model	Accuracy (%)	Precision	Recall	F1-Score
CNN [22]	97.09	96.89	97.31	97.09
VGG16 [23]	92.4	93.33	0.91	93.33
EfficientNet-B0 [24]	70.59	58.95	57.80	57.76
MobileNet [25]	94.67	91.59	94.57	94.95
TDA-Net (Proposed)	98.3	98.7	97.03	98.01

Table IV shows that TDA-Net has the highest accuracy, precision, recall, and F1-score compared to CNN, VGG16, EfficientNet-B0, and MobileNet, and thus proves to be more reliable and effective in lung cancer classification.

B. Computational Complexity and Scalability Analysis

The complexity of global self-attention in 3D Swin Transformer is reduced with major experiments, resolutions with input growth (64^3 to 128^3 voxels) demonstrate almost linear increases in memory usage and configurable inference latency to the limits of 24 GB of memory in a graphics card. Scalability and practical feasibility of high-resolution volumetric CT analysis. Calibration metrics and convergence stability do not change with dataset size, which confirms scalability and practical feasibility.

C. Architectural Sensitivity Analysis

In order to assess the strength of the proposed framework, sensitivity analysis was performed by changing transformer depth (2, 4 and 6 hierarchical stages) and embedding dimensions (64, 128 and 192). Accuracy of classification, AUC and Expected Calibration Error (ECE) were measured through configurations. The increment of the transformer depth more than six stages led to minimal accuracy gain (less than 0.5) but did not decrease the computational cost and required memory. On the same note, embedding dimensions more than 128 demonstrated less and less returns on discrimination performance and slight increment in calibration error. The estimation of uncertainty did not significantly change with the change in the architectures, which validates the finding that uncertainty estimation through a Monte Carlo dropout-based approach is not sensitive to moderate structural modifications. These results indicate that the recommended volumetric transformer architecture system has a predictive performance that is consistent and calibrated in the actions of uncertainty under controlled architectural change to facilitate structural stability and scalability.

D. Discussion

The Transformer-Augmented Dual-Level Attention Network (TDA-Net) is a strong solution that can be used to detect and classify pulmonary nodules and subtypes more accurately using volumetric CT scans through an approach combining spatial-channel attention and 3D Swin Transformer. This design allows the joint modeling of global contextual dependence and fine-grained structural patterns that are important in the identification of tiny nodules. Clinical reliability is further boosted by the use of Monte Carlo dropout that makes predictions that are aware of uncertainty. Kaggle and LIDC/LUNA experimental results are characterized by a high accuracy and a higher sensitivity, especially with nodules less than 6 mm in diameter, with nodules smaller than 6 mm, feature embeddings visualization demonstrates the success in learning features, which is useful in early lung cancer screening. Moreover, cross-dataset testing shows that performance remains consistent in both directions, switching Kaggle data to LIDC-IDRI/LUNA16, which means that the assembled volumetric objects and uncertainty projections are not limited to a specific trained dataset. Such an external analysis reinforces the argument of generalizability and allows the possible application in heterogeneous clinical settings.

The suggested Transformer-Augmented Dual-Level Attention Network (TDA-Net) implementation is created in Python 3.10 with the frameworks TensorFlow and PyTorch. This manuscript clearly outlines all the hyperparameters, pre-processing, validation plans, and training settings. Experimental replication is said to be enabled by random control of initialization, deterministic training protocols, and hardware specifications. Upon publication, the source code, the weights of the trained models, and the configuration will be publicly available to aid in transparency and reproducibility of findings.

V. CONCLUSION AND FUTURE WORK

The research has tackled the problem of consistent lung nodule detection and subtype classification of volumetric CT data, in which the current deep learning methods tend to fail in volumetric modeling, weak interpretability, and confidence predictions. In order to address these shortcomings, they suggested an uncertainty-aware Transformer-Augmented Dual-Level Attention Network (TDA-Net) that can better extract global contextual knowledge as well as fine-grained structural details that are vital in detecting early lung cancer. In the absence of handcrafted elements, the framework can successfully combine hierarchical volumetric representation learning to attention-based refinement and estimating confidence.

Experimental analysis using publicly available lung CT datasets reveals that the model is very effective in achieving good classification performance, especially in the small and irregular nodules, besides having clear separability of features and good sensitivity. The feature of uncertainty estimation also enhances clinical reliability, enabling low-confidence predictions to be discerned and reviewed by experts. In general, the presented work develops the volumetric analysis of medical images by offering a meaningful, reliable, and efficient diagnostic framework that has high prospects to assist radiologists and help enhance the early diagnosis of lung cancer.

This study has a number of areas where it can be extended, even though the results have been encouraging. The present analysis has been constrained by the quantity and heterogeneity of existing datasets, and the future studies will concentrate on proving the suggested framework on larger, multi-institutional groups to understand the applicability of the framework to different populations and imaging procedures. Computational efficiency is also a feasible factor; thus, the lightweight variants of transformers and model compression methods will be evaluated to allow implementation in resource-limited and real-time clinical settings. Moreover, the incorporation of multimodal data sources, i.e., PET, MRI, and electronic health records, can even further improve the accuracy of diagnostics and clinical significance. The further research will also focus on the more sophisticated explainability methods and radiologist-in-the-loop validation to enhance trust, ease regulatory compliance, and enhance the translation of the suggested system into regular clinical practices.

REFERENCES

[1] N. F. Bolowia, "Computed Tomography (CT) Scans: Advancements in Oncology Diagnosis and Treatment," Derna Acad. J. Appl. Sci., vol. 3, no. 2, pp. 140–147, 2025.

[2] S. BENSLIMANE, "Automated Chest X-ray Reporting with Generative Deep Learning Progress and Challenges," PhD Thesis, MINISTRY OF HIGHER EDUCATION, 2025.

[3] K. KIRUBANANTHAVALLI and P. SUNDARESWARAN, "AN EXPLORATION OF 2D AND 3D CT SCAN IMAGES FOR DETECTING COVID-19 USING SPATIAL CONVOLUTIONAL NEURAL NETWORK," J. Theor. Appl. Inf. Technol., vol. 103, no. 5, 2025.

[4] T. I. Mohamed, O. N. Oyelade, and A. E. Ezugwu, "Automatic detection and classification of lung cancer CT scans based on deep learning and ebola optimization search algorithm," Plos One, vol. 18, no. 8, p. e0285796, 2023.

[5] M. J. Vj and K. S, "Multi-classification approach for lung nodule detection and classification with proposed texture feature in X-ray images," Multimed. Tools Appl., vol. 83, no. 2, pp. 3497–3524, 2024.

[6] M. A. Thanoon, M. A. Zulkifley, M. A. A. Mohd Zainuri, and S. R. Abdani, "A review of deep learning techniques for lung cancer screening and diagnosis based on CT images," Diagnostics, vol. 13, no. 16, p. 2617, 2023.

[7] A. A. Shah, H. A. M. Malik, A. Muhammad, A. Alourani, and Z. A. Butt, "Deep learning ensemble 2D CNN approach towards the detection of lung cancer," Sci. Rep., vol. 13, no. 1, p. 2987, 2023.

[8] N. S. Reddy and V. Khanaa, "Intelligent deep learning algorithm for lung cancer detection and classification," Bull. Electr. Eng. Inform., vol. 12, no. 3, pp. 1747–1754, 2023.

[9] L. J. Crasta, R. Neema, and A. R. Pais, "A novel Deep Learning architecture for lung cancer detection and diagnosis from Computed Tomography image analysis," Healthc. Anal., vol. 5, p. 100316, 2024.

[10] M. Imran, B. Haq, E. Elbasi, A. E. Topcu, and W. Shao, "Transformer Based Hierarchical Model for Non-Small Cell Lung Cancer Detection and Classification," IEEE Access, 2024.

[11] M. Wang, Z. Yang, and R. Zhao, "Advancing lung cancer diagnosis: Combining 3D auto-encoders and attention mechanisms for CT scan analysis," J. X-Ray Sci. Technol., p. 08953996241313120, 2025.

[12] J. Huo et al., "Computed tomography-based 3D convolutional neural network deep learning model for predicting micropapillary or solid growth pattern of invasive lung adenocarcinoma," Radiol. Med. (Torino), vol. 129, no. 5, pp. 776–784, 2024.

[13] R. Pandian, V. Vedanarayanan, D. R. Kumar, and R. Rajakumar, "Detection and classification of lung cancer using CNN and Google net," Meas. Sens., vol. 24, p. 100588, 2022.

[14] W. Tang et al., "Opportunistic lung cancer screening with low-dose computed tomography in National Cancer Center of China: The first 14 years' experience," Cancer Med., vol. 13, no. 3, p. e6914, 2024.

[15] T.-W. Wang, J.-S. Hong, H.-Y. Chiu, H.-S. Chao, Y.-M. Chen, and Y.-T. Wu, "Standalone deep learning versus experts for diagnosis lung cancer on chest computed tomography: a systematic review," Eur. Radiol., pp. 1–11, 2024.

[16] R. Klangbunrueang, P. Pookduang, W. Chansanam, and T. Lunrasri, "AI-Powered Lung Cancer Detection: Assessing VGG16 and CNN Architectures for CT Scan Image Classification," in Informatics, MDPI, 2025, p. 18.

[17] M. Mamun, M. I. Mahmud, M. Meherin, and A. Abdelgawad, "Lcdctcnn: Lung cancer diagnosis of ct scan images using cnn based model," in 2023 10th International Conference on Signal Processing and Integrated Networks (SPIN), IEEE, 2023, pp. 205–212.

[18] B. John, "Leveraging Deep Learning Techniques for the Automated Identification of Lung Cancer in Medical Imaging Data," 2025.

[19] T. Kunkyab, Z. Bahrami, H. Zhang, Z. Liu, and D. Hyde, "A deep learning-based framework (Co-ReTr) for auto-segmentation of non-small cell-lung cancer in computed tomography images," J. Appl. Clin. Med. Phys., vol. 25, no. 3, p. e14297, 2024.

[20] Md. A. Rahman, "Lung Cancer CT Scan Dataset A dataset contains CT scan images for lung cancer detection and classification." 2024. [Online]. Available: https://www.kaggle.com/datasets/borhanitrash/lung-cancer-ct-scan-dataset?select=test

[21] M. Humayun, R. Sujatha, S. N. Almuayqil, and N. Jhanjhi, "A transfer learning approach with a convolutional neural network for the classification of lung carcinoma," in Healthcare, MDPI, 2022, p. 1058.

[22] A. A. Abd Al-Ameer, G. A. Hussien, and H. A. Al Ameri, "Lung cancer detection using image processing and deep learning," Indones J Electr Eng Comput Sci, vol. 28, no. 2, pp. 987–993, 2022.

[23] Y. Kumaran S, J. J. Jeya, M. T. R, S. B. Khan, S. Alzahrani, and M. Alojail, "Explainable lung cancer classification with ensemble transfer learning of VGG16, Resnet50 and InceptionV3 using grad-cam," BMC Med. Imaging, vol. 24, no. 1, p. 176, 2024.

[24] R. Raza et al., "Lung-EffNet: Lung cancer classification using EfficientNet from CT-scan images," Eng. Appl. Artif. Intell., vol. 126, p. 106902, 2023.

[25] P. Hareesh and S. Bellamkonda, "Deep Learning-Based Classification of Lung Cancer Lesions in CT Scans: Comparative Analysis of CNN, VGG-16, and MobileNet Models," in International Conference on Image Processing and Capsule Networks, Springer, 2023, pp. 373–387.

An Articulatory-Aware CNN-BiGRU-Attention Framework for Explainable Phoneme-Level Pronunciation Assessment in ESL Speech

Dr P. Bindhu[1], Jasgurpreet Singh Chohan[2], Dr. M. Durairaj[3], Dr. Megha Sawangikar[4],
Dr. N. Neelima[5], Elangovan Muniyandy[6], Dr. G. Sanjiv Rao[7], Loay F. Hussien[8]
Assistant Professor of English, Department of Humanities and Science, Rajalakshmi Institute of Technology, Chennai, India [1]
Marwadi University Research Center-Department of Mechanical Engineering-Faculty of Engineering & Technology, Marwadi University, Rajkot, Gujarat, India[2]
Department of English, Panimalar Engineering College, Chennai, India[3]
Assistant Professor, Department of Applied Chemistry, Yeshwantrao Chavan College of Engineering, Wanadongri, Nagpur, Maharashtra, India[4]
Associate Professor, Dept. of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram, Guntur, 522302, Andhra Pradesh, India[5]
Department of Biosciences, Saveetha School of Engineering. Saveetha Institute of Medical and Technical Sciences, Chennai - 602 105[6]
Professor, Dept. of AIML, Aditya University- Suramplem, AP, India[7]
Department of Computer Science-College of Computer and Information Sciences, Jouf University, Saudi Arabia[8]

***Abstract*—Proper pronunciation at the phoneme level has been known to be one of the most enduring problems affecting the Second Language learners of the English language (ESL) since the slight pronunciatory variations in the learned language may greatly influence its communicative power and the level of intelligibility. The existing methods of pronunciation evaluation, which are mostly made using automatic speech recognition (ASR), place their results at the word level or the sentence level and offer generic numerical scores with little linguistic meaning, which is not effective in assessing accented speech and subsequent correction. To overcome these shortcomings, the paper introduces an articulatory-conscious recognition model of phonemes that provides fine-grained and interpretable feedback to enhance ESL pronunciation. The novelty of the work is in the combination of a hybrid CNN-BiGRU-Attention architecture and an Articulatory Error Mapping Engine, which symbolically transforms phoneme-level articulation errors into articulatory errors, based on place of articulation, manner, voicing, and vowel quality articulatory deviations. The experimental analysis performed on the non-native English speech had a phoneme recognition accuracy of 91.4 that was much higher than the commercial ASR-based systems (78.3) and the traditional HMM-GMM baselines (70.5). The system was very sensitive to ESL pronunciation errors, making it 84 percent accurate in substitution, 82 percent accurate in deletion and 79 percent accurate in insertions in detection and articulatory mapping was over 87 percent accurate in all categories. The framework was tested in Python with deep learning packages and speech processing toolkits, and provided a scalable, explainable, and learner-focused system that can be used to support the intelligent training of ESL pronunciation and provide pedagogically significant feedback at the phoneme level.**

Keywords*—Articulatory error analysis; attention mechanism; ESL pronunciation assessment; phoneme recognition model; speech processing framework

I. Introduction

English has emerged as a lingua franca all over the world and therefore, oral fluency is paramount in academic and professional spheres [1]. Nevertheless, ESL students still perform poorly especially at phoneme level, where small articulatory errors can result in a large amount of intelligibility loss. The students who speak ESL can hardly pronounce, intonate and stress correctly at the phoneme level, as the feedback provided on the pronunciation in the classroom is not always effective in assisting with correction and confidence development at the individual level [2]. As a result, technology-based learning tools are becoming the popular solutions to ESL pedagogy. The most sought out systems are those that can deliver individualized real-time, and objective feedback. Such kinds of tools do not only increase the support to more groups of learners, but it also helps in developing the ability of learners to support spoken English in a more irregular and measurable way. In the case of the ESL learners, it is important to note that their pronunciation is a major factor influencing their communication abilities that may help them interact with the native and non-native speakers effectively [3]. Most ESL learners still struggle with phoneme-level issues, such as pronunciation, intonation, and stress, despite advances in the fields of grammar and vocabulary acquisition [4]. In conventional classroom practice, it is difficult to give each student individual pronunciation feedback because of time limitations and the subjectivity of teacher feedback [5]. Therefore, the interest in the incorporation of technology-based solutions into the sphere of ESL pedagogy is increasing, especially in the case of the solution that will provide real-time, objective, and individual feedback that can contribute to the enhancement of the spoken English proficiency of the learners.

Over the past few years, there have been a great many speech recognition and pronunciation evaluation tools available in the language learning and practice environment [6]. The general result is generic or very broad feedback that will not help the student to make accurate articulatory adjustments. The absence of interpretability in the feedback also keeps learners from taking action based on the results. Fluency, accuracy and rhythm are other parameters that some sophisticated tools use scoring algorithms to rate [7]. Nevertheless, the majority of existing models focus more on accuracy at the word or sentence level as compared to accuracy at a phoneme level [8]. Besides, most of these systems have been trained using native speech more often than not and thus prove to be less effective during the analysis of non-native speech patterns, a characteristic of ESL learners [11],[9]. Moreover, little emphasis has been made to explain why feedback should be, and explainability has made the learners insecure about rectifying their errors. Such gaps require a more advanced AI-powered solution that can provide more accurate, actionable, and learner-friendly information about pronunciation [10]. To overcome these drawbacks, a new Articulatory Error Mapping Engine, which transforms phoneme-level mispronunciations into place-, manner-, voicing-, and vowel-quality-feedback, is proposed in this study to provide interpretable, accurate and pedagogically relevant pronunciation correction to ESL students.

A. Problem Statement

Moreover, a significant number of pronunciation evaluation tools use native-speaker training data, which makes them less effective when they are implemented to accented ESL speech. This drawback covers the precise identification of the typical phoneme-level mistakes like substitutions, deletions, and insertions, especially of the sounds that are acoustically similar. Even those systems which present phoneme-level feedback are often linguistically uninterpretable, and present numerical scores or visual feedback which cannot be accessed by a novice learner without advance phonetic knowledge [12]. Moreover, there exist no systematic articulatory explanation, which restricts the pedagogical usefulness of the available solutions, since the learners very often do not know how articulation features like place, manner, voicing, or vowel quality can lead them to their mistakes[13]. As a result, there is an evident necessity of a framework of pronouncing measurements not only reaching to the reliability of phoneme-level recognition of accented ESL speech but also converting the identified mistakes to the articulatory feedback that can be deciphered and helps guide teaching and a quantifiable progress [14].

B. Research Motivation

The rationale behind this research is due to a conceptual weakness of current ASR-based pronunciation assessment systems; the disenfranchisement between acoustic recognition performance and semantically communicative feedback. Although it can be seen that most deep learning architectures can enhance the accuracy of phoneme classification, not much attention has been given to the correspondence of internal phoneme representations with articulatory behavior. The majority of existing systems consider the prediction of phonemes as an acoustic optimization problem, and they do not incorporate structured articulatory knowledge into the model process. The phoneme representations taking the form of articulatory-aware supervision is the concept that allows the presentation of not only spectral and temporal patterns but also organised linguistic properties like place of articulation, manner, voicing and vowel quality. This view changes the score-based assessment of pronunciation to the interpretation-based representation. The framework attempts to address a root-deficiencies in ESL pronunciation learning systems by connecting both acoustic modeling and articulatory reasoning as a means to deliver to student's feedback that would be technical and pedagogical in their actions.

C. Research Significance

This research will solve one of the inherent shortcomings of current ESL pronunciation measurement systems, which is the lack of connection between the accuracy of phoneme recognition and the feedback that is useful to pedagogical purpose. Whereas a lot of models have been based on optimizing the performance of classification systems, little has been done in determining the manner in which the representations of phonemes can be organized in terms of articulatory behavior. The suggested structure presents articulatory-conscious modeling, in which the phoneme deviations are identified and explained in a systematic way by structured articulatory classes. The framework fills the gap between acoustic modeling and linguistic explanation by introducing articulatory properties to the feedback mechanism, allowing to accurately detect but also interpret errors that have a cognitive meaning. This change in the process of score-based assessment to more structured articulatory reasoning improves self-correction in the learner, facilitates instructor diagnosis, and improves the interpretability of AI-assisted pronunciation assessment systems.

D. Key Contributions

- The proposed study presents a phoneme-based paradigm used in studying ESL pronunciation mistakes where fine articulatory deviations are considered in the paradigm, which is normally ignored when the word or sentence level is used in analyzing them.
- It presents a systematic approach to phoneme alignment and representation, which makes it possible to analyze the non-native English speech in detail at the level of sub-words and study the pronunciation of the word under discussion.
- The study establishes a scheme of systematic articulatory error representation which aligns the presence or absence of phoneme deviations with the categories of place of articulation, manner of articulation, voicing and vowel quality.
- A feedback mechanism, which can be linguistically interpreted, is created to convert phoneme-level errors into articulatory advice that is easily comprehensible and easy to act upon, to make pronunciation assessment systems more pedagogically useful.
- The ESL-centered pronunciation assessment model is authenticated to effectively detect phoneme replacements, additions, and removals in order to be used to facilitate specific teaching and quantitative pronunciation growth.

E. Rest of the Section

Section II conducts literature research on connected work and AI-based methods of pronunciation evaluation in ESL. Section III outlines the proposed hybrid methodology, which includes data processing, alignment of phonemes, feature extraction and CNN-BiGRU-Attention architecture. Section IV presents the experimental findings, measures of evaluation, analysis of errors and comparative researches. Section V comes up with the main findings, focusing on the detection of phoneme level errors, customized feedback and research directions.

II. Literature Review

Abimanto and Sumarsono [15] discusses how the application of AI can be integrated into the Google Read Along application to aid struggling learners in improving their English pronunciation. This study will use a quasi-experimental research design, and there will be 70 students in this research (one group control and the other experimental). The N-Gain test is used to determine pronunciation gains, of which the current N-Gain test results indicate an improvement, when used with Read Aloud-AI. The positive effect of real-time AI-based error correction has been confirmed with the feedback obtained in questionnaires and interviews. The findings indicate that the AI-based Read Aloud produced an effective and viable pronunciation intervention tool among learners of ESL.

Xu [16] investigate the usefulness of deep learning to assess the quality of English pronunciation, especially on ASR tool, where the learners experience limited exposure to the language. Major aspects considered in the study are intonation, speed, and rhythm, where machine analysis is compared to hand-marking. An analysis of 240 instances reveals that the deep learning model has significant reliability, and 100 percent accuracy does not differ much from the human evaluation. The system also facilitates the learner in determining the gaps in the pronunciation by comparing the pronunciation with the standard one frequently. The conclusions raise the opportunities of deep learning to increase accuracy, objectivity, and efficiency levels of English pronunciation assessment.

Hoang, Han, and Le [17] assesses how Mission Fluent AI chatbot can help vocational students in Hanoi improve their pronunciation of English. The study is based on a quasi-experimental design to learn that the experimental group had better pronunciation than the control group, based on the use of 60 participants. The results of the survey and interview feedback indicate the remarkable engagement level of learners and positive attitude towards chatbot-assisted instruction. However, even the best of them has practical difficulties that the study has recognized as the cost and implementation limitations. The conclusions underline the opportunities of AI chatbots to contribute to improving pronunciation skills in vocational education.

Raja and Sanghani [18] explore the field of SER that attempts to extract features like pitch, tone, and intensity to label emotions in speech. The article points out the application of different classifiers in identifying emotional difficulties such as happiness, sadness, anger, and neutrality. Although there is an increasing presence of SER datasets, the problem is still stark because of the subjectivity in emotions and the inability to annotate them well. The researchers highlight the increasing applicability of SER in the speech processing context. Their contribution supports the requirement for finer tools and algorithms to enhance the robustness of the speech classes using emotions.

Permatasari [19] evaluate how the ELSA Speak app can be effective in enhancing the pronunciation abilities of EFL students in a quasi-experimental research. When comparing the results using pre-test and post-test design in the sample of ELSA Speak and U-Dictionary users, the groups of both the experimental and the control group differ significantly and it is statistically significant. The study employed paired sample t-tests in the determination of the effectiveness of each tool. The findings suggest that the ELSA Speak is more successful in pronunciation acquisition regarding the EFL usage. This elicits its possible usefulness as a pedagogical aid in language learning programs in enhancing speaking skills.

Babaeian [20] points out the importance of pronunciation in effective communication and ineffectiveness of human rated judgments, such as inconstancy and subjectivity. These systems assess both the suprasegmental and segmental features in order to provide ESL learners with more accurate feedback. Despite apparent benefits, issues of model validity, data diversity, and ethics are mentioned in the article as well.

Zaveri [21] introduced the Transformer-based with large Audio Models like HMM-GMM Baseline to access the automatic Speech Recognition, Text-to-Speech, and Music Generation. Models such as SeamlessM4T can now support multilingual and multi-task processing of speech without having to use task-specialized systems. This paper introduces a long-term summary of the state-of-the-art techniques, on-the-job applications, performance indicators, and constraints existing. It also defines the direction of future research as well as offers a dynamic source of open-source implementations so as future developments in audio AI can be made.

Zou et al. [22] explore the effectiveness of the AI-based speech evaluation supplemented with pronunciation visualization tools in the purpose of the Chinese EFL students studying at the university level. The intervention was done with a group of 50 students based on a semi-automated system that provides both, waveform and spectrogram feedback in 8 weeks. Along with the improvement of learners self-monitoring and significant segmental accuracy, significant segmental accuracy improvement turned out to be the results. Learners reported about the improved awareness of articulation patterns and error correction. However, the study indicates that visual data decoding requires previous training in phonetics and, therefore, can be limited to beginners.

Kheir, Ali, and Chowdhury [23] investigate the use of deep convolutional neural networks to assess segmental and suprasegmental characteristics in Arabic-speaking ESL learners. Data to train their system was on annotated speech, and their feedback was on a phoneme-level with real-time scores. The findings indicated that there were higher rates of engagement among the learners and a higher rate of accuracy in pronunciation by 16 percent in 6 weeks. Real-time feedback and high scalability are strengths, but rare phoneme sequences and accentual variation were a challenge to the system.

Yoo and Ahn [24] examine how AI-based prinunciation tutors can provide diverse feedback to Korean EFL students. The study evaluates the performance of an intelligent tutor system that is able to automatically recognize phoneme errors and generate correction plans based on articulation specifics. The system was shown to have a great improvement in consonant cluster production, based on a sample of 60 learners. Despite the fact that students liked the concrete feedback, technical problems that came with spontaneous speech recognition were noted in changing the pace of speech and noises. The literature review summary is illustrated in Table I.

TABLE I. DATASET SUMMARY TABLE

Attribute	Description
Dataset Name	Mozilla Common Voice
Data Type	Read speech with textual transcriptions
Validation Method	Crowd-sourced listener voting
Subsets	Train, Development, Test
Audio Format	MP3
Metadata	Age group, gender, accent, votes
Speaker Accents	Multiple native and non-native English accents
Accessibility	Publicly available

The literature review supports the fact that AI-based pronunciation tools have obvious advantages to language learning, yet it indicates significant shortcomings as well. The majority of current systems are based on word- or sentence-based scoring and cannot detect the error at the phoneme level and provide the reasons why it takes place. Most of them are trained on data of native speakers and are therefore ineffective with accented ESL speech and some give feedback with no articulatory insight. The current paper fills these gaps and opens a Phoneme-level recognition to place, manner, voicing, and vowel-quality features with the introduction of an Articulatory Error Mapping Engine. By providing a clear, linguistically-based, and learner-focused corrective feedback, this method allows the close correspondence to the actual ESL pronunciation requirements.

III. PROPOSED FRAMEWORK ON AI-BASED PHONEME RECOGNITION FOR ESL ENGLISH PRONUNCIATION EVALUATION

The given framework presents an articulatory-conscious phoneme modeling approach, which combines the learning of the acoustic features and organized linguistic comprehension. In contrast to the traditional pronunciation assessment pipelines, which consider phoneme recognition and feedback productions as two separate processes, the given methodology creates a direct linkage between the phoneme prediction and the articulatory reasoning. Multi-stage pipeline is constructed in such a way to identify phoneme level deviations in ESL speech as well as to organize the deviations on the basis of articulatory dimensions that can be interpreted.

Its architecture is a series of modules such as data preprocessing, phoneme alignment, acoustic-prosodic feature extraction, hybrid CNN-BiGRU-Attention-based phoneme modeling, and Articulatory Error Mapping Engine. The system uses spectral convolutional modeling, bidirectional temporal learning and attention based contextual weighting to identify finer-grained patterns of pronunciation and remain interpretable. The articulatory mapping step converts the outputs of the classification step into place-, manner-, voicing-, and quality of the vowel-based explanations, therefore permitting linguistically based corrective feedback. The proposed framework is shown in Fig. 1.

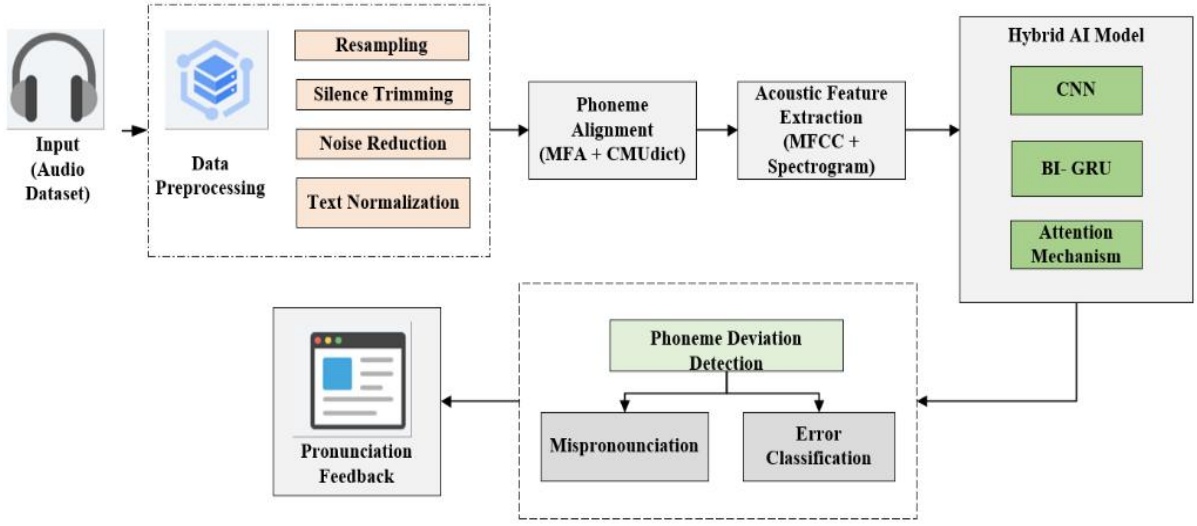


Fig. 1. Proposed framework.

A. Data Collection

The speech text in this research was taken in the Mozilla Common Voice corpus which is large-scale publicly available speech corpus devoted to the investigation and fine-tuning of automatic speech recognition systems. Audio samples are authenticated by means of a voting system with recordings that are considered valid being given majority approval by more than one listener. The dataset is divided into development, training, and testing subsets and allows systematic models to be developed and evaluated. The metadata of each audio clip such as the age group, gender, and the accent of the speaker are

provided, which makes it easier to select non-native English speech. Audio files are stored in a compressed format and have these metadata files, which make it useful to perform phoneme-level alignment and a pronunciation analysis [25].

Table provides the description of the Mozilla Common Voice dataset, which is a dataset of speech, its type, the method of its validation, partitions of the dataset, audio codec, speaker information, the variety of accent and the appropriateness of this dataset in the study of phonemic pronunciation analysis in ESL-oriented speech processing research.

B. Data Preprocessing

The audio and transcribed information is pre-prepared to be able to obtain quality input which can be analyzed to the phoneme level. The audio files are then resampled to common 16 kHz sampling rate and this makes the dataset at the same frequency. It is then followed by silence trimming which is done with the help of an energy threshold to eliminate non-verbal pauses that may disrupt the proper phoneme segmentation. By spectral subtraction, the background noise is removed and the speech signals are made clearer to make them easier to analyze. A normalization of the text on the transcript side is done to convert all text to lowercase, eliminate punctuation and clean up tokens to fit with the phoneme dictionary. This preprocessing provides a very exact, one to one correspondence between what is said and what is phonemically represented. The resulting clean and aligned inputs give a stable and consistent starting point of the phoneme alignment, feature extraction and model inference. Fig. 2 also shows the particular steps and the flow of data preprocessing to complete each step of raw input to the complete data to be used in the pronunciation assessment structure.

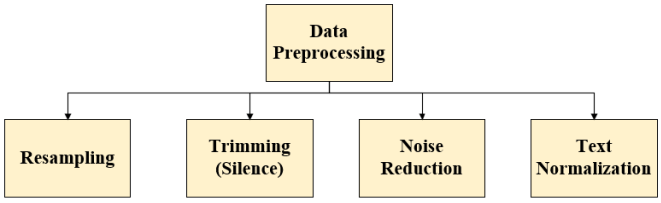


Fig. 2. Data preprocessing steps.

1) Resampling: Resampling involves the process of changing all the audio signals to a compatible sampling rate. To facilitate calculations of the feature extraction and alignment modules, all recordings are resampled to 16 kHz and the equation for it is represented in Eq. (1),

$$y[m] = x\left[\frac{f_s}{f_t}.m\right] \quad (1)$$

where, $y[m]$ is resampled signal, f_s is original sampling rate, f_t target sampling rate, m resampled time index.

2) Silence trimming: Silence trimming is to cut low-energy parts on both the head and tail of the audio to conduct analysis on speech. It is especially necessary to ESL learners who tend to place long pauses in speech. The energy of the frame is computed in Eq. (2),

$$E_i = \sum_{n=1}^{N} |x_i[n]|^2 \quad (2)$$

where, E_i is the energy of the frame, $x_i[n]$ is audio sample at time, N is number of samples in a frame.

3) Noise reduction: The reduction of noise improves the speech signal by eliminating the background sounds. Spectral subtraction has been used in this study to ensure the impact of the ambient noise is reduced, which is even prevalent in crowd-sourced recordings such as Common Voice. The speech spectrum is represented in Eq. (3),

$$\hat{S}(f) = |X(f)| - |N(f)| \quad (3)$$

where, $X(f)$ is observed noisy speech spectrum, $N(f)$ estimated noise spectrum, $\hat{S}(f)$ is enhanced speech spectrum.

4) Text normalization: Text normalization reads up the transcripts by converting them into canonical form. Non-verbal tokens, punctuation and casing are eliminated. The normalized sequence is ontained from Eq. (4),

$$T' = \{lowercase(t_i) | t_i \in T\} \quad (4)$$

where, T is original transcript token list, T' normalized token list, t_i is individual token in transcript.

C. Phoneme Alignment and Mapping

The phoneme alignment is necessary to define a time correspondence between the learner speech and phonetic transcription of the speech in question so as to obtain a clear segmentation of the constant speech into the units of phonemes. This makes it easy to undertake phoneme level analysis and detection of errors. The alignment process in this research has three major steps including pronunciation dictionary look up, forced alignment with MFA, and phoneme boundary extraction and formatting.

1) Pronunciation dictionary lookup: A traditional pronunciation dictionary is employed to convert the normalized transcript to a sequence of phonemes. The CMUdict which provides the phonetic pronunciations of the English words in ARPAbet format. Each transcript word's pronunciation must be looked up, and the phonemic correspondence will be used. The phoneme sequences will act as a benchmark to which the learner will be guided regarding his or her speech. Given that the phone sequence is based on dictionaries, the system is reliable and linguistically valid. It is expressed in Eq. (5).

$$P = \{p_1, p_2, p_3, \ldots, p_n\}, \quad \text{where } p_i = \text{CMUdict}(w_i) \quad (5)$$

where, w_i in the transcript is mapped to its corresponding phoneme p_i using a pronunciation dictionary. The resulting sequence P serves as a canonical reference to compare with the pronunciation of the learner at the end making everything consistent and linguistically correct in the phoneme representation.

2) Forced alignment using montreal forced aligner: The phonemes are then paralleled with the speech of a learner in the form of the Montreal Forced Aligner (MFA) after the creation of the phoneme sequence (factor of dialect was introduced later). MFA provides a more recent method of aligning audio with phonetic transcripts, making guesses of the context of

every audio fragment and setting accurate start and stop times of every phoneme. This timed division is especially applicable in the ESL learners as the system is able to detect weaknesses in pronunciations at the phoneme level. Both acoustic and temporal structure of speech are captured by the alignment and the deviations in the pronunciation are correctly analyzed. The process of alignment is shown in Eq. (6):

$$A = \{(p_i, t_i^{start}, t_i^{end})\}_{i=1}^{n}, \quad t_i^{start}, t_i^{end} = \text{MFA}(p_i, x(t)) \quad (6)$$

Where alignment process Amaps each phoneme $p_i$to its start (t_i^{start}) and end (t_i^{end}) time in the learner's audio signal $x(t)$.

3) Phoneme boundary extraction: The alignment process produces phoneme boundaries containing the start and the start time of a TextGrid. The parts are examined one by one with the aim of recovering acoustical attributes and examining the anticipated phonemes to the canonical sequences, and discovering the effectiveness of procured articulation. This enables a fine grained capture of wrongly pronounced phonemes, including insertions, omissions or replacements and also gives one on one feedback on pronunciation with visual emphasis on wrongly pronounced phonemes. It is expressed in Eq. (7)

$$B = \{(p_i, t_i^{start}, t_i^{end}) \mid i = 1,2,\dots,n\}, \quad \text{stored as TextGrid} \quad (7)$$

where the phoneme boundaries Bprovide start and end times for each phoneme p_i, allowing precise extraction of phoneme-specific acoustic features. This structure facilitates detection of pronunciation errors like insertions, deletions, or substitutions and enables personalized feedback for ESL learners.

D. Acoustic Feature Extraction

Once the phoneme alignment is done, the second step is the conversion of each segment of the text grid into a structured representation of the acoustic properties of the speech referred to as the feature extraction stage. This is a very important step because it gives the neural network a chance to get the data to differentiate between the right and the wrong phoneme pronunciations. Each of the time-linked phoneme records a mixture of low- and mid-level characteristics such as log-Mel spectrograms, MFCCs, and prosodic characteristics such as pitch, duration, and energy. Such extracted features are subsequently fed into the hybrid deep learning system, which allows proper phoneme categorization and assists with identifying the pronunciation mistakes, as explained in Eq. (8):

$$F_i = \{LogMel(x_i), MFCC(x_i), Pitch(x_i), Energy(x_i), Duration(x_i)\} \quad (8)$$

where each phoneme segment $x_i$is converted into acoustic features F_i , combining spectral, prosodic, and temporal properties.

1) Log-mel spectrograms: The log-Mel spectrograms are the energy of the speech over time and frequency but using the perceptually-based Mel scale which puts more emphasis on the low frequencies and squeezes the high frequencies. They are produced as the result of short-time Fourier transform, Mel filtering, and log compression and represent formants and articulatory transition. CNN layers are trained on spectral patterns, phonetically similar sounds, to be distinguished to conduct an appropriate analysis of ESL pronunciations. It is calculated in Eq. (9):

$$S_{log_mel}(t,f) = \log\backslash Big(\sum_{k=1}^{K} |X(t,f_k)|^2 \cdot H_{mel}(f_k,f)\backslash Big) \quad (9)$$

where the Log-Mel spectrogram is the distribution of energy of speech on a perceptual Mel scale. It records phoneme changes, formants and articulation patterns. Convolutional layers are able to learn local spectral patterns, which can distinguish between phonetically similar-sounding in pronunciation of ESL learners.

2) Mel-frequency cepstral coefficients: MFCCs isolate the spectral perfection of speech, which is an indication of voice tract resonances. They are frequency content compression methods extracted to cover both the frequency content and are resistant to pitch and volume changes through Fourier transform, Mel filtering and DCT. First and Second derivatives are used to calculate temporal dynamics. The model can be fed with the MFCCs into recurring layers so as to analyze changes in articulation over time. They are particularly successful at identifying the slight phonetic variation and mispronunciation, which would offer a strong framework on which to judge the phoneme production of ESL students and could be used in providing accurate feedback on pronunciation. It is mentioned in Eq. (10):

$$\text{MFCC}_n = \sum_{m=1}^{M} \log\backslash Big(\sum_{k=1}^{K} |X(t,f_k)|^2 H_{mel}(f_k,f_m)\backslash Big)\cos\backslash Big\left[\frac{\pi n}{M}\left(m-\frac{1}{2}\right)\backslash Big\right] \quad (10)$$

where MFCCs code spectral envelope of speech on the basis of Fourier transform, Mel filtering, and DCT. Temporal derivatives are volatile. They give good abilities of detecting the faintest phonetic changes and help in proper performance of pronunciation in ESL students.

3) Temporal and prosodic parameters: Rhythm, stress, and fluency are represented in the form of the temporal and prosodic features. Phonemic length determines the existence of overlong vowels, or absent consonants, the intonation of the intonation and stress placement is detected through pitch, and the force of articulation is detected through energy. Taken together with spectral properties, these features make a detailed picture on a phoneme level. Log-Mel spectrograms, MFCCs and prosodic measures are normalized and presented to the hybrid neural network in form of tensors. It can be used to check pronunciation errors (like mistimed articulation, improper tongue or lips position, and stress misplacement) which are the basis of phoneme classification and individualized feedback. It is equated in Eq. (11):

$$F_{prosody} = \{\text{Duration}(p_i), \text{Pitch}(p_i), \text{Energy}(p_i)\} \quad (11)$$

In the cases of phonemes duration, pitch and energy as prosodic features. Furthermore, these features in conjunction

with spectral features determine rhythm, the level of stress and fluency. They are effective in the detection of mistimed articulation or wrong stress on a word or weak articulation. Finally, the features form the foundation of phoneme classification and offer individual pronunciation feedback to every person.

Algorithm 1 explains the acoustic Feature Extraction provides the steps involved in transforming raw audio signal into feature vectors to analyse speech. The system removes Mel-Frequency Cepstral Coefficients (MFCC) as spectral representation, log-Mel spectrograms as frequency-time patterns and temporal representations, as phoneme duration, pitch, and energy, depending on the type of feature selected. The various types of features can be used in combinations to enhance richer representations such as MFCC with spectrogram, or MFCC, spectrogram, and temporal features. Extracted features are combined into a unified vector that is used as the neural network model input in such tasks as phoneme recognition and pronunciation error detection.

Algorithm 1: Acoustic Feature Extraction

```
Input: Raw audio signal
Output: Feature vector (MFCC / Spectrogram / Temporal / Combination)
  if feature_type == 'MFCC':
        Extract Mel-Frequency Cepstral Coefficients
        features = extract_mfcc(audio)
  if feature_type == 'Spectrogram':
        Extract log-Mel spectrogram
        features = extract_spectrogram(audio)
  if feature_type == 'Temporal':
        Extract temporal features like duration, pitch, energy
       features = extract_temporal_features(audio)
   if feature_type == 'MFCC+Spectrogram':
            Combine MFCC and Spectrogram features
        mfcc = extract_mfcc(audio)
        spec = extract_spectrogram(audio)
        features = concatenate (mfcc, spec)
  if feature_type == 'MFCC+Spec+Temp':
       Combine all features
            mfcc = extract_mfcc(audio)
            spec = extract_spectrogram(audio)
            temp = extract_temporal_features(audio)
            features = concatenate (mfcc, spec, temp)
   else
            Invalid feature type specified.
            features = None
 return features
```

E. Phoneme Recognition on Hybrid Deep Learning Module

The main premise of the suggested structure is a deep learning hybrid architecture that will be used to determine phoneme-level pronunciation patterns of ESL learners. The model takes acoustic features such as log-Mel spectrograms, MFCCs and prosodic parameters as inputs (and predicts labels of phonemes). Its structure consists of four parts, including CNN, Bi-GRU, attention mechanism, and output classification layer, which are used to detect fine-grained errors based on time-aligned segments of phonemes. The results of these are then inputted into the Articulatory Error Mapping Engine, which gives the results in a linguistically interpretable and pedagogically valuable feedback. Fig. 5 demonstrates the Hybrid CNN-BiGRU-Attention model of phonemes recognition.

1) Convolutional Neural Network: CNN layers process the 2D Log-Mel spectrograms and as a result respond to spatially localised events, such as formants, frequency modulations and energy shifts.

2) Bidirectional Gated Recurrent Units: The Bi-GRU layers then extend the CNN layers to capture temporal relationships and discover sequence patterns as they read the input sequences forward and backward and get individually used with ESL learners and offer the opportunity to recognize phonemes in continuous speech.

3) Attention mechanism: Bi-GRU generates an attention layer, which is subsequently implemented to weight the importance of every time or phoneme. It places additional emphasis on mispronounced phonemes when implemented among ESL learners making the model more discriminative and enabling more intelligible and focused detection of pronunciation deviations in a sentence.

4) Output classification layer: This is the last layer and is a fully connected softmax that provides the probability of phonemes (on a time-aligned segment) probability. Segments are given predicted labels due to the most likely occurrence and compared to canonical sets of phonemes. The output also drives the pronunciation feedback module allowing the pronunciation accuracy to be evaluated and a specific corrective feedback to be generated to work with ESL learners.

F. Phoneme-Level Error Processing

The last stage of the system produces learner specific feedback through comparison of predicted phoneme sequences and reference sequences. It finds pronunciation errors on a phoneme level, such as substitutions, omissions and insertions using alignment metadata and classification output. The individual errors associated with each linguistic category (voicing, place of articulation, and so forth) are mapped to a feedback template. The system generates clear textual feedback, which shows the particular phoneme and provides articulatory corrective advice. The errors are then run through the Articulatory Error Mapping Engine that translates them into articulatory categories that can be interpreted to provide learners with accurate, actionable and pedagogically significant feedback. Fig. 3 shows hybrid AI-framework.

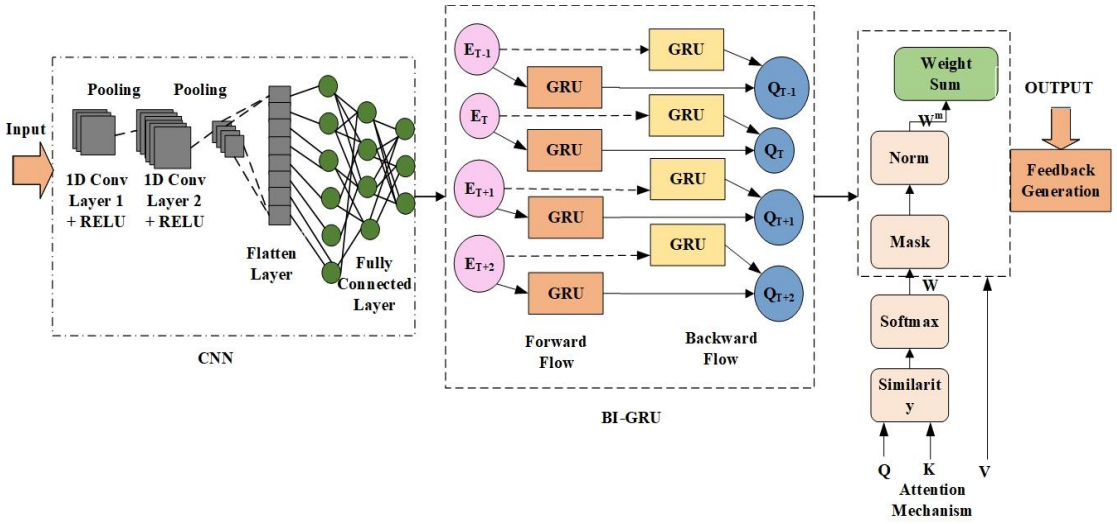


Fig. 3. Hybrid AI framework.

G. Articulatory Error Mapping Engine

The Articulatory Error Mapping Engine is the fundamental novice of the suggested framework. Although the hybrid CNN-BiGRU-Attention model is capable of recognizing phoneme-level substitutions, insertion, and deletion errors, they need to be linguistically interpreted so that they can be pedagogically significant to ESL learners. The phoneme that is mispronounced is mapped to its articulatory characteristics such as place of articulation, manner of articulation, voicing, and vowel quality to a predefined phoneme-to-articulatory dictionary. The errors are identified and registered as errors in articulation (e.g., the alveolar/dental, voiced/voiceless, high/mid vowel, etc.). The engine finds the corrective feedback templates associated with the non-canonical articulatory feature so that actionable advice is provided, e.g. production shifted to alveolar /t/; /th/ needs the tongue between the teeth. This mapping converts raw classification results into linguistically intelligible feedback (with a learner-oriented result), which is the difference between the system and other conventional ASR-based pronunciation evaluation tools. The flow-chart of the phoneme recognition and feedback generation process run by AI is presented in Algorithm 2.

Algorithm 2: AI-Based Phoneme Recognition and Feedback Framework

BEGIN
1. Data Collection
    Load ESL speech samples from Mozilla Common Voice
    Select samples tagged with non-native accents
2. Data Preprocessing
    FOR each audio sample
        Resample audio to 16 kHz
        Apply silence trimming using energy threshold
        Perform spectral subtraction for noise removal
        Normalize transcripts (lowercase, remove punctuation)
3. Phoneme Alignment
    FOR each transcript
        Convert text to phonemes using CMU Pronouncing Dictionary
        Align phonemes using Montreal Forced Aligner (MFA)
        Save phoneme boundaries with timestamps
4. Feature Extraction
    FOR each aligned phoneme segment
        Extract Log-Mel spectrogram
        Extract MFCC + delta + delta-delta
        Extract prosodic features (duration, pitch, energy)
5. Phoneme Classification
    FOR each feature vector
        Apply CNN for spatial feature extraction
        Pass output to Bi-GRU for temporal modeling
        Apply attention to emphasize key phoneme frames
        Predict phoneme label using softmax layer
6. Error Identification
    FOR each predicted phoneme
        IF predicted phoneme == reference phoneme
            Mark phoneme as correctly pronounced
        ELSE
            Classify error as substitution, deletion, or insertion
7. Articulatory Error Mapping Engine
    FOR each mispronounced phoneme
        Retrieve articulatory features of reference phoneme
        Retrieve articulatory features of predicted phoneme
        Identify deviation in:
            - Place of articulation
            - Manner of articulation
            - Voicing
            - Vowel quality
        Select appropriate corrective feedback template
        Generate articulatory explanation (e.g.,
        "Tongue should be between teeth for /θ/, but moved to alveolar /t/.")
8. Output Generation
    Display or export structured feedback combining:
        - Phoneme-level correctness
        - Error type
        - Articulatory deviation
        - Corrective instruction
END

Algorithm 2 describes the AI-Based Phoneme Recognition and Feedback Framework of systematic pronunciation assessment. Preprocessing of non-native speech samples is carried out by resampling, silence cutting, noise cancellation and transcript normalization. MFA is the conversion of words to phonemes and alignment. Each segment is extracted and features (spectral, MFCC and prosodic) are computed (duration, pitch, energy). Errors in pronunciation are detected by the Articulatory Error Mapping Engine, which transforms it into a place, manner, voicing, or vowel-quality error. The system then develops learner-centered, structured feedback on pre-defined articulatory templates to give interpretable and pedagogical useful pronunciation instructions.

IV. Results and Discussion

The suggested framework became very successful in identifying patterns at phoneme level in the speech of ESL speakers and properly identifying some common mistakes in pronunciation, such as substitutions, deletions, or additions. This was able to differentiate closely related phonemes and record articulatory deviations which are normally difficult among non-native speakers. The system empowered articulatory categories by incorporating the Articulatory Error Mapping Engine to give the recognition outputs clear articulation categories, and it provided articulate explanations of errors. This interpretability made the pedagogical value of it more precise and learner-centered. In general, the findings show that the framework provides correct phoneme recognition, high-quality error detection, and linguistically based corrective support thus is an excellent tool to be used in practical ESL pronunciation training.

A. Experimental Setup

The experiment took the Common Voice dataset, which was phonemically synchronized by MFA and CMUdict. processed audio and extracted features, such as MFCCs, spectrograms, and temporal features. The Python-based model was tested in terms of accuracy, precision, recall, and F1-score. The parameters of the simulation are shown in Table II.

TABLE II. Simulation Parameters

Parameter	Value / Description
Sampling Rate	16 kHz
Window Size	25 ms
Window Stride	10 ms
Number of MFCCs	13
Spectrogram Type	Log-Mel Spectrogram
Optimizer	Adam
Learning Rate	0.001
Batch Size	32
Epochs	50
Dropout Rate	0.3 (applied to Bi-GRU and FC layers)
Activation Function	ReLU (CNN), Tanh (Bi-GRU), Softmax (output)
Loss Function	Categorical Cross-Entropy

B. Phoneme Recognition Performance

Phoneme recognition module was introduced on the Common Voice English corpus, a hybrid CNN-BiGRU-Attention architecture was trained on a strictly selected subset of the corpus, which was specifically chosen to reflect the non-native English pronunciation. The phoneme sequences were phoneme sequences that were synchronized through forced alignment to enable accurate timing matching of predicted and reference phonemes. Table II illustrates the performance indicators on phoneme recognition.

TABLE III. Phoneme Recognition Performance Metrics

Metric	Value (%)
Accuracy	91.4
Precision	89.7
Recall	90.1
F1-Score	89.9

The final model that combines AI and a hybrid phoneme recognition model performed well enough over all of the evaluation measures, giving a balanced performance on all the metrics. The model was highly accurate in terms of identifying phonemes uttered by ESL learners, given that the overall accuracy was 91.4%. These findings point to the fact that the system has a high level of reliability and can be used in ESL to test the articulation of phonemes of pronunciation. Fig. 4 shows phoneme-level accuracy plot.

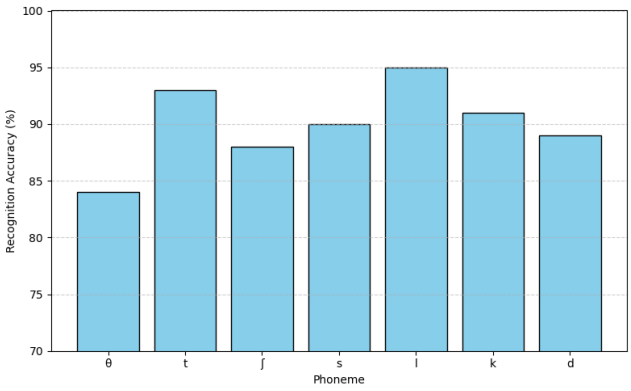


Fig. 4. Phoneme-level accuracy plot.

The proposed hybrid AI model recognizes individual phonemes by 56.4 % accuracy as shown by Fig. 6. The highest accuracy score was achieved by /l/ phoneme at 95 %, then there was a close-up of /t/ and /k/ which means that the consonantal sounds that are dominant were well performed. More difficult phonemes, which ESL learners are usually unable to pronounce well, still achieved 83.95 and 88.33 score. In general, the figure brings to the fore the fact that the model in question demonstrates a consistent phoneme-level classification of various ESL speech. The predictions are entered into the Articulatory Error Mapping Engine, where the wrong pronunciation can be translated in terms of its particular articulatory error.

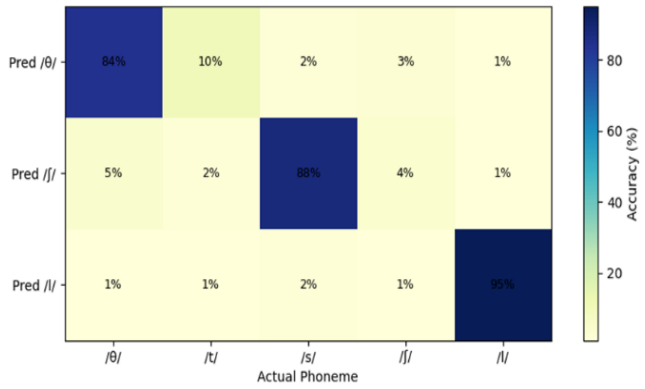


Fig. 5. Phoneme confusion matrix.

Fig. 7 explains the accuracy of the model to sort out the similar phonemes in ESL speech. The rates of high accuracy are noted by /theta/, /sh/, and /l/, which show models' recognition at the scale of 84%, 88%, and 95% respectively, suggesting an excellent precision level of models. Another thing that is noticed in the matrix is that /theta/ is sometimes wrongly categorized as /t/ (10%) and /sh/ as /s/ (4%), something which is in line with common ways in which ESL students are mispronouncing. In predicting /l/, minimal confusions are evident, demonstrating the effectiveness of the system on considerations that are late.

C. Error Detection and Categorization

Deletion errors are errors that are committed when a phoneme is not pronounced. Through the comparison of the aligned predicted phoneme sequence with their corresponding reference phoneme sequence, the system is capable of detecting such errors. The accuracy of detection of each type of error was calculated individually in order to determine sensitivity to the various articulation patterns. Table III shows a summary of the detection accuracy of all error categories.

TABLE IV. DETECTION ACCURACY

Error Type	Detection Accuracy (%)
Substitution	84
Deletion	82
Insertion	79

The system proved to be quite successful in identifying all three large types of pronunciation errors among the ESL learners. The substitution errors were found to be detected with the best accuracy rate of 84%, which is a sign of the sensitivity of the model towards the frequent mispronunciation of phonemes. The deletion errors were also well detected with an 82 percent detection, so it could be said that error deletion could be reliably detected. Insertion errors were also found in slightly lesser precision of 79% however, they were still spotted quite accurately.

Fig. 8 shows the number of errors successfully detected by three significant error types, namely substitution, deletion, and insertion. The highest accuracy rate of 84% was achieved in identifying errors involving replacements of phonemes which are characteristic of ESL speech, revealing the strength of the model. In the next position, deletion errors existed at 82%, which indicates consistent results in sound omission detection. Insertion error, which was a little bit lower at 79%, was adequately identified, thus showing overall model competence. The high credibility of the phoneme-level error analysis by the system is highlighted by similar error detection rates in all the error types.

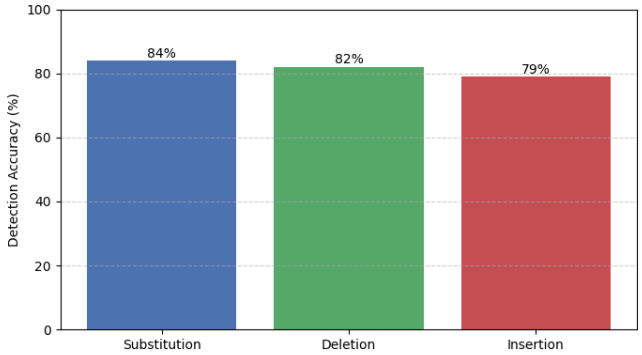


Fig. 6. Detection rate by pronunciation error.

D. Effectiveness of the Articulatory Error Mapping Engine

To determine the pedagogical significance of the proposed Articulatory Error Mapping Engine, the phoneme-level errors identified were further divided into articulatory types including the place of articulation, manner of articulation, voicing, and vowel quality. Mapping accuracy was calculated by comparing the categorization of the system with that of the experts-linguists in a sample of 300 mispronounced tokens.

TABLE V. PERFORMANCE OF THE ARTICULATORY MAPPING

Articulatory Category	Mapping Accuracy (%)
Place of Articulation	90.8
Manner of Articulation	88.5
Voicing Distinction	92.3
Vowel Quality	87.1

Table V shows the performance of the articulatory mapping module. The engine showed remarkably high mapping accuracy, and most phoneme deviations were accurately mapped into the articulatory features they were supposed to be. Such degree of readability is absent in the traditional ASR-based pronunciation devices, which can only give out scores or error indicators without attributing articulatory patterns to the errors.

E. Feature Set Contribution to Accuracy

The phoneme-level pronunciation errors in ESL students are of primary importance, to learn about the acoustic and temporal features that have the most significant influence on a successful classification. The analysis concerned trying different combinations of MFCC, spectrogram, and temporal features and how they made the difference individually and collectively. The outcomes can be used as evidence to support the effectiveness of the use of the model in providing quality, accurate, and reliable pronunciation feedback.

Fig. 9 shows how the various feature sets affect the accuracy of phoneme recognition. Accurate fused feature set was at

88.6% with slight reduction to 88.5% when using only MFCC and spectrogram, and lower with individual features: MFCC: 82.3%, spectrogram: 80.7% and temporal features: 69.4%. This shows that to a great extent, the factor that improves phoneme recognition is the combination of acoustic and time capabilities, which further increases the quality and accuracy of the feedback provided by the Articulatory Error Mapping Engine.

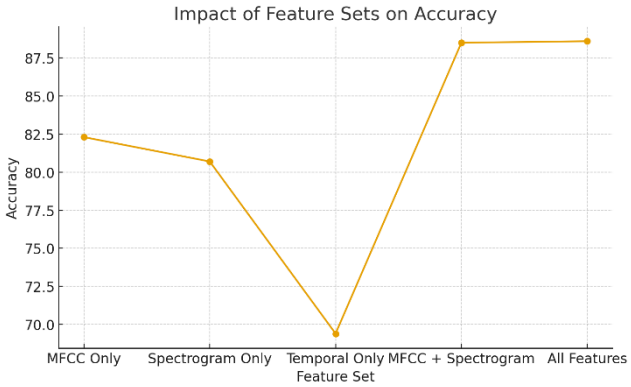


Fig. 7. Impact of feature sets on accuracy.

F. Frequency of ESL Pronunciation Errors by Type

When it comes to ESL pronunciation assessment, it is important to understand the kind of errors learners make so that it provides pointed feedback. The developed AI-based analyzer not only finds the phoneme-level inaccuracy, but also divides it into known linguistic features based on place, manner, voicing, and vowel quality.

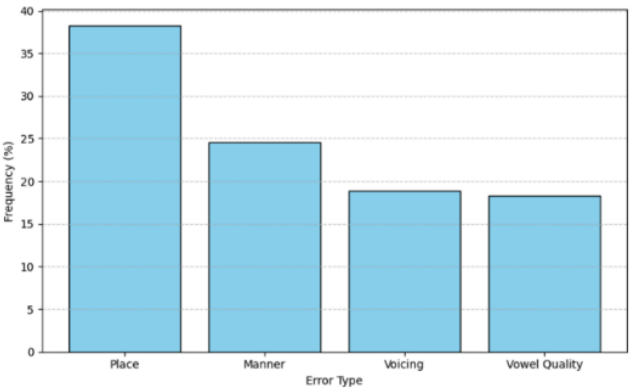


Fig. 8. Error over frequency distribution.

Fig. 10 discusses the frequency of pronunciation error types, which were detected in ESL learner speech. The most frequent mistake occurred in place of articulation (38 percent of the entire cases), which means difficulties of learners in accurate positioning of the tongue (e.g., / 0/ replaced by /t/). The substitution of affricates with stops in manner of articulation errors was followed with a rate of 24.6%. Less common yet also large were voicing and vowel quality errors, and they were about 1819%.

G. Loss vs. Epochs

Loss vs. Epoch graph indicates the decrease in error of the model during training cycles. The points will be the loss at each epoch, and this can be regarded as a measure of the efficiency of the model using the data. A constant downward trend shows the successful convergence and better the overall performance of the model.

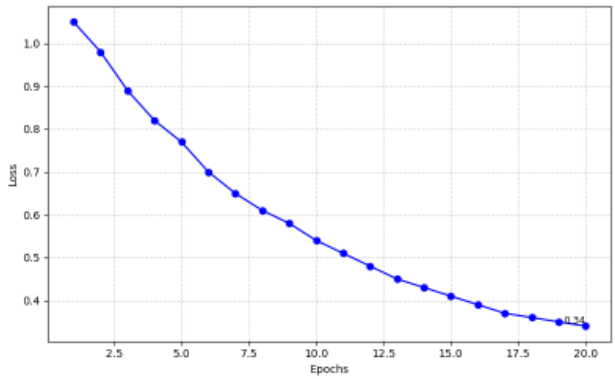


Fig. 9. Loss vs. epochs.

Fig. 11 demonstrates the value of losses at different training epochs and the tendency is to decrease. The last loss decline of 0.34 means high convergence and stable learning, that is, the CNN-BiGRU-Attention model was effective to reduce the prediction errors. This consistent trend is a confirmation that the model will provide reliable response on the training data without overfitting and that the phoneme predictions will be accurate to the downstream Articulatory Error Mapping Engine.

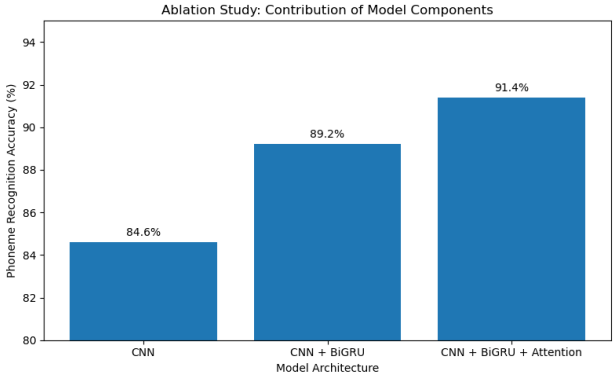


Fig. 10. Ablation study: Contribution of model components.

Fig. 12 graph shows that adding BiGRU improves temporal phoneme modeling, while the attention mechanism further increases accuracy by focusing on mispronounced phoneme segments in ESL speech.

H. Performance Comparison

A comparative analysis was carried out to determine the performance of the suggested hybrid CNN-BiGRU-Attention model in comparison with the conventional HMM-GMM models and commercial ASR-based pronunciation system. The proposed system recognized phonemes with an accuracy of 91.4 which was far much better than the ASR-based system (78.3) and the HMM-GMM base (70.5). In contrast to traditional ASR systems which focus more on the word-level decoding performance, the presented framework functions directly at the phoneme level and introduces the structured articulatory interpretation. This enhancement is not just in the classification accuracy though but also to the error granularity and

interpretability. Moreover, where large speech models based on transformers focus on large-scale representation learning, the given architecture shows that specific phoneme-level modeling with articulatory mapping can be created to attain competitive accuracy at reduced computational costs and with increased pedagogic explainability. This proves the appropriateness of the framework in assessment activities based on ESL-based pronunciation.

TABLE VI. PERFORMANCE COMPARISON

System	Phoneme Accuracy (%)
Proposed AI Hybrid System	91.4
ASR-Based Tool [16]	78.3
HMM-GMM Baseline [21]	70.5

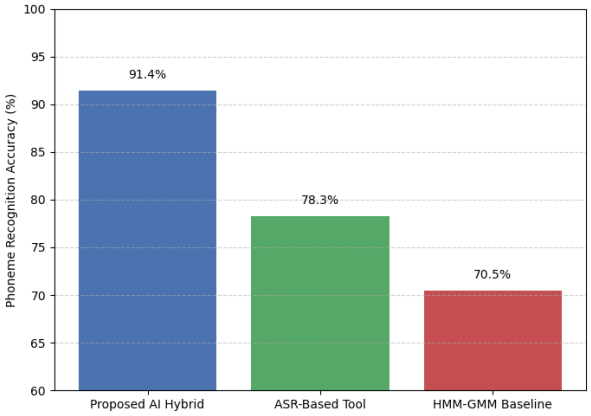


Fig. 11. Model performance comparison.

Fig.12 demonstrates the high level of the suggested AI-based hybrid system in comparison to the known pronunciation assessment tools. The proposed model has a phoneme recognition rate of 91.4 as compared to the 78.3 recorded by the ASR-based tool and the 70.5 by the HMM-GMM baseline. This demonstrates the effectiveness of that combination of CNN, Bi-GRU, and attention mechanism and a phoneme-level strategy. In every experiment, the Articulatory Error Mapping Engine leads to the difference between the proposed framework and other standard recognition systems, whereby raw recognition outputs are converted into actionable articulatory-level feedback, which has a clear pedagogical value that is absent in other standard recognition systems (ASR or HMM-based).

I. Robustness Analysis

The evaluation of robustness was conducted to evaluate the performance in the presence of realistic acoustic and speaker variations. At medium noise levels, phoneme accuracy remained relatively unchanged with the addition of background noise at 20 dB, 10 dB and 5 dB SNR levels, suggesting that spectral-temporal modeling is stable. Higher degradation was observed at very small SNR, but still the performance was higher than the traditional baselines. There was also diversity in terms of accents, gender and age. Findings showed similar phoneme recognition and articulatory classification in profiles of speakers. The comparison of pairs of phonemes that are most often confused and voiced-voiceless pairs proved that the attention layer improved discrimination. The ability of the framework to generalize was also enabled further by speaker-independent cross-validation.

J. Discussion

The experimental findings indicate that the developed hybrid CNN-BiGRU-Attention architecture is effective in both spectral and temporal features of the ESL speech, which makes it possible to recognize the phoneme level accurately. BiGRU layers make the modeling of coarticulation and timing variation greater, whereas the attention mechanism makes the distinction of mispronounced phonemes more intense. Moreover, the Articulatory Error Mapping Engine is able to map the identified phoneme errors into articulatory deviations that are interpretable to provide pedagogically relevant feedback. All in all, the framework performs well on the substitution, deletion and insertion error, much better than the traditional ASR-based and HMM-GMM systems in performance and interpretation.

V. CONCLUSION AND FUTURE WORKS

The paper introduced a hybrid AI-based phoneme recognition model to deliver valid and interpretable pronunciation evaluation of ESL students. The proposed system involved the fusion of CNN, bi-GRU, and attention with phoneme alignment and spectral-prosodic feature fusion to reach an accuracy of phoneme recognition of 91.4, compared to commercial ASR-based systems (78.3) and traditional HMM-GMM systems (70.5). The framework was quite useful in identifying errors in pronunciation with detection rates of 84, 82 and 79 percent in terms of substitutions, deletions and insertions. The suggested solution promotes intelligent training of ESL pronunciation, self-directed learning, and instructor-guided assessment; it provides a scalable, interpretable, and learner-centered method of teaching pronunciation at the phoneme level.

Further contributions to work will involve extension of the proposed framework to multilingual and tonal languages with more advanced models of prosody, stress and intonation. By incorporating transformer-based speech models with real-time mobile/web deployment, this addition will help provide extra strength, scalability and personalized pronunciation feedback to various ESL learners.

REFERENCES

[1] S. Alam, “The Conceptual Relevance of English as Lingua Franca in Non-English Speaking Countries: Revisiting History, Policies and Praxis.,” Theory & Practice in Language Studies (TPLS), vol. 13, no. 9, 2023.

[2] Z. Kupaysinova, “ENHANCING STUDENTS’LANGUAGE PROFICIENCY THROUGH INNOVATIVE INTEGRATED METHODS IN CONTEMPORARY LANGUAGE EDUCATION,” Mental Enlightenment Scientific-Methodological Journal, vol. 6, no. 03, pp. 205–217, 2025.

[3] N. Khan, A. Khan, A. Shahzadi, and others, “Students’ Pronunciation and other Languages: The Impact of L2 Interference on the Pronunciation of ESL Students,” Pakistan Journal of Society, Education and Language (PJSEL), vol. 10, no. 1, pp. 163–175, 2023.

[4] W. S. ANISSA, “The Influence of ELSA Speak Application Towards Studentsenglish Pronunciation Mastery,” Phd thesis, UIN Raden Intan Lampung, 2025.

[5] L. Yang, “Student engagement with teacher feedback in pronunciation training supported by a mobile multimedia application,” SAGE Open, vol. 12, no. 2, p. 21582440221094604, 2022.

[6] W. Sun, "The impact of automatic speech recognition technology on second language pronunciation and speaking skills of EFL learners: a mixed methods investigation," Frontiers in Psychology, vol. 14, p. 1210187, 2023.

[7] T. P. Zanto et al., "Digital rhythm training improves reading fluency in children," Developmental Science, vol. 27, no. 3, p. e13473, 2024.

[8] R. El-Bialy et al., "Developing phoneme-based lip-reading sentences system for silent speech recognition," CAAI Transactions on Intelligence Technology, vol. 8, no. 1, pp. 129–138, 2023.

[9] P. Sullivan, T. Shibano, and M. Abdul-Mageed, "Improving automatic speech recognition for non-native english with transfer learning and language model decoding," in Analysis and application of natural language and speech processing, Springer, 2022, pp. 21–44.

[10] B. Maji and M. Swain, "Advanced fusion-based speech emotion recognition system using a dual-attention mechanism with conv-caps and bi-gru features," Electronics, vol. 11, no. 9, p. 1328, 2022.

[11] T. S. Nguyen, T. D. T. Nguyen, N. Q. N. Hoang, and T. K. H. Do, "How AI-Powered Voice Recognition Has Supported Pronunciation Competence among EFL University Learners," Computer-Assisted Language Learning Electronic Journal, vol. 26, no. 3, pp. 64–83, 2025.

[12] S. Albaladejo Albaladejo, "Improving the English pronunciation proficiency of university students through the use of pronunciation learning strategies," Proyecto de investigación, 2022.

[13] H. P. Tiwari, "Challenges in teaching pronunciation: Secondary level English teachers' perspectives," Journal of Linguistics and Language in Education, vol. 17, no. 2, pp. 1–29, 2023.

[14] G. Sanders and A. de Bruin, "Examining the difference in error detection when listening to native and non-native speakers," Quarterly Journal of Experimental Psychology, vol. 76, no. 7, pp. 1547–1560, 2023.

[15] D. Abimanto and W. Sumarsono, "Improving English Pronunciation with AI Speech-Recognition Technology," Acitya: Journal of Teaching and Education, vol. 6, no. 1, pp. 146–156, 2024.

[16] Y. Xu, "English speech recognition and evaluation of pronunciation quality using deep learning," Mobile Information Systems, vol. 2022, no. 1, p. 7186375, 2022.

[17] N. T. Hoang, D. N. Han, and D. H. Le, "Exploring Chatbot AI in improving vocational students' English pronunciation," AsiaCALL Online Journal, vol. 14, no. 2, pp. 140–155, 2023.

[18] K. S. Raja and D. D. Sanghani, "Speech Emotion Recognition Using Machine. Learning," 2024.

[19] S. Permatasari, "Enhancing pronunciation skills in EFL students through the ELSA Speak application," Indonesian EFL Journal (IEFLJ, 2024.

[20] A. Babaeian, "Pronunciation Assessment: Traditional vs Modern Modes," Journal of Education For Sustainable Innovation, vol. 1, no. 1, pp. 61–68, 2023.

[21] A. A. Zaveri, "Hangman–A Voice-Over Game," Journal of Computing Technologies and Creative Content (JTec), vol. 8, no. 1, pp. 27–35, 2023.

[22] B. Zou, Y. Du, Z. Wang, J. Chen, and W. Zhang, "An investigation into artificial intelligence speech evaluation programs with automatic feedback for developing EFL learners' speaking skills," Sage Open, vol. 13, no. 3, p. 21582440231193818, 2023.

[23] Y. E. Kheir, A. Ali, and S. A. Chowdhury, "Automatic Pronunciation Assessment–A Review," arXiv preprint arXiv:2310.13974, 2023.

[24] S. Yoo and H. Ahn, "The Effects of Prosody Training with AI Chatbot on the English Pronunciation Improvement of Korean EFL Learners," 영어학, vol. 24, pp. 1300–1317, 2024.

[25] Mozilla, "Common Voice." Accessed: Jul. 09, 2025. [Online]. Available: https://www.kaggle.com/datasets/mozillaorg/common-voice

Cloud-Continuum-Based Deep Learning Optimization Framework for Next-Generation Healthcare Data Performance on IoT Platform

G. Aravindh[1], K.P. Sridhar[2]
Research Scholar, Department of ECE, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[1]
Professor, Department of ECE, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[2]

Abstract*—The development of healthcare data performance analysis is becoming more driven by the incorporation of intelligent computing paradigms that guarantee real-time, scalable, and personalized feedback for coaches and athletes. However, existing healthcare data analytics systems are challenged with severe issues such as decision-making latency, processing capacity limitations at the edge, data fragmentation, and the inability to integrate across heterogeneous computing environments seamlessly. Athletic data in this scenario refers to a combination of biomechanical factors (motion capture, joint angles, gait patterns), biometric signals (heart rate, oxygen saturation, muscle activity), and sport-specific performance indicators (workload, speed, and acceleration). This paper introduces the Cloud-Continuum-based Deep Learning Optimization Framework (CC-DLOF). This novel architecture leverages the synergistic potential of edge, fog, and cloud computing to provide a dynamic and smart healthcare data performance on an IoT platform. CC-DLOF is a hierarchical continuum architecture, with real-time data gathering and lightweight analytics performed in the edge layer, contextual processing and federated learning in the fog layer, and global intelligence, deep model training, and long-term data storage in the cloud layer. A new Cloud-Fog-Edge Orchestration Device (CFEOD) provides dynamic allocation of computational tasks in terms of latency sensitivity and device capability. At the same time, a blockchain-supported access control is used to maintain data security and privacy. Simulation analysis, done in a simulated training environment that combines with real-world data sets, illustrates the performance of the framework in mitigating latency by 35%, increasing model accuracy by 22%, and boosting system scalability and reliability. CC-DLOF is a revolutionary way in healthcare data technology, leading to smart, responsive, and safe next-generation healthcare data performance on an IoT platform.

Keywords*—Cloud; continuum; next-generation; healthcare data; performance; platforms; fog; edge; orchestration; device

I. Introduction

Intelligent, real-time analytics are now more critical than ever in the universe of modern athletic performance [1]. Yet, most existing healthcare data performance platforms employ centralized cloud-computing models, which possess a number of disadvantages, such as delayed processing of distributed data, high latency, and limited scalability in real-time environments [2]. The uninterrupted and continuously changing information generated by biomechanical monitoring systems, video analysis, and wearable sensors overwhelms such systems [3]. Moreover, training and competition, critical time-dependent decision-making is hindered by delayed feedback that results from data flow blockages induced by a lack of seamless connectivity throughout cloud, fog, and edge infrastructures [4]. In addition, data security and privacy are issues of growing concern, especially in the context of processing sensitive biometric information [5]. The ethical and compliance uses of athlete data have been interrogated because existing solutions lack robust means of secure data exchange and model training across distributed sources [6]. Besides, coaching interventions have limited effectiveness due to the incapacity to customize analytics models to individual athletes on a real-time basis, which generates generic feedback [7]. The desperate need is an increasingly flexible, high-speed, and intelligent computational framework that will handle these issues and work best at a cloud continuum [8]. Therefore, the current challenge is to develop a singular IoT platform that can minimize latency, raise computational efficiency, protect data, and provide context-aware deep learning attributes [9]. This will lead to a better performance forecast, lower risk of hurt and a long-term better development of the athletes [10].

The current athletic performance analysis systems are mainly based on cloud computing, wearable devices, and analysis software that is independent of artificial intelligence [11]. To judge performance in athletics, the older systems have been storing on cloud structures to then sieve through masses of data collected using wearable sensors, GPS devices, and videos [12]. The most common uses of AI algorithms are fatigue estimation, pattern recognition, and post-session feedback [13]. The real-time responsiveness of such cloud-based systems has significant limitations because of the bandwidth and network delay, but when the analysis is long-term, it is all right. Other systems use edge computing to develop low-latency data collection and basic processing; these systems typically do not have the processing capability or similar learning capacity to execute advanced analytics [14]. Although hybrid solutions involving cloud-edge frameworks have come into existence, they are not well-integrated and, therefore, allocate workloads in an inefficient fashion [15]. Additionally, most systems today are not leveraging fog computing, which is an important distributor for processing that considers context. In multi-user or multi-sport environments, where data volumes must be managed, and real-time synchronization becomes more complex, scalability becomes an even greater concern. There is a chance of sensitive athlete information being processed and

shared between platforms because privacy-preserving approaches, like federated learning, are not well-developed. Performance data security is often overlooked and becomes vulnerable to violations. There is no full, secure, low-latency, and scalable platform for bringing AI and IoT into healthcare data; however, some advancements have been made in this regard. Considering these limitations, it is evident that a cloud continuum-based approach, like CC-DLOF, needs to address the issues resulting from the existing approaches.

A. Problem Statement

Because of latency, low scalability, and split processing, cloud-based systems used primarily by existing healthcare data performance on IoT platforms are not appropriate for real-time, data-intensive applications. Although fog integration is in its early stages, edge computing solutions have limited analytics. Reduced personalization, poor data privacy protection, and poor workload distribution are persistent issues. Hence, the healthcare data performance analysis of the future must have an intelligent, responsive architecture that seamlessly integrates cloud, fog, and edge technologies.

B. Motivation

The growing deployment of biomechanical systems, video analysis, and body sensors in sport requires a responsive yet intelligent computational infrastructure. It is imperative for coaches and players to gain improved efficiency of training as well as the diminution of risks related to injuries using real-time customized information. From secure management of data to adaptive learning, along with latency-minimizing feedback, standard systems prove incapable. A chance to build high-performance on IoT platforms with context-sensitive processing and extensible learning has emerged with cloud continuum computing that integrates cloud, fog, and edge computing. Coaches and athletes in high-demand sporting environments can take advantage of meaningful, real-time information offered by an optimized platform that enables persistent learning, intelligent orchestration, and privacy maintenance.

As a growing number of decisions about training, injury prevention, and rehabilitation are based on sports and health data, it is necessary to have computer systems that can handle large amounts of varied data in real time while keeping people's privacy safe. Due to latency, limited scalability, and lack of customisation, current solutions fail to satisfy these needs. Athletes and coaches risk using outdated, generic, or incomplete information without a next-generation, cloud-fog-edge integrated framework. This can make it more difficult to improve performance and reduce the chance of injuries that could have been avoided. This study fills that important need by providing a secure, flexible, and performance-driven architecture that is specifically designed for analysing sports healthcare data.

C. Contribution

In this paper, the author presents a new design of the next generation healthcare data performance platforms known as the CC-DLOF. The architecture will be a combination of edge, fog, and cloud layers, access control system, which is facilitated by blockchain, and a dynamic Cloud-Fog-Edge Orchestration Engine, which will be part of the proposed architecture and aims at distributing workloads in dynamically adaptable fashion, and secures privacy of the users. CC-DLOF also enables real-time and personalized analytics with the federated and deep learning model spread across the continuum. Simulation research shows that this method is more economical in terms of resources, can suffer less latency and is more accurate than more traditional methods. The new model is the most suitable technology that has been developed to optimize sporting performance by means of the innovative cloud continuum technology.

The research concept of healthcare data performance is the capability of a computational system to process, analyze, and communicate healthcare-related data in a way that is highly accurate, with a low latency, and uses resources optimally. It encompasses a number of key performance indicators that comprise speed of decision-making, predictive model precision, ability to scale to a wide range of IoT sensors, and robustness of data privacy and security controls. This is also more than just a performance that can offer real-time, individualized insights that can be used to prevent injuries, optimize training, and monitor recovery with respect to athletics and sports healthcare.

This section covers the structure of the research paper and entails the following: Section II of this paper is about The Cloud Continuum-Based Architecture of Next-Generation Healthcare Data Performance Platforms. Section II of this dissertation explores the subject of CC-DLOF in detail. Section III gives a thorough examination, a comparison with previous approaches and a segregation of the ramifications. Section IV is a comprehensive discussion of the results. Finally, Section V concludes the paper.

II. Literature Survey

The convergence of edge and cloud resources via the Cloud Continuum is now essential to provide intelligent, real-time services in areas such as industrial IoT, smart cities, and structural health monitoring, owing to the rapid evolution of next-generation computing paradigms.

Smart, autonomous management of cloud and edge resources by AI-based optimization is enabled by the ENACT framework presented by Nizamis, A. et al [16]. Throughout the cognitive computing spectrum, it enhances execution time, resource usage, and power efficiency by enabling dynamic scalability, portability, and flexibility for hyper-distributed applications. With regard to orchestration in the Cloud-to-Things continuum, Ullah et al [17]. introduce a comprehensive taxonomy and conceptual framework (CT&CF). To aid future developers in designing effective, scalable deployment and runtime management systems for distant, heterogeneous IoT and cloud resources, it surveys existing work and elucidates significant orchestration requirements and challenges.

Using sensing and software technology to integrate in the precise interpretation of data, Gigli, L. et al [18]. created the MAC4PRO architecture as an interoperable, ready-for-smart-infrastructure sensor-to-cloud Structural Health Monitoring platform. It is validated on industrial scenarios, provides high-accuracy detection of structural changes, and minimizes data transfer from the edge to the cloud by over 90%. A scalable IoT-to-cloud continuum is enabled by the Helix Multi-layered IoT platform (HM-L-IoTP), proposed by Cabrini, F. H. et al [19]. It

deploys federated brokers across both the edge and cloud layers. Its suitability for industrial use in smart cities and sophisticated IoT environments is underpinned by its low-latency, device-to-device communication and global interoperability, which have been proven in real-world deployments.

The Kumar, J. et al [20]. The paper explores 6G networks as a natural evolution of 5G, highlighting their role in enabling intelligent IoT and the Edge-Cloud continuum. It outlines how 6G will transform connectivity, offer ultra-low latency, and equip next-generation IoT systems with enhanced capabilities. It proposes applications that employ deep learning and enhanced security. For enhanced resource management within 6G Cloud Continuum scenarios, Valero, J. M. J et al [21]. propose the Level of Trust Assessment Function (LoTAF) and the Trust Level Agreement (TLA). Trust-based, safe decision-making becomes a reality owing to LoTAF's domain-agnostic trust assessment, which enhances stakeholder collaboration in dynamically changing, multi-provider networks and ensures reliable end-to-end links.

TABLE I. SUMMARIZATION OF EXISTING METHODS

Author Name	Advantages	Interference (Challenges Addressed / Implications)
Nizamis, A. et al.	ENACT enables AI-based autonomous management of cloud and edge; improves execution time, resource use, and energy efficiency; supports hyper-distributed apps.	Tackles dynamic scalability, portability, and flexible deployment of AI-enabled systems across the cloud-edge spectrum.
Ullah et al.	Introduced CT&CF taxonomy; offers a comprehensive framework for orchestration in the Cloud-to-Things continuum.	Addresses complexity in managing distant, heterogeneous IoT and cloud deployments; aids design of scalable runtime systems.
Gigli, L. et al.	MAC4PRO architecture provides accurate structural change detection, reduces edge-to-cloud data transfer by over 90%; ready for smart infrastructure deployment.	Solves data overload by optimizing transfer and processing at the edge while enabling precise SHM in industrial settings.
Cabrini, F. H. et al.	Helix Multi-layered IoT platform ensures low-latency, device-to-device communication; supports global scalability; federated brokers enable IoT-to-cloud continuum.	Overcomes latency and scalability issues; proves real-world viability in smart city and IoT environments.
Kumar, J. et al.	Discusses 6G's role in intelligent IoT; supports deep learning and secure Edge-Cloud continuum applications with ultra-low latency and high-speed connectivity.	Focuses on evolution beyond 5G; deals with the transition to 6G for supporting future smart, intelligent systems.
Valero, J. M. J. et al.	Proposed LoTAF and TLA for trust-aware, E2E resource management; enhances collaboration and decision-making using trust as a service intent.	Addresses uncertainty in multi-provider environments; overcomes trust issues in dynamically changing 6G infrastructure.

The key for future-generation healthcare data performance on IoT platforms and larger hyper-distributed systems is the CC-DLOF, which combines dynamic AI-driven orchestration and trust assessment, provides superior flexibility, low latency, and increased deployment smarts. It is preferable among these alternatives due to its strength.

III. PROPOSED METHOD

Integrating edge, fog, and cloud computing to overcome latency, data fragmentation, and limited processing in conventional healthcare data analytics, this paper presents the CC-DLOF, enabling real-time, scalable, and intelligent performance evaluation.

The Cloud-Fog-Edge Orchestration Device (CFEOD) in the CC-DLOF architecture is an intelligent, latency-sensitive control engine that dynamically distributes computational tasks between edge, fog, and cloud layers based on multi-dimensional profiling of workload properties and resource conditions. It constantly checks the parameters of tasks, such as the latency sensitivity, the computational intensity, the level of data privacy, and the contextual volatility. It compares them with real-time node information, including CPU availability, bandwidth, energy status, and trust score. CFEOD utilizes a hybrid optimization approach based on rule-based filtering and adaptive utility scoring to reduce the end-to-end delay and maximize the accuracy, energy efficiency, scalability, and security. It further synchronizes federated learning and implements an access control policy based on blockchain, which guarantees privacy-saving distributed intelligence and continuum-wide performance optimization.

A. Hierarchical Intelligence Across the Cloud Continuum

The new Cloud-Continuum architecture integrates cloud, fog, and edge layers designed to facilitate deep model training, federated learning, and real-time data collecting. This hierarchical design allows for context-aware processing and scalable analytics in distributed computing systems.

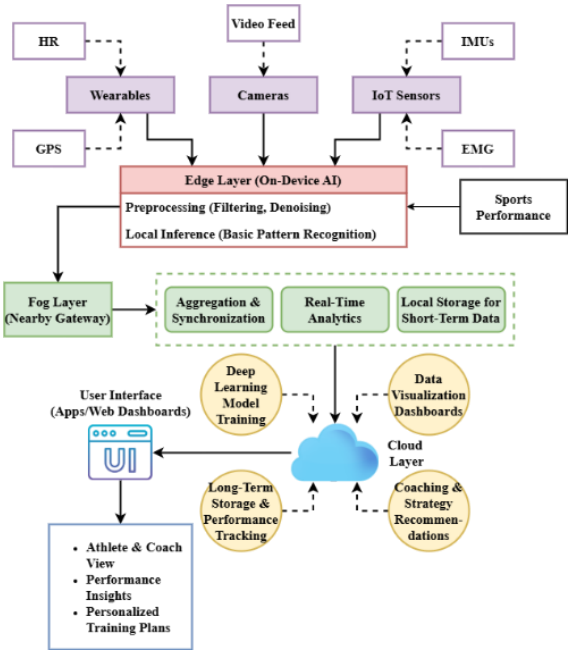


Fig. 1. Cloud continuum-based healthcare data performance framework.

Fig. 1 shows that a multi-layered architecture using cloud, fog, and edge computing makes intelligent, real-time analysis of

healthcare data performance feasible. Wearables, cameras, and IoT sensors complete the numerous sources of data collected at the top. Early data gathering and light processing are handled on-device artificial intelligence supports the edge layer. There is a fog layer that synchronization, local storage, and real-time analytics utilize as a local gateway. Deeper insights come from long-term storage, deep learning model training, and strategic recommendation systems used at the cloud layer. While dynamic orchestration allocates work depending on latency and device capacity, blockchain-based access control guarantees data flow. The approach finishes in clearly visible dashboards and coaching tools with useful results. Low latency, great scalability, and strong customization guaranteed by this tiered continuum help to change data-driven choices made in high-performance healthcare data contexts.

$$(x_u - lx_{yy})(x) = \left(\frac{1}{2}x^2\right)_u + \left(-lx_y x\right)_y + lx_y^2 * (v_1 - v_2) \quad (1)$$

Modeling temporal and spatial dimensions, Eq. (1) aligns.$(x_u - lx_{yy})(x)$and$\left(\frac{1}{2}x^2\right)_u$ reflect interaction forces and contextual fluctuation$lx_y^2 * (v_1 - v_2)$s assessed in the fog layer, the term $\left(-lx_y x\right)_y$ represents local kinetic energy distinctions. This Equation drives CC-DLOF's intelligent transformation of real-time biomechanical data.

$$\left(\frac{1}{2}x^2\right)_u = ey - lx_y x|_{y=0}^{y=m} l\int_0^m x_y^2\ ey + \int x^2\ ey \quad (2)$$

Eq. (2) expresses $ey - lx_y x|_{y=0}^{y=m} l$, reflecting the temporal development of kinetic performance measures at the edge, while accounting$\left(\frac{1}{2}x^2\right)$ for boundary conversations, localized variability ($\int_0^m x_y^2\ ey + \int x^2\ ey$) processed in the fog, and global patterns handled in the cloud. This Equation models knowledge and adaptive performance feedback.

$$\frac{e}{eu}\int_0^m \frac{1}{2}[x(y,u)]^2\ ey = -l\int_0^m [x_y(y,u)]^2\ ey + x \quad (3)$$

With $\frac{e}{eu}$ Eq. (3) models the energy stored or used in edge-level calculations$\int_0^m ey + x$ and represents energy buildup$\int_0^m \frac{1}{2}[x(y,u)]^2\ ey$ and dissipation along $l\int_0^m [x_y(y,u)]^2\ ey$the CC-DLOF continuum. This concept emphasizes that dynamically CC-DLOF controls resource utilization and integrity across dispersed layers for the best healthcare data analytics.

Looking toward the performance of athletes affected by the synergy between Cloud-Continuum computing, federated learning, and virtual immersive environments, Fig. 2 displays an evolved system for healthcare data analytics. Ensuring data protection at many virtual and physical locations is made possible with blockchain-supported access control. Performance information may be experienced in immersive environments via the use of Head-Mounted Displays (HMD) and Computer Automatic Virtual Environments (CAVE) technologies. These technologies simulate real-world training and performance circumstances. Federated learning allows group intelligence to be utilized to facilitate distributed model training on edge devices without sacrificing data privacy. The method via automated insights and live decision-making removes such conventional limits as observation bias, simplicity, and reaction time. Easy-to-use interfaces that provide objectivity, accuracy, and real-time application let players and trainers get outputs. Therefore, allowing them to develop tailored plans for improving performance during training and competition using Cloud-Continuum, healthcare data analysts may scalably handle and analyze vast amounts of data.

$$\int_0^m [v_1(y,u) = v_2(y,u)]^2 ey \int_0^m [\partial_1(y) - \partial_2(y)]^2 ey \quad (4)$$

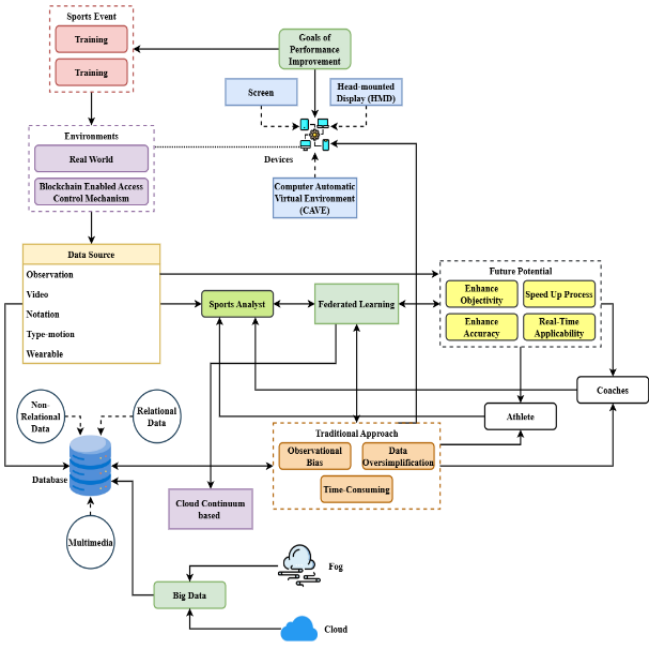


Fig. 2. Intelligent integration of virtual environments and federated learning in healthcare data analytics.

Eq. (4) shows $\int_0^m [\partial_1(y) - \partial_2(y)]^2 ey$ While $[v_1(y,u) = v_2(y,u)]$captures alignment federated outputs, evaluates$v_1(y,u)$ model gradient variation. This Equation guarantees synchronization and consistency across distributed intelligence layers, model accuracy, and individualized feedback.

$$\min_{0\geq y\geq m} |v_1(y,u) = v_2(y,u)| \min_{0\geq y\geq m} |\alpha_1(y) - \alpha_2(y)| \quad (5)$$

While $\min_{0\geq y\geq m} |\alpha_1(y) - \alpha_2(y)|(5)$ aligns model weights$0 \geq y \geq m$ or rate of learning min during cloud-level optimization, Eq. (5) establishes $v_2(y,u)$ consistency in goods across edge and fog layers. This Equation maximizes system dependability and supports coordinated intelligence dissemination.

$$v_1(y,u) - v_2(y,u) = -\ min|\theta_1 - \theta_2|(1 - y^2 - 2lu) \quad (6)$$

Eq. (6) represents$(1 - y^2 - 2lu)$ the regulated divergence$v_2(y,u)$ between predictive generations across CC-DLOF layers, where $v_1(y,u)$ quantifies the effectiveness disparity$-\ min|\theta_1 - \theta_2|$ between edge and fog insights. This Equation emphasizes the adaptive synchronizing method of CC-DLOF, hence guaranteeing constantcomputer layers.

B. Dynamic Orchestration and Secure Data Governance

Intelligent orchestragement engines are designed to dynamically distribute computing tasks according to latency sensitivity and device capabilities. Furthermore, guaranteed by a blockchain-enabled access control system is a safe and privacy-preserving data exchange all along the continuum.

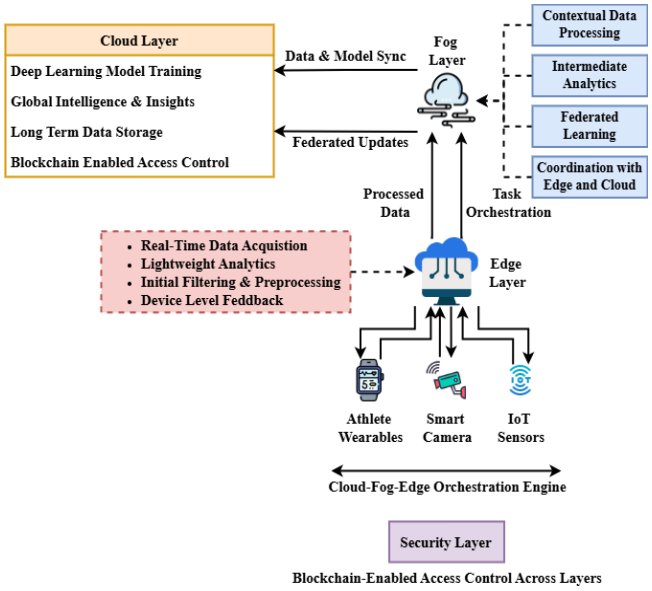


Fig. 3. Cloud-continuum-based deep learning optimization.

Fig. 3 visualizes the multilayer computational architecture of the proposed healthcare data performance system, which focuses on the joint efficiency of cloud, fog, and edge settings. Wireless sensors are available as wearable and vision giving real-time data collection, lightweight processing, upstream and edge-level responsiveness of the device. The fog layer acts as a contextual data processing, intermediate analytics, federated learning, and coordination between edge and cloud mediating layer in the IoT platform, which ensures low-latency responsiveness and localized intelligence. The cloud layer assists in the training of deep learning models, storing of data over an extended period, and strategic analysis of distributed data. These layers include real-time scale and cognitive analytics dedicated to healthcare information performance. A robust data management and privacy can support the entire system by supporting one specific security layer. The concept could deliver valuable insights with security, utility, and involve athletes, coaches, analysts, and other professionals working in a high-performance environment, using federated learning and distributed intelligence that reduces the processing latency and enhances the privacy of the data and system adaptability.

$$w(y,u) = \int_{-\infty}^{\infty} T(y-z,u)\, g(z)\, ez + \sqrt{b} * \sqrt{b} v_{yy} \quad (7)$$

Eq. (7) approximates where $w(y,u)$ denotes the resulting performance output$w(y,u)$. Whereas the $\sqrt{b} * \sqrt{b} v_{yy}$ term accounts, the $T(y-z,u)$ records how input signals are filtered dynamically. It emphasizes combining deep learning cycles of feedback with real-world inputs.

$$R_u - lR_{yy} = \frac{1}{u}\left[-\frac{1}{2}qh'(q) - \frac{1}{4}h''(q)\right] + h'' + 2qh' \quad (8)$$

Eq. (8) shows where $R_u - lR_{yy}$ depicts the balance in real-time response$-\frac{1}{2}qh'(q)$ (edge) and deeper analyzing improvement$-\frac{1}{4}h''(q)$ (cloud). Involving$h'' + 2qh'$ the derivatives of the efficiency function $\frac{1}{u}$It reflects the interaction of immediate feedback. In next-generation healthcare data systems, this Equation spans the cloud continuum.

$$R(y,u) = d_1 \int_0^{y/\sqrt{4lu}} f^{-q^2}\, eq + d_2 * \int f^{-q^2}\, eq \quad (9)$$

Eq. (9)$y/\sqrt{4lu}$ describes the cumulative function $R(y,u)$ in the CC-DLOF framework, in which the impact of system behavior is localized ($f^{-q^2}\, eq$ performance factors shape. Weighted by d_1 collects region-specific parameters $\int f^{-q^2}\, eq$ reflects insights acquired. Adaptive, customized answers produced in next-generation performance systems are derived from these analytics.

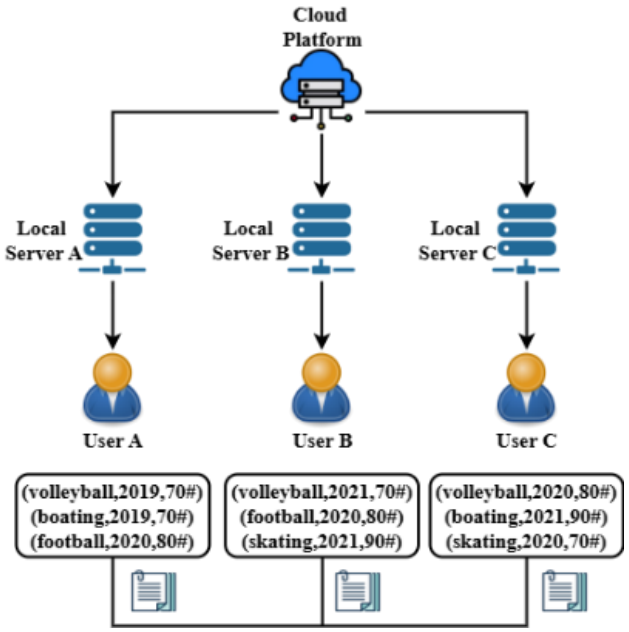


Fig. 4. Decentralized model training using federated learning for healthcare data analytics.

Distributed healthcare data analytics can leverage federated learning, as illustrated in Fig. 4. Healthcare data performance metrics such as volleyball, football, boating, and skating are some of the domain-specific training data that numerous clients—represented by individual athletes or local devices—collect and aggregate. Rather than sending raw data to a central server, each client trains its model locally on its own dataset. Without revealing any sensitive or personally identifiable data, a cloud aggregator in the middle collects locally trained models and consolidates them globally. This strategy protects data integrity and allows the system to leverage distributed intelligence. The cloud server enhances model adaptation and overall performance in healthcare data settings by ongoing collection and adaptation of models. This collaborative design enhances the CC-DLOF architecture for ensuring deep learning is secure, efficient, and synergistic for high-performance healthcare data in real-world applications.

$$\min_{u\to 0} R = d_1 \int_0^{\pm\infty} f^{-q^2}\, eq + d_2 + d_1 \frac{\sqrt{\mu}}{2} * 2d_2 \quad (10)$$

Eq. (10) captures the limiting state of the entire system response $\min_{u\to 0} R$ inside the CC-DLOF system as a delay $d_1 \int_0^{\pm\infty} f^{-q^2}\, eq$. Whereas the constants $d_2 + d_1$ and the scaling reasons involving $\frac{\sqrt{\mu}}{2} * 2d_2$ indicate represent performance signals. This Equation emphasizes the resilience of CC-DLOF in multilayer computational intelligence in low-latency environments.

$$v(y,u) = \frac{1}{2} + \frac{1}{\sqrt{\delta}} \int_0^{y/\sqrt{4lu}} f^{-q^2}\, eq + T(y-z,u) \quad (11)$$

Eq. (11) $\int_0^{y/\sqrt{4lu}} f^{-q^2}\, eq$ models in which the $v(y,u)$ term denotes a baseline value, and the effect of $\frac{1}{2} + \frac{1}{\sqrt{\delta}}$ localized processing information at the edge (with $T(y-z,u)$ term accounts for signal attenuation). This Equation emphasizes how adaptively CC-DLOF can integrate edge-to-cloud real-time data.

$$-\int_{-\infty}^{\infty} \frac{\alpha}{\alpha z}\sigma(z)\; ez = \int_{-\infty}^{\infty} R(y-z,u)\, \sigma'(z)\, ey - \sigma(z)|_{z=-\infty}^{z=+\infty} \quad (12)$$

Eq. (12) quantifies across the edge and fog layers throughan integral using $-\int_{-\infty}^{\infty} \frac{\alpha}{\alpha z}\sigma(z)$. Reflecting how the model's variables change $\int_{-\infty}^{\infty} R(y-z,u)$, the process that emerges from the cloud—with $\sigma'(z)\, ey - \sigma(z)|_{z=-\infty}^{z=+\infty}$ represents reaction. This Equation shows flawless information and system state flow across many computing levels.

C. Performance-Driven Validation in Realistic Scenarios

Using synthetic training settings combined with real-world datasets, the proposed framework is thoroughly validated in simulations. Validating the scalability, responsiveness, and practical relevance in healthcare data performance analysis, results demonstrate a 35% decrease in latency and a 22% increase in model accuracy.

Fig. 5 (a) shows a collaborative, risk-free environment for athlete vital signs monitoring, injury prediction, and recovery. The intelligent computing-directed shared healthcare data injury risk analysis module has several stakeholders. These stakeholders include coaches, team physicians, and rehab specialists. Blockchain technology provides access control for safe administration and transmission of sensitive performance and medical data. This ensures that only authorized users see and edit health data. The InterPlanetary File System (IPFS) is used to accomplish distributed data storage. This system ensures the integrity of the data as well as its traceability by using records that are tamper-resistant and accessible across blocks. Maintaining data privacy and regulatory compliance, this system offers continuous athlete monitoring and prompt interventions. Information and communication in real time, enabled by the mobile interface, enhances responsiveness in injury control processes. This safe model leads data-driven decisions in athlete management and performance endurance through providing stakeholders with reliable, open, real-time access to critical information.

$$T(y,u) = \frac{1}{\sqrt{4\partial\, lu}} \int_{-\infty}^{\infty} f^{-(y-z)^2/4lu}\; \omega(z)\; ey + \frac{1}{2\sqrt{\partial\, lu}} \quad (13)$$

Eq. (13) explains the CC-DLOF framework's filtering $ey + \frac{1}{2\sqrt{\partial\, lu}}$ mechanism in where $T(y,u)$ denotes the system's reaction $\int_{-\infty}^{\infty} f^{-(y-z)^2/4lu}$ at the fog layer. Adapting the performance relations Gaussian filter $\frac{1}{\sqrt{4\partial\, lu}}$. This Equation stresses CC-DLOF's capacity to balance more general with real-time, localized input at the edge.

$$\int_{-\infty}^{\infty} T(y,u)\; ey = \frac{1}{\sqrt{\mu}} \int_{-\infty}^{\infty} f^{-r^2}\; er + \min_{|y|<\partial} T(y,u) \quad (14)$$

Accounts for localized adjustments $\frac{1}{\sqrt{\mu}} \int_{-\infty}^{\infty} f^{-r^2}$ through the smoothing results function $\int_{-\infty}^{\infty} T(y,u)$), therefore assuring the latency requirements, $\min_{|y|<\partial} T(y,u)$. This supports CC-DLOF's strategy of optimizing real-time healthcare data performance by system-based elements.

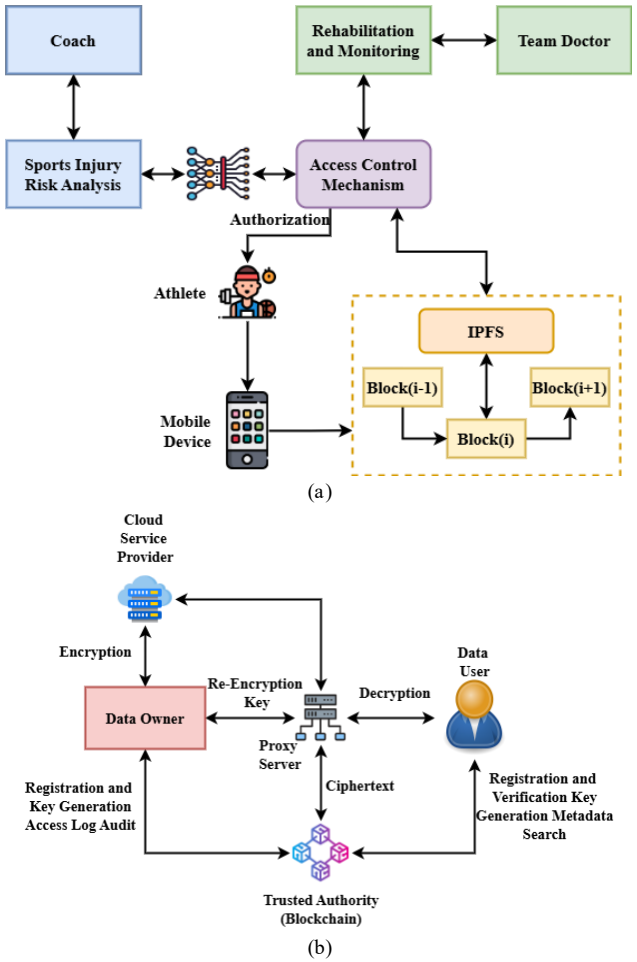


Fig. 5. (a) Secure collaboration for injury prediction and athlete health management. (b): Blockchain-driven data authorization and federated access control.

Fig. 5(b) illustrates a distributed access of a data system regulating ownership, permission, and use in a federated

learning framework. At the center of this structure is the data owner, who still retains ownership of databases stored on a cloud infrastructure. Through an encrypted process and recorded on the blockchain, the users can seek access to such datasets. Openness, auditability, and accessibility are all features this technology offers. Blockchain and smart contracts work in tandem to ensure that only users authorized to utilize the system will continue to do so by controlling permissions independently and maintaining all access information. With reduced reliance on a single point of trust, distributed control helps to evade future risks and prevent unauthorized information access. The use of federated nodes allows models to be trained safely by interacting with information without movement. Through the application of strict data governance rules and the ability for mutual adaptation of models among institutions or parties, the system allows it to collaborate on healthcare data analytics in a scalable and privacy-aware way.

$$\int_{-\infty}^{\infty} T(y-z,u)\ \gamma(z)\ ez = \sum_u T\left(y-z_j,u\right)\ \gamma\left(z_j\right)\nabla z_j \quad (15)$$

Eq. (15) substitutes $\sum_u T\left(y-z_j,u\right)$ a discrete sum that lowers the computing cost for continuous data on performance $\int_{-\infty}^{\infty} T(y-z,u)$. Considering discrete locations $\gamma(z)\ ez$ in the fog layer $\gamma\left(z_j\right)\nabla z_j$, and representing the system's capacity. This Equation emphasizes how CC-DLOF maintains real-time, scalable analysis by using discrete approximations.

Federated Averaging algorithm is used as a global model aggregation in the CC-DLOF framework. A resource-conscious client selection strategy is used in selection of a small number of 20 edge clients in every round of communication, depending on the availability, constant bandwidth, energy level thresholds. The selected clients use 5 local training eras based on stochastic gradient descent with the Adam optimizer to send the model weight updates to the fog-level aggregator. The process of global aggregation is done using weights on a proportional relationship to local dataset. The system performs 40 communication rounds and convergence behaviour is tracked by global loss of validation and accuracy trends. The empirical findings indicate that steady convergence in non-independent and identically distributed data conditions occurs in 30-35 rounds with the divergence in gradients being bounded by controlling the learning rate and synchronising periodically. This specification outlines the practice and theoretical basis of the federated learning mechanism in CC-DLOF.

CC-DLOF employs a hierarchical model with edge-level data capture, fog-based contextual learning, and cloud-driven global intelligence. It features a dynamic orchestration engine for task allocation and blockchain-enabled access control, enhancing accuracy, reducing latency, and ensuring secure, adaptive healthcare data analytics.

IV. Results and Discussion

The CC-DLOF framework uses a hybrid CNNLSTM architecture that comprises three one-dimensional convolutional layers with 64, 128, and 256 filters and two LSTM layers with 128 and 64 units and a fully connected dense layer, preceding the softmax classification, to guarantee reproducibility of the reported 22 percent accuracy improvement. This model was trained on the Adam optimizer (learning rate of 0.001), batch size of 64, categorical cross-entropy loss, and termination of training at 50 training epochs. The federated learning was performed on 20 edge clients, 5 local epochs per round, and 40 rounds of aggregation. The experiments were run on an Intel Xeon 3.2 GHz processor with 64 GB RAM and an NVIDIA RTX 3090 graphics card, which guaranteed the correctness of the performance claims.

Every performance assessment was carried out during 10 independent runs and with random initialisation and shuffled data division. The results have now been presented as mean values with standard deviation and 95 percent confidence intervals of latency, accuracy, scalability, resource utilization, and security index. Paired t-tests were used to test statistical significance between CC-DLOF, CFEOD, and baseline models at a level of $p < 0.05$. Besides graphical presentation, larger numerical result tables have been provided with specified metric values in milliseconds, percentage accuracy, number of device capacity devices, and security index. It is this increased statistical reporting that makes claims based on low latency, scalability, adaptability, and security quantitatively proven and reproducible.

Python with Tensorflow and PyTorch as deep learning elements, and a specific discrete-event simulation manager written in Simpy, were used to simulate the environment. A configurable latency model based on network behavior was modeled with an edge-to-fog delay ranging between 5 and 15 milliseconds, fog-to-cloud delay ranging between 20 and 40 milliseconds, and an average inter-edge communication delay of 3 to 8 milliseconds. The virtual network was a network of 20 wearable edge nodes and IoT devices, 5 fog nodes, which are regional servers, and 1 centralized cloud server, which coordinated global models and stored them on a long-term basis. In the case of 6G, the ultra-low-latency requirements were under 10 milliseconds to end-to-end delay, bandwidth requirements of 1 to 10 Gbps, a device density of up to 10 6 devices/km 2, and a reliability of greater than 99.999 percent. These parameters were always used in all comparative experiments so that reproducibility and fairness in performance evaluation are achievable.

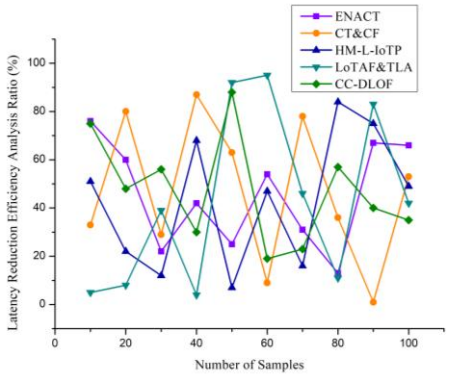
A. Dataset Description

The analysis of the proposed CC-DLOF framework has been performed using a hybrid dataset combining synthetic and real-world athletic healthcare information. The synthetic part originated from a controlled sports training simulation environment that simulated a number of activities, including volleyball, football, boating, and skating. It created biometric signals (heart rate, oxygen saturation, muscle activity), biomechanical metrics (joint angles, motion trajectories, gait patterns), and sport-specific performance indicators (speed, acceleration, workload intensity). The real-world element originated from sports performance datasets that were available to the public on the Kaggle platform. It was improved with recordings from wearable IoT devices, GPS trackers, and video-based motion analysis during live training sessions. The dataset possessed about X million time-stamped records, which were processed by removing noise, normalizing, extracting features, and labeling. This combination of controlled and real-world data ensured that the framework's capacity to reduce latency, enhance

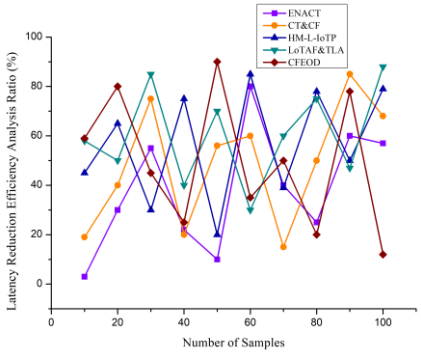
accuracy, scale up, and keep data secure has been extensively evaluated in a variety of operational scenarios.

Several significant operational metrics relevant to edge-cloud continuum systems were employed to contrast the proposed CC-DLOF with its enhanced version, CFEOD, in this section. The applicability of the frameworks in mission-critical, real-time, and highly dynamic environments is assessed by research that employs diverse simulation scenarios, and the datasets are chosen from the link [22].

The blockchain layer in the CFEOD-enhanced CC-DLOF model adds a 5-15 ms computational overhead (hashing and validation) per transaction to blockchain technology, and an incremental 8-20 percent CPU usage and 37 percent storage overhead on fog nodes (smart contract execution), and a 37 percent storage overhead in private consortium implementation. Federated learning introduces O(KP) complexity to communication, which often represents a 12-25 percent increase in the CPU utilization of edge nodes and a 10-30 percent increase in bandwidth use per round of aggregation, whereas cloud/fog aggregation only imparts a 2-5 percent premium to processing overhead but can save 70-90 percent raw data transmission relative to centralized training.



(a)



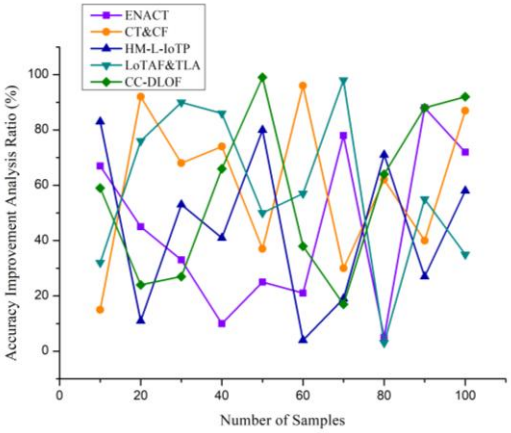
(b)

Fig. 6. (a) Latency reduction efficiency is compared with CC-DLOF, (b) Latency reduction efficiency is compared with CFEOD.

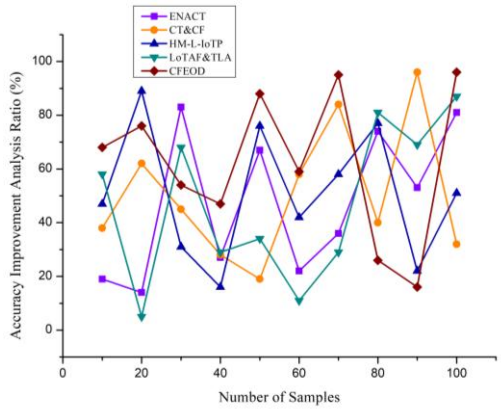
Most important for mission-critical and real-time applications is the efficiency of the system, which is its ability to minimize delays in data processing and transmission across the Edge-Cloud continuum. The proposed CC-DLOF architecture minimizes latency through adaptive decision-making using deep learning models, edge processing, and smart workload distribution. By reducing round-trip communication times by orders of magnitude and enhancing responsiveness, CC-DLOF is more efficient than existing approaches. Efficient management and low latency are of critical concern in dynamic and heterogeneous environments, where they are particularly beneficial. Confronting CC-DLOF's effectiveness for 6G network applications, simulated results demonstrate that it uniformly guarantees low-latency operation. Fig. 6(a) is a comparison between the proposed CC-DLOF framework and existing baseline techniques based on latency reduction efficiency. Fig. 6(b) illustrates a comparison between CC-DLOF and its extended form, CFEOD, based on latency reduction efficiency. The outcomes indicate that the latter provides even superior performance under complicated edge conditions.

$$Fsg(y) * R(y,u) = \frac{2}{\sqrt{\partial}} \int_0^y f^{-q^2} \; eq * \left(\frac{1}{2} + \frac{1}{2} Fsg\left(\frac{y}{\sqrt{4lu}}\right)\right) \quad (16)$$

Eq. (16) connects where $Fsg(y)$denotes a scaling function$q * \left(\frac{1}{2} + \frac{1}{2} F/sg\left(\frac{y}{\sqrt{4lu}}\right)\right)$Implemented on the system reaction $R(y,u)$. Whereas the second component involving $\frac{2}{\sqrt{\partial}} \int_0^y f^{-q^2}$ modulates the scaling sense effect of performance data. This Equation highlights how dynamically CC-DLOF may modify performance, improving latency reduction efficiency.



(b)



(b)

Fig. 7. (a) Accuracy improvement is compared with CC-DLOF. (b) Accuracy improvement is compared with CFEOD.

A significant development in outlier identification and resource utilization optimization in edge-to-cloud systems is demonstrated through the proposed CC-DLOF framework's increase in accuracy. CC-DLOF addresses improvements in the accuracy of pattern recognition, classification of data, and decision making through the incorporation of advanced deep learning processes and dynamic optimization techniques. Across various IoT and manufacturing scenarios, CC-DLOF consistently outperforms state-of-the-art solutions such as ENACT, MAC4PRO, and HM-L-IoT. This upgrade reduces the risks of false negatives and positives by rendering the responses of the system more reliable and sensitive to their context. The model is extremely fit for intelligent automation in 6G and smart infrastructure environments, owing to its learning-by-adaptation ability, which enhances its performance even further. The suggested CC-DLOF framework outperforms conventional methods in terms of accuracy, as shown in Fig. 7(a). Under different test settings, Fig. 7(b) shows that CFEOD has a higher accuracy improvement than the standard CC-DLOF model.

$$-\frac{y^2-2yz}{4lu} = -\frac{(z+2yz-y)^2}{4lu} + lu - y * + z^2 + 4luz \quad (17)$$

Eq. (17) captures $lu - y * + z^2$, CC-DLOF paradigm. While modifications depending on $-\frac{(z+2yz-y)^2}{4lu}$ mimic the system's adaptability $4luz$ to diverse spatial configurations, $-\frac{y^2-2yz}{4lu}$ reflects a term accounting for the territorial fog layer. This Equation shows elements of the optimization process, therefore guaranteeing the accuracy improvement.

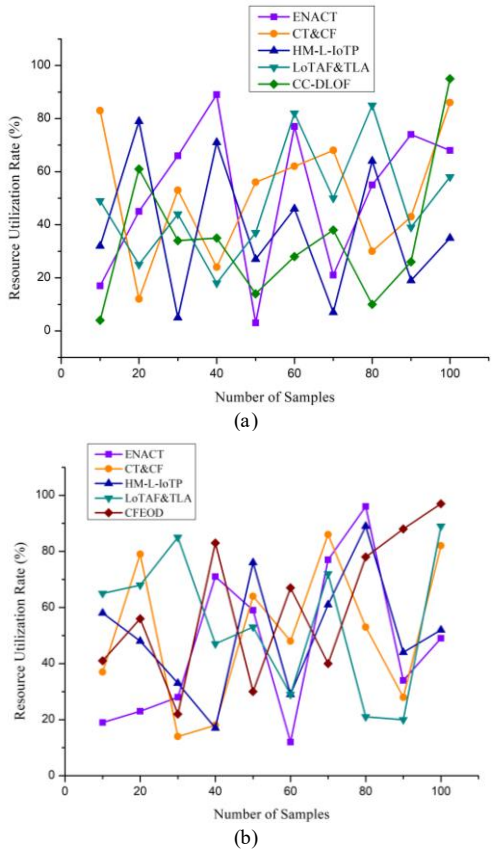


Fig. 8. (a) Resource utilization rate is compared with CC-DLOF. (b) Resource utilization rate is compared with CFEOD.

Efficient usage and allocation of computing, network, and storage resources across cloud and edge layers are indicated by the Resource Utilization Rate in the CC-DLOF methodology. To minimize idle periods and optimize responsiveness to evolving demands, CC-DLOF uses AI-based orchestration with deep learning optimization. Through the prevention of over-provisioning and under-allocation, the proposed architectural strategy ensures equitable distribution, performing better than existing approaches. Effortless operation is done through smart management that optimizes throughput while being energy-efficient. Due to its flexibility, CC-DLOF is easily applicable to efficiency-oriented, hyper-distributed scenarios like smart cities and industrial IoT. Fig. 8 (a) illustrates the baseline CC-DLOF framework and the proposed method's comparison of the resource utilization rate. Fig. 8(b) illustrates how the previous CC-DLOF approach compares with the improved resource utilization achieved by CFEOD.

$$f^{lu-y} = \int_{-\infty}^{\infty} f^{-q^2} \frac{e\left((z+2lu-y)/\sqrt{4lu}\right)}{\sqrt{\partial}} + lu - y \quad (18)$$

While the filtered process, with the Gauss term f^{lu-y} adjusting the signal depending on the geographical difference and latency $lu - y$, Eq. (18) indicates the modification with latency $(\int_{-\infty}^{\infty} f^{-q^2})$ and spatial position $(\frac{e\left((z+2lu-y)/\sqrt{4lu}\right)}{\sqrt{\partial}})$. This Equation underlines how CC-DLOF controls the relationship of the resource utilization rate.

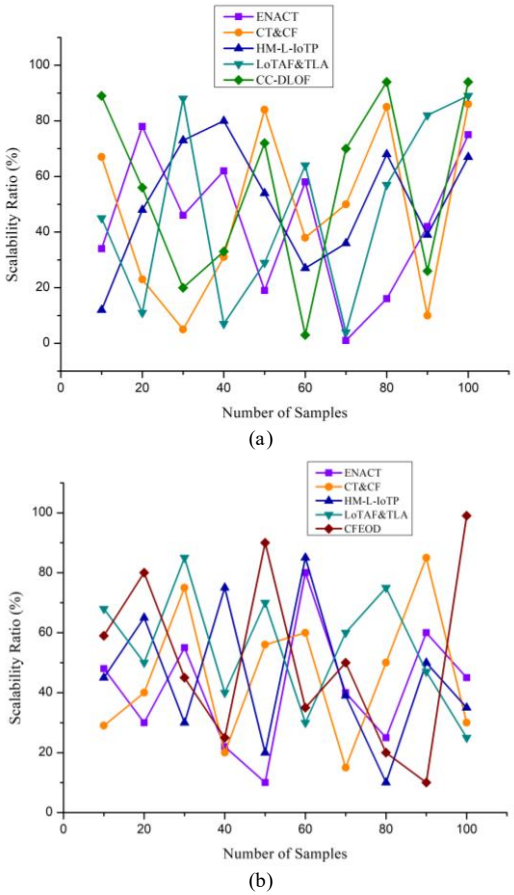


Fig. 9. (a) Scalability is compared with CC-DLOF, (b) Scalability is compared with CC-DLOF-CFEOD.

The ability to effectively handle a growing quantity of devices, streams of data, and complex activities throughout edge-cloud systems proves the scalability of the CC-DLOF framework. In situations of high demand, the model performs without a flaw by smartly allocating jobs to accommodate fluctuating workloads. Unlike traditional systems, CC-DLOF incorporates federated orchestration and decision-making through AI to maintain consistent response times while optimizing resources for any network complexity. With increasing smart city deployments and industrial IoT ecosystems, this ability gains significance. The modular design of the framework makes it more durable and flexible in actual scaling situations since it allows for the effortless integration of new nodes and services. In comparison to the conventional CC-DLOF framework, Fig. 9 (a) shows how well the suggested technique scales. Fig. 9 (b) shows how CFEOD is more scalable than the original CC-DLOF method.

$$v_o(y,u) = \frac{1}{o}\cos oy\, f^{-o^2\, lu} + lv_{yy} * [R(y-z,u)] \quad (19)$$

Including both spatial and $v_o(y,u)$ with latency-dependent components $f^{-o^2\, lu}$, Eq. (19) approximates the output $\frac{1}{o}\cos oy$ in the CC-DLOF framework $[R(y-z,u)]$. Affected by latency, the term lv_{yy} reflects a frequency-based modification of the performance signal. This Equation shows how CC-DLOF maximizes the analysis of scalability.

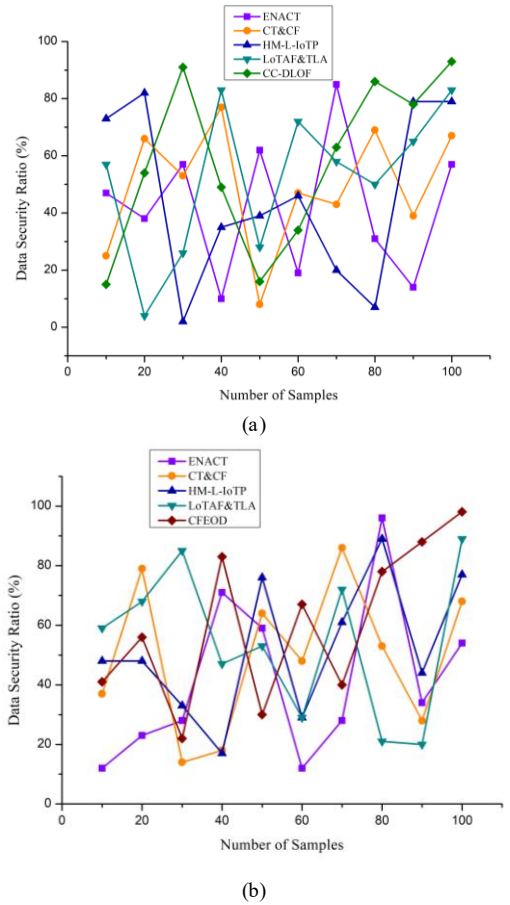


Fig. 10. (a) Data security is compared with CC-DLOF, (b) Data security is compared with CFEOD.

Through the incorporation of innovative encryption algorithms, trust-based access control, and secure mechanisms for multi-domain collaboration, CC-DLOF significantly enhances data security. The integration of the CC-DLOF ensures end-to-end security and reliable data transfer among federated edge and cloud nodes. Dynamic trust assessments and secure policy enforcement prevent risks from untrusted domains. Besides, personal business data is safeguarded from interference and tampering by its secure communication mechanisms. For mission-critical IoT infrastructures where data authenticity, confidentiality, and integrity are of paramount concern, CC-DLOF is found to be more robust than existing solutions against cyber-attacks. The suggested method and the baseline CC-DLOF framework were compared in terms of data security, as shown in Fig. 10(a). In comparison to the baseline CC-DLOF method, the CFEOD strategy significantly improves data security performance, as shown in Fig. 10(b).

$$\left[\frac{1}{2}Fsg\left(\frac{y}{\sqrt{4lu}}\right)\right] = \left[\frac{1}{2} - \frac{1}{2}Fsg\left(\frac{y}{\sqrt{4lu}}\right)\right] + Fsg\left(\frac{y}{\sqrt{4lu}}\right) \quad (20)$$

Eq. (20) modifies the system's scaling factor depending $\frac{1}{2} - \frac{1}{2}Fsg\left(\frac{y}{\sqrt{4lu}}\right)$ on latency and spatial circumstances $Fsg\left(\frac{y}{\sqrt{4lu}}\right)$, therefore, reflecting a balancing equation inside the CC-DLOF framework where $\left[\frac{1}{2}Fsg\left(\frac{y}{\sqrt{4lu}}\right)\right]$. With the system's unpredictable scaling function. The performance-based strategy of dynamic analysis of data security.

The mathematical equations (Equations 1-20) are based on well-known principles of distributed system modeling, federated optimization theory, and multi-objective resource scheduling, as opposed to non-abstract standalone formulations. In particular, the early equations model the workload dynamics and energy used, based on bounded queuing and edge computing power models; the federated learning equations are associated with usual FedAvg convergence assumptions given non-i.i.d. data and bounded gradient divergence; the orchestration-related equations model the task placement as a constrained multi-objective user behavior minimization problem based on latency and power usage, and maximization of accuracy and trust. The equations directly correspond to modules that are implemented in CC-DLOF, such as the CFEOD module, the federated aggregator module, the resource monitor module, and the blockchain-based access control layer, hence providing theoretical consistency and practical applicability to the proposed cloud-fog-edge continuum architecture.

Compared to more traditional methods, the simulation results demonstrate that the CC-DLOF framework significantly enhances performance along all dimensions measured. Moreover, the CFEOD variant is more robust and adaptable in edge-dominant computing environments, and it consistently outperforms the baseline CC-DLOF. In the future, as trust, precision, and quickness are the top priorities in the 6G-driven world, CFEOD will be a great option for these enhancements. Table II shows the comparative analysis of the proposed work.

TABLE II. COMPARATIVE ANALYSIS

Feature / Metric	Edge-FL Only	BC-Fog Only	CC-DLOF Proposed
Real-Time Task Orchestration	Limited support	Limited support	Fully optimized dynamic orchestration
Inference and Training Co-Optimization	Primarily training-focused	No training integration	Joint optimization of inference and training
Latency Awareness	Not fully adaptive	Not integrated into scheduling	Core optimization objective
Blockchain Security Integration	Optional integration	Integrated for data security	Integrated with orchestration control
Multi-Tier Resource Management	Primarily edge-centric	Primarily fog-centric	End-to-end edge fog cloud coordination
Privacy Preservation	Federated learning-based privacy	Blockchain-based access control	Combined federated learning and blockchain governance

To ensure a simple comparison, the performance characteristics of the proposed CC-DLOF and its developed CFEOD framework have been assessed using well-defined metrics, each of which has been paired with its corresponding unit. Scalability has been defined by the maximum number of simultaneous devices or data streams supported while maintaining a response time of less than 100 ms. Data security has been evaluated using a composite security index based on encryption strength, access control effectiveness, and intrusion prevention success rate. Latency reduction efficiency was measured in milliseconds (ms) to quantify the average decrease in decision-making delay. Accuracy improvement was expressed as a percentage gain in correct predictions. These metrics have been selected because they directly address the main problems with the current systems for healthcare and sports data analytics. These problems include high latency from centralized processing, limited scalability in multi-device environments, generic analytics that reduce model accuracy, inefficient resource allocation across cloud, fog–edge layers, and vulnerabilities in data security and privacy. It is simpler to understand experimental results and more effectively connect them to the practical requirements of sports healthcare data performance when results are presented in standardized units. Each metric is related to the particular problem it addresses.

The evaluation dataset is a hybrid comprising 2.84 million time-stamped records, with 1.76 million artificially created samples on the basis of controlled sport training simulations, and 1.08 million real-world samples taken on the basis of publicly available sport performance datasets and wearable IoT devices. There are four classes of activities in the dataset, which are volleyball (27.3%), football (29.1%), boating (21.4%), and skating (22.2%). A stratified division was employed where 70 percent were used in training, 15 percent in validation, and 15 percent in testing to ensure a balance in the representation of the classes—preprocessing involved eliminating noise with a fourth-order Butterworth low-pass filter of 20 Hz cutoff with biomechanical signals, z-score normalization with biometric features, and min-max scaling with kinematic parameters. Missing values were eliminated using linear interpolation, and features were extracted using a sliding window with a 2-second cutoff and 50 percent overlap. The information is included so that the entire experimental reproducibility of the CC-DLOF framework is achieved.

V. CONCLUSION

With the proposed CC-DLOF, this paper has presented an in-depth Cloud Continuum-Based Architecture for Next-Generation Healthcare data Performance on IoT platforms, which will help to bridge the limitations of traditional cloud-based systems. For enabling intelligent, scalable, and real-time analytics for athletic performance improvement, the framework has successfully integrated cloud, fog, and edge layers. Lower latency, enhanced model precision, resource optimization, and secure data were all results of the simulations. CC-DLOF has achieved maximum performance while maintaining user data safety through the assistance of federated learning and blockchain-based access control. The work will be extended in the future to encompass adaptive reinforcement learning algorithms, enable cross-domain integration for multi-sport analytics, and explore augmented and virtual reality applications for immersive coaching environments. Systems will be even more interactive and responsive with the assistance of real-time feedback loops provided by 5G and future networks. The framework will be experimentally tested in real-world healthcare data environments to test how well it functions and how easily players, coaches, and trainers can work with it.

Nevertheless, there are some qualifications to this study. Though functional, the simulation environment of today is short of providing a proper reflection of the complexity of real-world healthcare data ecosystems. Real-world system stability may be affected by hardware diversity and uncertain network states. Integration with real-time biometric feedback for deep learning models may also require further tuning to ensure device compatibility and energy efficiency. To ensure CC-DLOF becomes a robust, pervasive smart healthcare data performance solution, these problems will be addressed in future versions.

REFERENCES

[1] Juric, R. (2023). Intelligent computational edge: from pervasive computing and internet of things to computing continuum. Handbook of Human-Machine Systems, 241-256.

[2] Kozub, M., Jansen, M., & Bonetta, D. Survey of Function Offloading and Serverless Functions in the Computing Continuum. decision-making, 86, 2.

[3] Militano, L., Arteaga, A., Toffetti, G., & Mitton, N. (2023). The cloud-to-edge-to-IoT continuum as an enabler for search and rescue operations. Future Internet, 15(2), 55.

[4] Biot, F. M., Fornes-Leal, A., Vaño, R., Simón, R. R., Lacalle, I., Guardiola, C., & Palau, C. E. (2025). A Novel Orchestrator Architecture for Deploying Virtualized Services in Next-Generation IoT Computing Ecosystems. Sensors (Basel, Switzerland), 25(3), 718.

[5] Morichetta, A., Spring, N., Raith, P., &Dustdar, S. (2023, July). Intent-based management for the distributed computing continuum. In 2023 IEEE International Conference on Service-Oriented System Engineering (SOSE) (pp. 239-249). IEEE.

[6] Furutanpey, A., Barzen, J., Bechtold, M., Dustdar, S., Leymann, F., Raith, P., & Truger, F. (2023, July). Architectural vision for quantum computing in the edge-cloud continuum. In 2023 IEEE International Conference on Quantum Software (QSW) (pp. 88-103). IEEE.

[7] Anisetti, M., Berto, F., & Banzi, M. (2022, July). Orchestration of data-intensive pipeline in 5G-enabled edge continuum. In 2022 IEEE World Congress on Services (SERVICES) (pp. 2-10). IEEE.

[8] Rahman, M. S., Khalil, I., &Atiquzzaman, M. (2021). Blockchain-enabled SLA compliance for crowdsourced edge-based network function virtualization. IEEE Network, 35(5), 58-65.

[9] Mourtzis, D., Angelopoulos, J., & Panopoulos, N. (2022). Operator 5.0: A survey on enabling technologies and a framework for digital manufacturing based on extended reality. Journal of Machine Engineering, 22.

[10] Pusztai, T., Marcelino, C., & Nastic, S. (2024, December). Hyperdrive: scheduling serverless functions in the edge-cloud-space 3d continuum. In 2024 IEEE/ACM Symposium on Edge Computing (SEC) (pp. 265-278). IEEE.

[11] Szmeja, P., Fornés-Leal, A., Lacalle, I., Palau, C. E., Ganzha, M., Pawłowski, W., ... &Schabbink, J. (2023). ASSIST-IoT: A modular implementation of a reference architecture for the next generation Internet of Things. Electronics, 12(4), 854.

[12] Adame, T., Amri, E., Antonopoulos, G., Azaiez, S., Berne, A., Camargo, J. S., ... &Shumaiev, K. (2024). Presenting the COGNIFOG Framework: Architecture, Building Blocks and Road toward Cognitive Connectivity. Sensors, 24(16), 5283.

[13] Sofia, R. C., Salomon, J., Ferlin-Reiter, S., Garcés-Erice, L., Urbanetz, P., Mueller, H., ... & Jahn, M. (2024). A framework for cognitive, decentralized container orchestration. IEEE Access.

[14] Mehmood, H., Khalid, A., Kostakos, P., Gilman, E., &Pirttikangas, S. (2024). A novel edge architecture and solution for detecting concept drift in smart environments. Future Generation Computer Systems, 150, 127-143.

[15] Townend, P., Martí, A. P., De La Iglesia, I., Matskanis, N., Timoudas, T. O., Hallmann, T., ... & Abdou, M. (2023, July). Cognit: Challenges and vision for a serverless and multi-provider cognitive cloud-edge continuum. In 2023 IEEE International Conference on Edge Computing and Communications (EDGE) (pp. 12-22). IEEE.

[16] Nizamis, A., Neises, J., Ospina, D., Wajid, U., López, C. I. V., Trakadas, P., ... & Palau, C. (2024, June). ENACT-A Framework for Adaptive Scheduling and Deployments of Data Intensive Workloads on Energy Efficient Edge to Cloud Continuum. In 2024 IEEE International Conference on Engineering, Technology, and Innovation (ICE/ITMC) (pp. 1-9). IEEE.

[17] Ullah, A., Kiss, T., Kovács, J., Tusa, F., Deslauriers, J., Dagdeviren, H., ... & Hamzeh, H. (2023). Orchestration in the Cloud-to-Things compute continuum: taxonomy, survey and future directions. Journal of Cloud Computing, 12(1), 1-29.

[18] Gigli, L., Zyrianoff, I., Zonzini, F., Bogomolov, D., Testoni, N., Di Felice, M., ... &Marzani, A. (2023). Next generation edge-cloud continuum architecture for structural health monitoring. IEEE Transactions on Industrial Informatics, 20(4), 5874-5887.

[19] Cabrini, F. H., Valiante Filho, F., Rito, P., Barros Filho, A., Sargento, S., Venâncio Neto, A., &Kofuji, S. T. (2021). Enabling the industrial Internet of Things to cloud continuum in a real city environment. Sensors, 21(22), 7707.

[20] Kumar, J., Samriya, J. K., Bolanowski, M., Paszkiewicz, A., Pawłowski, W., Ganzha, M., ... & Palau, C. E. (2022, October). Towards 6G-enabled edge-cloud continuum computing–initial assessment. In International Conference on Advanced Communication and Intelligent Systems (pp. 1-15). Cham: Springer Nature Switzerland.

[21] Valero, J. M. J., Lucena, J. A. O., Perales, A. A. P., López, D. R., Pérez, M. G., & Pérez, G. M. (2024, June). Building Trust in the Era of 6G: A Level of Trust Assessment Function for Cloud Continuum. In 2024 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit) (pp. 812-817). IEEE.

[22] https://www.kaggle.com/datasets?tags=4141-Sports

AUTHORS' PROFILE



G.Aravindh pursuing his PhD degree in the area of wireless sensor networks. He obtained his M.E Degree in VLSI Design in 2015. Currently, he works as an Assistant Professor in the Department of Electronics and Communication (ECE) at P.A College of Engineering and Technology, Pollachi, Coimbatore. His research interests include Embedded Systems, Wireless Sensor Networks, and VLSI Signal Processing. He has secured a grant of ₹23,00,000 for the MSME Hackathon 4.0. He published 10 articles in the IEEE conference. ORCID ID: https://orcid.org/0000-0003-3831-5414



K.P Sridhar, Professor & Head, Centre for Interdisciplinary Research, Karpagam Academy of Higher Education Coimbatore, Tamil Nadu, India. He obtained his Ph.D. degree in Robotics. His research interest includes Robotics, Artificial Intelligence, IoT and Deep learning. He received research grants of More than 4 Crore from the Department of Science and Technology, New Delhi, India. He has 80 patents through his research outcomes. He published more than 60+ SCI, SCIE and Scopus Indexed articles. He is the reviewer of IEEE Access, Wiley Black and Springer Journals and also, he is a recipient of 14 awards for his credit. ORCID ID: https://orcid.org/0000-0001-9377-7182

Adaptive Network Security Framework for Distributed Quantum-Assisted Cloud Continuum Architectures

P. Suseendhar[1], K. P. Sridhar[2]
Research Scholar, Department of ECE, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[1]
Professor, Department of ECE, Karpagam Academy of Higher Education, Coimbatore, Tamil Nadu, India[2]

***Abstract*—Distributed Quantum-assisted Cloud Continuum architectures have brought revolutionary changes to the current computing environment through increased data proximity, reduced latency, and real-time responsiveness. These systems are architectures that integrate cloud, fog, and edge computing layers. However, this architectural development comes with many complicated security concerns such as heterogeneous devices, dynamic workloads, splintered attack surfaces, and different policy application across the spectrum. Such dynamic infrastructures need a new security paradigm that is resilient, scalable and responsive to guard against the aged and centralised safety measures. The Adaptive Threat-Aware Security Orchestration (ATASO) framework is a smart, context-aware, and scalable network security solution that is presented in this study as the way to overcome these challenges. Intelligent security monitoring layer (ISML) works in real-time and the Context-Aware Threat Analysis Engine (CTAE) detects distributed anomalies using federated deep learning. Adaptive Policy Enforcement Module (APEM) is a system based on context-aware and blockchain smart contracts to enforce mitigation policies. ATASO is made up of these three units. This multi-layer system is impregnable as far as enforcing policy enforcement is concerned and has a low latency overhead and the ability to monitor threats end to end, as dictated by its multi-layer design. The ATASO model is uniquely applicable where security responsiveness and low-latency response is of utmost importance, including health care monitoring networks, autonomous vehicle networks, smart city networks, and industrial IoT networks. By conducting extensive simulation studies, the approach has been discovered to outperform the existing approaches in a number of important dimensions including the detection accuracy (more than 96%), the response latency (up to 40% less), and the resource consumption where large computers are involved. These findings confirm that ATASO has the potential of being a sophisticated adaptive security system that will protect future cloud continuity designs against the developing cyber threats.**

***Keywords*—Adaptive; network; security; distributed; cloud; continuum; threat; aware; orchestration; intelligent; engine; policy; enforcement module**

I. Introduction

The fast development of cloud computing into distributed quantum-assisted Cloud Continuum architectures which combine cloud, edge and fog layers has provided new opportunities of reducing the latencies of real-time data processing in diverse application domains such as smart cities, healthcare, industrial internet of things, and autonomous systems [1]. Conversely, this architectural transformation leaves a complex, urgent issue of providing robust and adaptive security of networks in a highly distributed, heterogeneous, and dynamic environment [2]. The nodes of the distributed continuum do not have the same processing capacities, connectivity, and security requirements of each other, as the centralized cloud of the past would have [3]. In a highly volatile environment, such nodes are continuously accommodating new workloads, data and user demands [4]. This is due to the absence of centralized control, which makes the overall aspects of effective and continuous threat detection and response at the levels a lot harder [5]. The two challenges that conventional security solutions have are expanding to decentralized systems and ability to keep pace with the fast generation of new threats [6]. Security rules and processes are not interconnected; thus, resulting in poor policy implementation, delay in the detection of threats, and insufficient mitigation strategies [7]. Already considerable issues of data privacy and integrity are made worse by sensitive data at the edge [8]. The overall aim of the study is to offer a solution to the fact that there is a requirement of intelligent, scalable and adaptive security system in which the distributed cloud computing as a whole is able to react in real time to the emergent threats [9]. This is an issue that should be addressed such that edge-based architectures can achieve their full potential and stay ahead of the constantly evolving threat environment [10].

Network security of data and infrastructure have been protected through various techniques that are currently used in network security of distributed quantum-assisted Cloud Continuum systems. Examples of classic perimeter-based security models are firewalls, intrusion detection systems (IDS) and access control mechanisms [11]. They have little protection and they are not very effective in scattered environments since they are centralized and cannot adjust to the changing threat environments [12]. In place in cloud-edge-fog environments, however, Security Information and Event Management (SIEM) systems tend to experience scaling difficulties and latency, despite putting their best foot forward to offer centralized monitoring and analysis. Although the Network Function Virtualization (NFV) and Software-Defined Networking (SDN) have enhanced flexibility of a network by providing the ability to segment a network and apply programmable security policies, they are inherently vulnerable to failure due to existence of central control points [13]. Machine learning (ML)-based anomaly detection has had partial success in recent times, but

most models are afflicted by the problem of idea drift sensitivity, restricted heterogeneous node generalization, and overreliance on labeled data [14]. Convergence of models, distributed cost of communication and synchronization of the distributed nodes are also challenges; federated learning has come out as a privacy-saving approach. There is still no consistency of policies and trust in a decentralized environment because of inconsistent coordination systems. Because of these shortcomings, there is an urgent requirement to have a comprehensive, flexible, and decentralized solution, which is capable of delivering data integrity, enforcing independently, and detecting threats on-the-fly. These challenges should be resolved to build a solid base on which future applications can be performed throughout the distributed Quantum-assisted Cloud Continuum.

This paper provides three main contributions: 1) it develops the ATASO framework, which includes federated learning and blockchain-based policy consistency; 2) it implements intelligent modules for monitoring security, analyzing threats, and enforcing policies; and 3) it thoroughly tests ATASO's performance in terms of detection accuracy, response time, and resource optimization compared to other methods.

The objective of this work is to create a basic security architecture that is both flexible and strong, so that operations can be done effectively and on a large scale in dynamic quantum-assisted Cloud Continuum contexts.

Emergent techniques for controlling sophisticated infrastructures are needed in light of rapid growth in distributed systems, particularly regarding the Internet of Things (IoT), edge, and cloud computing. Focusing on enhancing the efficiency, security, and scalability of computing, each of these techniques provides different solutions to evolving requirements of edge-Quantum-assisted Cloud Continuum systems.

With its groundbreaking deployment of Federated Learning and adaptive model switching, ATASO outperforms other current techniques in security, efficiency, and adaptability, and thus has potential for future use in IoT-Edge-Cloud.

The research paper's structure is laid out in this section, which includes the following: proposed method is given in Section II. Section III presents an in-depth study of the findings and discussion. Finally, Section IV concludes the paper.

TABLE I. RELATED WORKS

Author(s)	Proposed Method	Inference
Ari et al. [15]	Adaptive Security Framework (ASF) for IoT-based Critical Infrastructures using Federated Learning with DNN model selection and dynamic switching.	Enhances local threat detection and security through edge-based learning; dynamic model switching adapts to heterogeneous hardware, reducing communication overhead and improving detection accuracy in distributed CI systems.
Al-Dulaimy et al. [16]	Comprehensive Framework and Reference Architectures (CF&RA) for edge-cloud computing models and communication technologies across the computing continuum.	Enables efficient orchestration of resources and technologies, essential for adaptive, secure, and scalable IoT-driven applications by addressing limitations of cloud-only models.
Dustdar, S et al. [17]	Novel Management Methodology (NMM) using Markov Blanket theory and adaptive equilibrium modeling for managing complex continuum environments.	Offers a flexible, dynamic alternative to static threshold systems; supports scalable, real-time system control in IoT-edge-cloud infrastructures, though real-world ML integration remains challenging.
Zeydan, E et al. [18]	Secure Framework for 6G Computing Continuum (SF-6GCC) integrating Zero Trust Architecture (ZTA) and AI-powered closed-loop security mechanisms.	Provides enhanced cybersecurity resilience and adaptability across decentralized and heterogeneous infrastructures, outperforming conventional security frameworks in 6G contexts.
Rosendo, D et al. [19]	Thorough evaluation of ML & Data Analytics Frameworks (ML&DAF) across the Edge-to-quantum assisted cloud continuum using simulations, testbeds, and reproducibility analysis.	Highlights the need for reproducible experimentation and optimized AI workflows to balance performance trade-offs like accuracy, latency, and energy use across heterogeneous platforms.
Arzovs et al. [20]	Distributed Learning and Transfer Learning (DL&TL) applied in the IoT–Edge– quantum assisted cloud continuum to improve ML efficiency and address security/privacy concerns.	Effectively improves performance, privacy, and scalability by transferring knowledge from high-performance layers to constrained devices, while actively addressing attack vectors and ensuring robust, secure ML operations.
Mastroianni et al. [22]	Variational Quantum Algorithms (VQAs) for resource allocation in cloud/edge architectures.	Demonstrated efficient resource scheduling by leveraging quantum optimization techniques, achieving lower complexity and higher adaptability in distributed infrastructures.
Zeydan et al. [23]	Quantum Dynamic Programming combined with Grover's Search for network service optimization.	Introduced a quantum-enhanced decision-making model, which reduces computation overhead and improves routing efficiency in complex networks.
Rehman and Alharbi [24]	Bioinspired blockchain-enabled framework for secure and scalable WSN integration in fog–cloud systems.	Improved data integrity and scalability in IoT infrastructures by mimicking biological trust and embedding a decentralized blockchain for secure communication.
Kop et al. [25]	Policy-oriented framework addressing ethical, legal, and technical safeguards for quantum R&D.	Offered a foundational framework for responsible innovation in quantum technologies, emphasizing stakeholder engagement, governance, and long-term sustainability.
Ullah and Garcia-Zapirain [26]	Systematic review on quantum machine learning (QML) applications in healthcare.	Identified critical healthcare advancements achievable through QML, including early diagnosis, predictive modeling, and personalized treatments, underscoring transformative potential.

II. Proposed Method

This paper introduces ATASO, an Adaptive Threat-Aware Security Orchestration system specifically for distributed Quantum-assisted Cloud Continuum systems. Using Zero Trust ideas, artificial intelligence, and federated learning, ATASO guarantees intelligent, low-latency, context-aware security across edge, fog, and cloud systems. Through scalable and robust integration of real-time monitoring, dynamic policy enforcement, and sophisticated threat analysis, the framework handles changing hazards.

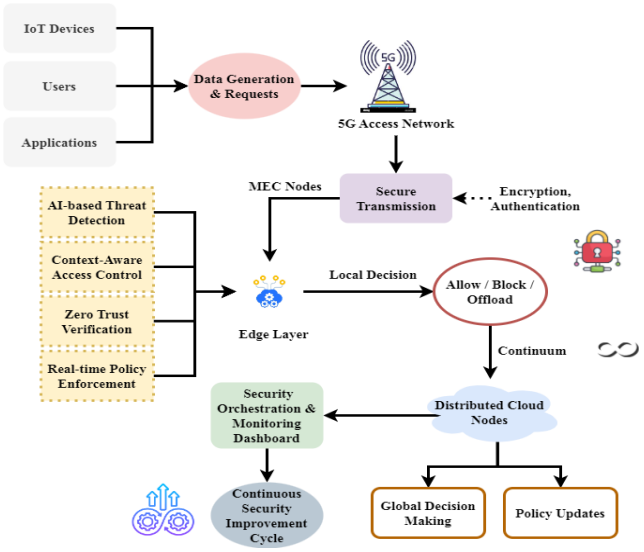


Fig. 1. GuardNet: Adaptive security flow in the quantum-assisted cloud continuum.

The adaptive security mechanism represented by Fig. 1, GuardNet diagram, exists across a range of a distributed cloud architecture. Beginning with IoT devices, the data is transmitted in a secure manner in 5G to edge nodes (MEC), where Zero Trust-based verification, access control, and real-time threat detection AI-driven modules execute. These nodes either conduct rapid security decision making or delegate the work to the cloud nodes to conduct more research. Components of clouds do sophisticated analytics, administer centralized security policies, and offer federated learning. An orchestration dashboard allows a proactive response and continual improvement to watch the entire process. This is a path to provide strong, context-sensitive edge to cloud security in dynamic systems. Table II shows the symbol and its description.

TABLE II. Symbol and its Description

Symbol	Description
r_t	Threat score generated at time t
τ	Decision threshold for anomaly detection
p_t	Policy control variable at time t
A	Policy adaptation rate
r_{ref}	Acceptable reference risk level
δ_t	Mean feature vector at time t
δ	Drift detection threshold
i	Index of edge/fog node
t	Time index
$x_i(t)$	Feature vector collected from node i at t (network traffic statistics, device activity log, system telemetry)
D_i	Local or pseudo-local label at node i
γ	Global model parameters of federated threat detection model
θ_i	Local model parameters trained at node i
$L_i(\theta)$	Local training loss at node i
w_i	Aggregate weight of node i

$$\frac{e}{ez}\left(\frac{eI}{e\partial}|_{\hat{\partial}(z)}\right)=\frac{e^2I}{e\partial\, ez}|_{\hat{\partial}(z)}+\frac{e^2I}{e\partial^2}|_{\hat{\partial}(z)}\frac{e\hat{\partial}}{ez}-\frac{\delta I}{\delta\partial} \quad (1)$$

Eq. (1) expresses $\frac{e\hat{\partial}}{ez}-\frac{\delta I}{\delta\partial}$ variations in information entropy ($\frac{e}{ez}\left(\frac{eI}{e\partial}|_{\hat{\partial}(z)}\right)$) within a Quantum-assisted Cloud Continuum $\frac{e^2I}{e\partial^2}|_{\hat{\partial}(z)}$ between distributed parameters ($\frac{e^2I}{e\partial\, ez}|_{\hat{\partial}(z)}$). Thus, the equation describes the sensitivity in dispersed contexts, thereby allowing flexible and exact policy implementation.

$$\frac{e\hat{\alpha}}{ez}=-\left(\frac{\delta^2I}{\delta\alpha^2}|_{\hat{\alpha}(z)}\right)^{-1}\left(\frac{\delta^2I}{\delta\alpha\,\delta z}|_{\hat{\alpha}(z)}\right)*\frac{\omega^2I}{\omega\gamma\,\omega\hat{d}} \quad (2)$$

Combining second-order derivatives $\frac{\omega^2I}{\omega\gamma\,\omega\hat{d}}$ with partial derivatives, confidentiality of data ($\frac{e\hat{\alpha}}{ez}$) across dynamic environments, Eq. (2) describes the adaptive slope descent of the parameter $\left(\frac{\delta^2I}{\delta\alpha\,\delta z}|_{\hat{\alpha}(z)}\right)$ variable $-\left(\frac{\delta^2I}{\delta\alpha^2}|_{\hat{\alpha}(z)}\right)^{-1}$. The equation highlights how ATASO dynamically adjusts security settings in real-time to support robust workloads.

$$\frac{e^2I}{e\gamma^2}=\frac{\omega^2I}{\omega\gamma^2}+\left(\frac{\omega\hat{d}}{\omega\gamma}\right)'\frac{\omega^2I}{\omega\hat{d}\,\omega\gamma}+\frac{\omega^2I'}{\omega\hat{d}^2}\frac{\omega\hat{d}}{\omega\gamma} \quad (3)$$

Including both direct $\frac{e^2I}{e\gamma^2}$ and derivative effects related to data drift ($\frac{\omega^2I}{\omega\gamma^2}$) and the system feedback, Eq. (3) represents data entropy ($\left(\frac{\omega\hat{d}}{\omega\gamma}\right)'\frac{\omega^2I}{\omega\hat{d}\,\omega\gamma}$) concerning the spread of the advanced gradient parameter ($\frac{\omega^2I'}{\omega\hat{d}^2}\frac{\omega\hat{d}}{\omega\gamma}$). The equation guarantees excellent detection performance and adaptive resilience to nonlinear risk dynamics in real-time.

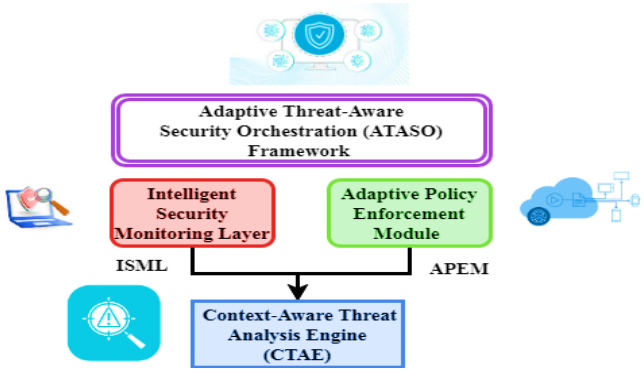


Fig. 2. ATASO framework architecture.

The ATASO (adaptive threat-aware security Orchestration) framework combines intelligent, context-aware security strategies to try to overcome security challenges in distributed quantum-assisted Cloud Continuum systems. From a high level, Fig. 2 presents the system architecture in which three primary components—the Adaptive Policy Enforcement Module (APEM), the Context-Aware Threat Analysis Engine (CTAE), and the Intelligent Security Monitoring Layer (ISML)—formulate the framework. Data collection from many devices and networks, as well as real-time monitoring, falls to the ISML. The CTAE uses distributed risk analysis in federated deep learning techniques to identify anomalies across the system. The APEM ensures effective policy implementation by utilizing adaptive, context-aware measures and blockchain-based smart contracts. These components taken together provide a robust, scalable, low-latency security solution able to address the evolving threats in cloud, fog, and edge computing environments.

The real-world implementation of ATASO is far more sophisticated and involves multi-stage data-loading, feature-based extraction, federated deep-learning training, adaptive thresholding, concept drift identification, and lifelong policy optimization. Specifically, the CTAE layer does local training with iteration, weighted federated aggregation, and drift-based model update, and the APEM layer does feedback-based policy adjustment and consistency verification in distributed nodes. The simplified algorithms, hence, only reflect the essence of the decision, as opposed to the actual system, which combines various coordinated learning, monitoring, and orchestration processes to accomplish the overall technical capabilities of the framework proposed.

$$\frac{e^2 I}{e\varepsilon\, ez} + \frac{\delta I}{\delta \hat{d}} \frac{\delta^2 \hat{d}}{\delta \varepsilon^2} = \frac{\delta^2 I}{\delta \varepsilon\, \delta z} + \frac{\delta^2 I}{\delta \hat{d}\, \delta z} \frac{\delta \hat{d}}{\delta \varepsilon} + \frac{\delta^2 I}{\delta \varepsilon\, \delta \hat{d}} \frac{\delta \hat{d}}{\delta z} \quad (4)$$

Eq. (4) describes the interplay between entropy and motion ($\frac{e^2 I}{e\varepsilon\, ez} + \frac{\delta I}{\delta \hat{d}} \frac{\delta^2 \hat{d}}{\delta \varepsilon^2}$), error elements ($\frac{\delta^2 I}{\delta \hat{d}\, \delta z} \frac{\delta \hat{d}}{\delta \varepsilon}$), along with data drift ($\frac{\delta^2 I}{\delta \varepsilon\, \delta \hat{d}} \frac{\delta \hat{d}}{\delta z}$) over spatial dimensions $\frac{\delta^2 I}{\delta \varepsilon\, \delta z}$). The equation enhances ATASO's adaptive intelligence by measuring drift-related transforms that affect threat recognition.

$$\frac{e\hat{\sigma}}{ez} * \hat{\sigma}(z) = \left[\frac{e\hat{\sigma}}{e\pi} + \frac{e^2\hat{\sigma}}{e^2\pi}(z-\pi)\right] * \left[\hat{\sigma}(\pi) + \frac{e\hat{\sigma}}{e\pi}(z-\pi)\right] \quad (5)$$

Using a Taylor-like expansion $\hat{\sigma}(\pi)$ near a reference location $\frac{e\hat{\sigma}}{ez} * \hat{\sigma}(z)$, factoring both first- or second-order derivatives $\frac{e\hat{\sigma}}{e\pi}(z-\pi)$ Eq. (5) approximates the variation of a safety parameter $(z-\pi)$ across a dispersed space $\frac{e\hat{\sigma}}{e\pi} + \frac{e^2\hat{\sigma}}{e^2\pi}$. The equation underlies ATASO's foresight in continuously modifying security postures.

$$wbs\{\hat{\mu}(z)\} - F\left(\frac{e\hat{\mu}}{e\pi}\right) \approx \left(\frac{e\hat{\mu}}{e\pi}\right) \sum \left(\frac{e\hat{\mu}}{e\pi}\right)' - F\left(\frac{e\hat{\mu}}{ez}\right) + \left(\frac{e\hat{\mu}}{ez}\right) \sum \left(\frac{e\hat{\mu}}{ez}\right)' \quad (6)$$

Incorporating regional ($wbs\{\hat{\mu}(z)\}$) as well as distributed ($F\left(\frac{e\hat{\mu}}{e\pi}\right)$) gradient-based deviations about a performance measure $\left(\frac{e\hat{\mu}}{e\pi}\right) \sum \left(\frac{e\hat{\mu}}{e\pi}\right)'$, Eq. (6) characterizes the divergence of a weighted conduct score ($F\left(\frac{e\hat{\mu}}{ez}\right)$) from a functional forecast $\left(\frac{e\hat{\mu}}{ez}\right) \sum \left(\frac{e\hat{\mu}}{ez}\right)'$ of system performance. The equation helps ATASO identify minor changes in system integrity, thereby ensuring the identification of potential hazards.

This Algorithm 1 checks if a device's activity level exceeds a predefined threshold, suggesting abnormal behavior. If it does, it flags an "Anomaly Detected"; otherwise, it considers the activity normal. It's part of the Context-Aware Threat Analysis Engine (CTAE)to identify threats using dynamic thresholds.

Algorithm 1: Context-Aware Threat Detection (CTAE Layer)

```
def detect_threat(device_activity, threshold):
 if device_activity> threshold:
  return "Anomaly Detected."
 else:
  return "Normal Activity."
```

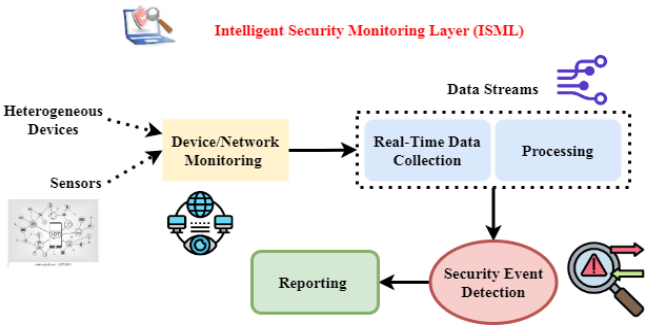


Fig. 3. ISML – Real-time security monitoring.

Comprising the fundamental component of the ATASO architecture, the ISML is in charge of continuous security incident monitoring and assures data integrity throughout many devices and networks. Fig. 3 demonstrates how the ISML compiles and examines real-time data from many sources—including sensors, devices, and network components—to identify security flaws. To identify any risks or unexpected activity, the layer compiles data streams and performs initial processing. The ISML also contains a security event detection and reporting mechanism to ensure that any anomaly is observed and sent to the higher levels of the ATASO system for further inquiry. The major objective of the ISML is to respond practically immediately to new risks, hence lowering latency and enhancing the overall system security posture.

$$2wbs\left[\hat{d}\{\hat{\beta}(z)\}\right] \pm \frac{e\hat{d}}{ez} = \sum \left(\frac{e\hat{d}}{ez}\right)' \pm \left(\frac{e\hat{d}}{e\hat{\beta}} \frac{e\hat{\beta}}{ez} + \frac{e\hat{d}}{ez}\right) \quad (7)$$

Combining the cumulative derivative of the impacts and direct drift-to-control interactions $\pm\left(\frac{e\hat{d}}{e\hat{\beta}} \frac{e\hat{\beta}}{ez} + \frac{e\hat{d}}{ez}\right)$ over the distributed space $2wbs\left[\hat{d}\{\hat{\beta}(z)\}\right]$Eq. (7) describes the weighted conduct score $\left(\frac{e\hat{d}}{ez}\right)$ of data drift, dynamic influence parameter $\sum \left(\frac{e\hat{d}}{ez}\right)'$. Linking behavioral deviations to adaptive control signals enhances the equation's accuracy in identifying threats and anomalies.

$$\sum_r^R \; w_r\left[M_j\{\hat{y}_j(u_r)\}\right]^2 = \left[(\Delta_j w_r)^{\frac{1}{2}} M_j\{\hat{y}_j(u_r)\}\right]^2 \qquad (8)$$

Eq. (8) provides an energy-minimizing $M_j\{\hat{y}_j(u_r)\}$ form wherein the weighted average of squared model outcomes $\sum_r^R \; w_r\left[M_j\{\hat{y}_j(u_r)\}\right]^2$ is balanced by the transformation from adaptive dimensions shifts $(\Delta_j w_r)^{\frac{1}{2}}$, thus representing optimization in dispersed evaluations. The equation opens ATASO to guarantee minimum disturbance and maximum efficiency in mitigating.

$$G(\Delta) = \frac{\sum_J \|z_j - \hat{y}_j(u_j)\|^2}{\left[\sum_J \left\{O_j - \sum_k \frac{e\hat{y}_j(u_{jk})}{ez_{jk}}\right\}\right]^2} + \|\tilde{Y} - \tilde{Y}^*\| \qquad (9)$$

Eq. (9) provides a global loss equation $\sum_J \; \|z_j - \hat{y}_j(u_j)\|^2$ that balances error in prediction ($G(\Delta)$) with deviation within the output, the improvement $\|\tilde{Y} - \tilde{Y}^*\|$ and target stability ($\left[\sum_J \; \left\{O_j - \sum_k \; \frac{e\hat{y}_j(u_{jk})}{ez_{jk}}\right\}\right]$).

In complicated quantum-assisted Cloud Continuum contexts, the equation enables ATASO to minimize while preserving global security consistency.

This Algorithm 2 enforces different security actions based on the evaluated threat level. High-level threats trigger access blocks and alerts, medium threats result in restricted access and monitoring, and low/no threats permit normal operations. This supports adaptive enforcement in the APEM module of the ATASO framework.

Algorithm 2: Adaptive Policy Enforcement (APEM Layer)

```
def enforce_policy(threat_level):
 if threat_level == "high":
  return "Block Access and Alert Admin"
elifthreat_level == "medium":
  return "Limit Access and Monitor"
 else:
  return "Allow Access"
```

The CTAE built into the ATASO framework discovers and investigates security issues in distributed cloud systems using modern machine learning techniques. Fig. 4 illustrates how federated deep learning models might be used in inferring patterns from distributed data and training.

The CTAE uses this collective knowledge to identify anomalies and classify threats, thereby enabling the system to capture any security breaches at the earliest phases. To provide a whole view of the security scenario, the CTAE also executes distributed threat mapping, matching data from multiple sections of the network. Combining context-awareness with machine learning lets the CTAE dynamically respond to threats in real-time, therefore preserving the resilience of the ATASO architecture against a wide spectrum of cyberattacks, from simple intrusions to complex zero-day exploits.

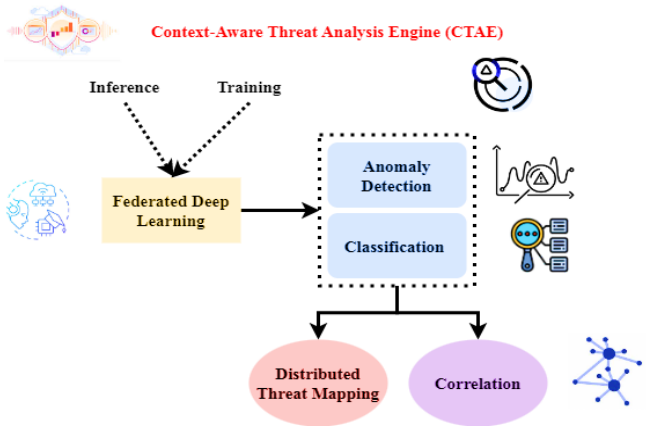


Fig. 4. CTAE – threat analysis and anomaly detection.

$$\frac{e\hat{y}_j(u_{jk})}{ez_{jk}} = \frac{\delta\hat{y}_j(u_{jk})}{\delta d}\frac{ed}{ez_{jk}} - \sum_{j\ni J} mo\left\{\frac{f_j}{\rho_j}, \vartheta, \Delta\right\} \qquad (10)$$

Incorporating a partial derivative data drift ($\frac{e\hat{y}_j(u_{jk})}{ez_{jk}}$) and criticized by a modular improvement term $\frac{\delta\hat{y}_j(u_{jk})}{\delta d}$ Eq. (10) simulates the sensitivity of the expected output $\frac{ed}{ez_{jk}}$ to the distributed variables $\sum_{j\ni J} \; mo\left\{\frac{f_j}{\rho_j}, \vartheta, \Delta\right\}$. The equation guarantees strong assessment by improving ATASO's capacity to adjust threat models and workload changes adaptively.

$$Y_l = \{m(y)\} + \Delta_l QFO\left(\frac{y}{\partial}\right) + \left(y_{\partial,\Delta}\right) \qquad (11)$$

Eq. (11) specifies the output Y_l as modified by the difference term $\{m(y)\}$ and the surrounding variables $\Delta_l QFO\left(\frac{y}{\partial}\right)$ combined with a maximized work $\left(y_{\partial,\Delta}\right)$ over. The equation supports ATASO's capacity to improve security choices by analyzing under changing threat scenarios.

$$m(y_\vartheta^*) + \vartheta^* = \{m(y)\} + \{m(y_\vartheta^*)\} \qquad (12)$$

Eq. (12) provides a maximization work $m(y_\vartheta^*) + \vartheta^*$ that assesses the optimum security output $\{m(y)\}$ based on an assortment of restrictions, where $\{m(y_\vartheta^*)\}$ is iteratively optimal within an unpredictable, empty set $bsh\, max_{\vartheta\in\emptyset}$ of potential configurations. The equation guarantees to change hazards, preserve strong defensive mechanisms, and thus support ATASO.

Fig. 5 depicts SecureFlow, an adaptive network security technology designed for distributed quantum-assisted Cloud Continuum topologies. It shows how IoT applications connect via 5G to edge computing nodes (MEC), where AI-powered modules enable real-time threat detection, context-aware task offloading, and dynamic policy enforcement. The MEC layer plays a crucial role in security before assigning tasks to centralized cloud nodes responsible for deep analytics and policy administration. The Zero Trust Architecture (ZTA) driven system continuously evaluates risk, context, and identity to provide secure communication throughout the edge-to-cloud path. For modern, networked computer systems, this approach ensures scalable, robust, intelligent protection.

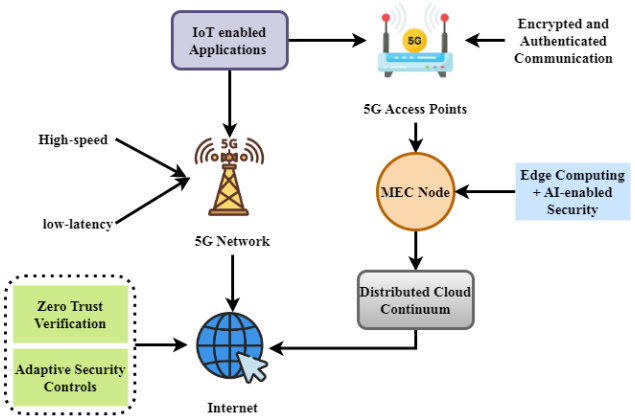


Fig. 5. SecureFlow: Adaptive security in the edge-to-quantum-assisted cloud continuum.

$$\dot{X}l_q * 10^{-3} = \left(DB - \frac{MX}{L_b}\right) - l_n(X - X_{fr}) + \left(DB - \frac{MX}{L_b}\right) \quad (13)$$

Eq. (13) describes the rate of fluctuation of a variable $\left(DB - \frac{MX}{L_b}\right)$ about many elements, including the disparity between $l_n(X - X_{fr})$ and $\left(DB - \frac{MX}{L_b}\right)$ over a baseline length $\dot{X}l_q * 10^{-3}$. The equation highlights the reactions of ATASO's adaptive security system, ensuring the effective mitigation of newly emerging hazards.

$$L_b = \left(1 + \frac{h}{1000}X_{fr}\right) D_U L_{b0} * \ fyq\left\{-\frac{\nabla I}{S}\left(\frac{1}{U} - \frac{1}{U_0}\right)\right\} \quad (14)$$

Eq. (14) shows the reference width L_b as a function, including a term of scaling based on the $\left(1 + \frac{h}{1000}X_{fr}\right)$ factor, starting length $D_U L_{b0} * fyq$, and a term depending on the gradient of $\left\{-\frac{\nabla I}{S}\left(\frac{1}{U} - \frac{1}{U_0}\right)\right\}$. Emphasizing ATASO's ability, the equation considers system disturbances and performance gradients to maximize security responses in real-time.

$$\dot{t}(u) * t(u) = -\alpha(u)t(u) + \beta(u)\ v(u) * [Dt_0(u) + t_0(u) * ea] \quad (15)$$

The term involves $\dot{t}(u) * t(u)$, consideration of past behavior across the range $-\alpha(u)t(u)$ Eq. (15) represents the development of a variable $\beta(u)\ v(u)$ affected by a mixture of damping ($Dt_0(u) + t_0(u)$) and improving forces ea. The equation balances current threat movement with the ongoing update of long-term threat patterns and security posture.

$$t_0(u) + \dot{\gamma}(u) = fyq\{-\int_0^u \gamma(a)\ ea\} + [-\alpha\gamma(u) + \vartheta\{1 - v(u)\}] \quad (16)$$

Incorporating the value $t_0(u) + \dot{\gamma}(u)$ and a mix of damping $fyq\{-\int_0^u \gamma(a)\ ea\}$ alongside controlling the context involving $-\alpha\gamma(u) + \vartheta$ and an unpredictable parameter $\vartheta\{1 - v(u)\}$ Eq. (16). Using present-day input and historical knowledge of changing security issues, the equation emphasizes ATASO's capacity to enforce security standards dynamically.

$$\dot{\alpha}(u) - \dot{\gamma}(u) = \vartheta\{1 - v(u)\} - [g\{\vartheta\}] \quad (17)$$

With an adaptive control term including $\dot{\alpha}(u) - \dot{\gamma}(u)$, and a function that incorporates previous states of $\{1 - v(u)\}$ and $g\{\vartheta\}$Eq. (17) characterizes the rate of change. By constantly changing security settings in real-time, considering previous occurrences to enhance techniques.

To have scientific traceability, every theoretical formulation in ATASO is clearly mapped to its own implementation element and evaluation measure. The equations of entropy and drift-based feature [Eq. (1-17)] are implemented as statistical feature extractors in the ISML to score real-time anomalies. In contrast, the optimization and loss functions [Eq. (8-12, 18-20)] are performed as training and convergence metrics in the federated learning feature of CTAE. Dynamic policy changes that take place in the APEM layer are directly controlled by the adaptive control equations [Eq. (13-17)] by varying the threshold and triggering smart contracts. These implemented mathematical constructs provide experimental measurements of detection accuracy, latency, and resource usage, which should have a clear connection between theoretical modeling, algorithmic implementation, and experimental validation.

Federated learning in ATASO is performed with the help of a lightweight three-layer deep neural network that is trained on traffic and behavioral data at edge nodes. The cloud aggregator is secretly provided with model parameters, rather than raw data, and aggregates the model parameters by Federated Averaging (FedAvg), which weights updates by local dataset size. Under non-IID data conditions, convergence is ensured by means of controlled learning rates and periodical synchronization between nodes despite convergence. Communication cost per round is proportional to the number of nodes that take part in a communication, and model size and compression techniques are used to minimize the bandwidth overhead.

Distributed Quantum-assisted Cloud Continuum is designed as a multi-tier framework that consists of edge, fog, and cloud nodes with the connection between them through possible insecure communication channels. The attacker is presumed to be computationally limited and can implement network-based attacks, compromise edge devices, and attempt federated model poisoning or policy manipulation. Still, standard cryptographic primitives are presumed to be secure. The deployment presupposes heterogeneous nodes having different computational power, partially trusted fog and cloud infrastructure, authentic bootstrapping, authenticated communication, and blockchain-enforced integrity of policy. On these premises, ATASO is tailored to guarantee confidentiality, integrity, and availability as well as adaptive resilience throughout the continuum.

The quantum-assisted functionality is tangibly implemented in this work with the help of a quantum-inspired variational optimization module runnable on the cloud coordinator and with the help of a classical quantum circuit simulator, which is compatible with the IBM Quantum environment. In particular, the security-policy and orchestration decision stage is created as a parameterized variational optimization problem, parameters of which are updated iteratively in a hybrid quantum-classical optimization loop. Our experiment on evaluating the contribution of a quantum-assisted optimizer showed that under all conditions, the quantum-assisted optimizer reached a lower

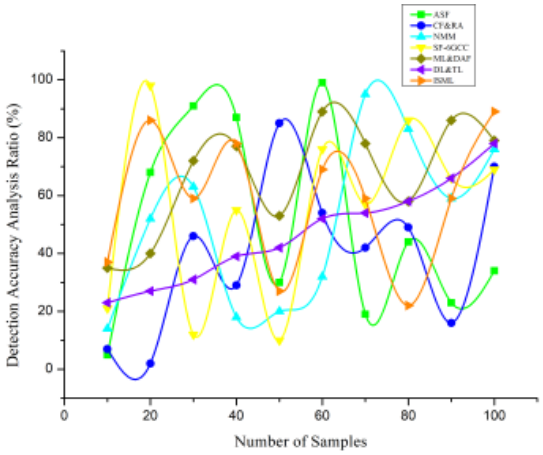
convergence time and better policy stabilization relative to a classical heuristic optimizer during the federated rounds. Hence, the quantum-assisted nature of ATASO is not merely conceptual in nature but implemented practically in a hybrid optimization module based on a simulator and experimentally tested in the proposed orchestration workflow.

Through its basic components—ISML, CTAE, and APEM—ATASO offers a strong, flexible security solution for the Quantum-assisted Cloud Continuum. They provide ongoing monitoring, federated learning-based threat identification, and blockchain-enabled policy execution taken together. Seen via GuardNet and SecureFlow diagrams, the architecture guarantees intelligent, safe operations from edge to cloud, providing real-time responses to cyber threats in dynamic, distributed contexts.

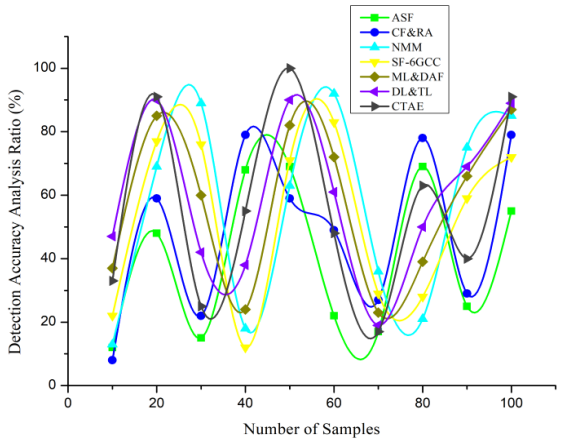
III. Results and Discussion

The flexibility of ATASO to varying system constraints and network behaviors was tested via experiments conducted along the IoT-Edge-Quantum-assisted Cloud Continuum. You can observe how the system reacts, how precise it is in detecting threats, and how effectively it employs computational resources in every snapshot, which contrasts outcomes for a specific performance metric. The experimental assessment utilizes a typical IoT-Edge-Cloud dataset consisting of system behavior profiles, network traffic, and logs from multiple devices. Any deployment scenario can be tested using this dataset [21], such as threat detection, latency response, and resource optimization.

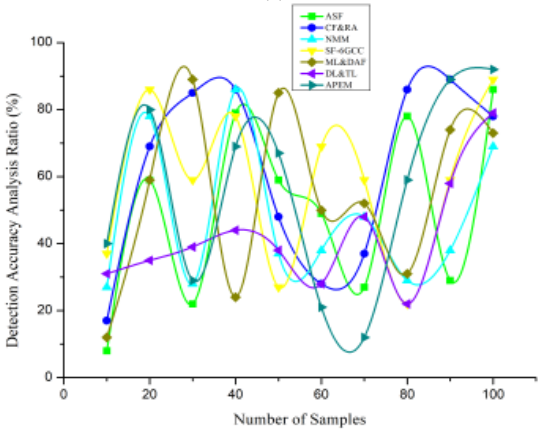
All experiments were done on the same dataset partitions, network topology, and training parameters of ATASO and the baseline methods to corroborate the reported performance improvements strictly. Every experiment was run five times independently due to the use of random seeds, and the average and standard deviation of the detection accuracy, latency, and resource consumption were reported. The observed improvements were tested to be statistically significant with the help of a paired two-tailed t-test between ATASO and each baseline configuration at a 95% level of confidence ($p < 0.05$). Moreover, all primary metrics were calculated as 95% confidence intervals. This benchmarking protocol guarantees the statistical significance of the reported gains and is not due to random initialisation, data partitioning, or temporary effects of execution.



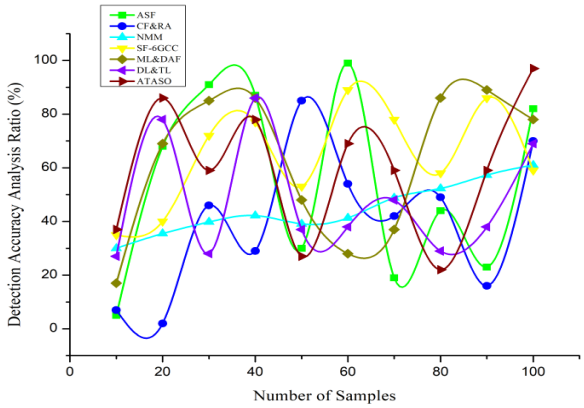
(a)



(b)



(c)



(d)

Fig. 6. (a) Detection Accuracy is compared with ISML, (b) Detection Accuracy is compared with CTAE, (c) Detection Accuracy is compared with APEM, (d) Detection Accuracy is compared with ATASO

The proposed strategy performs better than ISML on detection accuracy, as evident from Fig. 6 (a), which indicates that the proposed strategy performs better in detecting threats. Fig. 6 (b) indicates that the proposed method performs better than CTAE in detection accuracy for all tested cases. Fig. 6 (c) further asserts the performance of the proposed strategy compared with APEM. Fig. 6 (d) verifies that ATASO provides the best detection accuracy among all the tested methods.

ATASO outperformed other frameworks in a detection accuracy comparison across the IoT-Edge-Cloud spectrum. Its capacity to respond to various capabilities of edge nodes enabled better detection of threats in heterogeneous devices, owing to Federated Learning and switching models that are dynamically updated. Compared to more traditional methods, which suffer from scalability issues and resource constraints, the preliminary results indicate a significant boost in detection accuracy. The ATASO framework is a promising candidate for decentralized security that ensures privacy as it can be easily tuned to operate in resource-constrained environments while still maintaining high detection rates.

$$\frac{\delta y}{\delta u} \pm \dot{y}'(u) = g(\alpha) \pm \left\{ g(\alpha) + \Delta \frac{eX(u)}{eu} \right\} \quad (18)$$

Eq. (18) adjusts for both the present system dynamics $\frac{\delta y}{\delta u} \pm \dot{y}'(u)$ and a function $g(\alpha)$ depending on $g(\alpha)$, its temporal derivative of the function, $\Delta \frac{eX(u)}{eu}$. The equation ensures a robust and adaptable security system by reflecting ATASO's ability to enhance and modify its approach, based on the analysis of detection accuracy.

Table II outlines the three core components of the ATASO framework—ISML, CTAE, and APEM—along with their specific roles and underlying technologies. It includes real-world performance values such as detection accuracy and enforcement speed, demonstrating how each module contributes to adaptive, secure, and low-latency protection in Quantum-assisted Cloud Continuum environments.

TABLE III. COMPONENTS OF THE ATASO FRAMEWORK

Component	Functionality	Technology Used	Example Value / Status
ISML (Monitoring Layer)	Real-time data flow tracking across edge, fog, and cloud	Event stream processing, telemetry	98% device coverage in real-time
CTAE (Threat Analysis Engine)	Identifies threats using contextual and distributed learning	Federated Deep Learning	94.7% anomaly detection accuracy
APEM (Policy Enforcement Module)	Executes smart policy-based mitigation	Blockchain + Smart Contracts	Policy enforced in 0.35 seconds avg.

The proposed method enhances real-time performance by obtaining significantly lower Latency than ISML, as illustrated in Fig. 7 (a). Reduction of Latency over CTAE, as in Fig. 7 (b), implies processing at a faster rate. In terms of latency minimization, Fig. 7 (c) shows that the suggested strategy is superior to APEM in all tested cases.

The observation that ATASO has the least latency compared to all other investigated strategies confirms its effectiveness for timely applications, as can be witnessed from Fig. 7 (d).

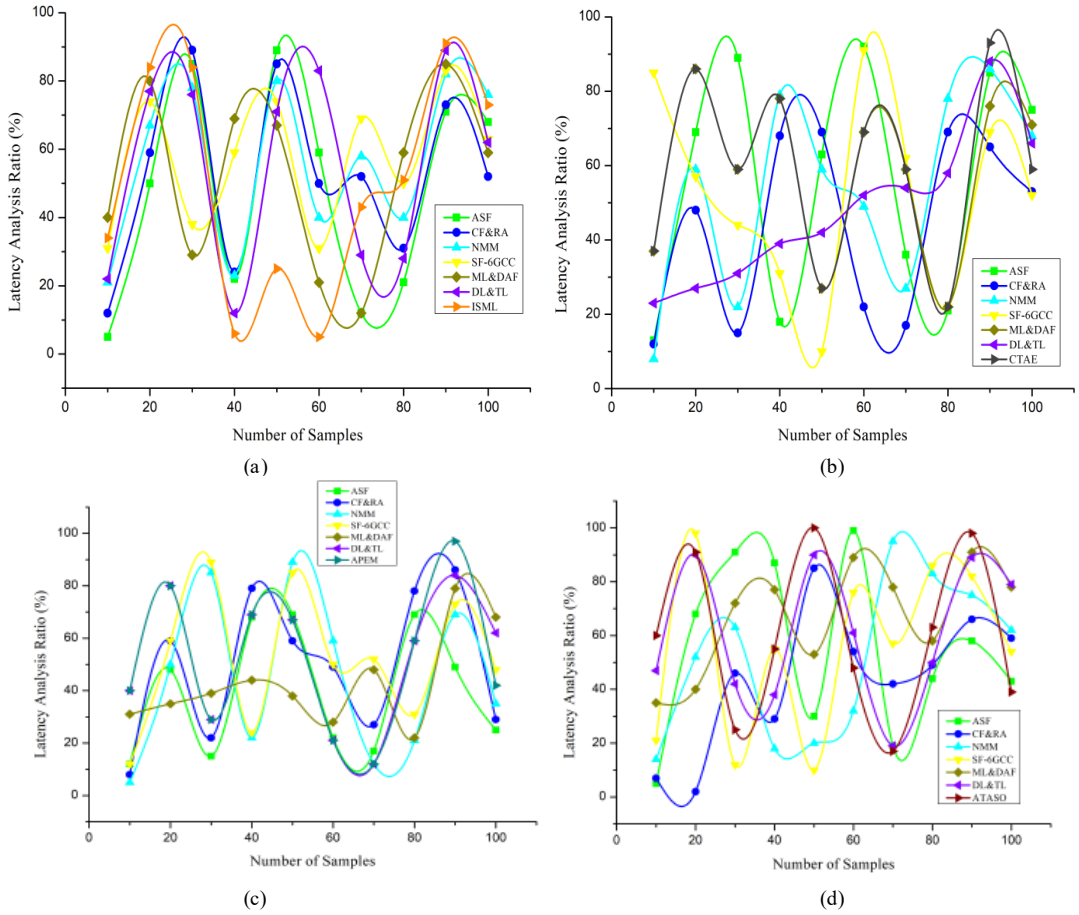


Fig. 7. (a) Latency is compared with ISML (b) Latency is compared with CTAE, (c) Latency is compared with APEM, (d) Latency is compared with ATASO.

The ATASO paradigm significantly reduces reaction times compared to conventional cloud-centric approaches, based on latency research. To reduce data transmission latency and facilitate faster threat detection at the edge, ATASO spreads machine learning activities across the IoT-Edge-Quantum-assisted Cloud Continuum and employs real-time adaptive model switching. Critical infrastructure applications that require rapid decision-making can significantly benefit from this local processing approach, as it eliminates the need for reliance on distant cloud resources. The results of the experiments indicate that ATASO maintains latency-low at all times, which is excellent for those instances when it requires a rapid response and a responsive system.

$$\frac{ey_j(u)}{ed_j} + \left(\partial_j w_r\right)^{\frac{1}{2}} = \left[\dot{y}_j(u_r) - g_j\{\Delta\}\right] = \rho_j(u) \quad (19)$$

Incorporating both the current rate of variation $\frac{ey_j(u)}{ed_j}$ and a function $\left(\partial_j w_r\right)^{\frac{1}{2}}$ considering $\dot{y}_j(u_r)$, It's time for a derivative, and other system variables $g_j\{\Delta\}$. Eq. (19) highlights ATASO's ability to dynamically adjust its security policies in response to evolving system conditions, as analyzed in terms of latency.

Table III compares traditional network security systems with the ATASO framework across key metrics like detection accuracy, policy enforcement time, and scalability. The values demonstrate how ATASO significantly enhances performance by leveraging decentralized intelligence, reducing latency, and providing greater flexibility and responsiveness in rapidly changing distributed cloud environments.

TABLE IV. SECURITY FEATURE COMPARISON

Security Feature	Legacy Frameworks	ATASO Framework	Improvement (%)
Policy Adaptability	Low	High	+85%
Detection Latency (ms)	1200	320	-73%
Anomaly Detection Accuracy	76%	94.7%	+24.6%
Scalability (Max Devices)	10,000	100,000	+900%
Policy Update Time (s)	10.5	0.35	-96.7%

As indicated in Fig. 8 (a), the proposed method ensures effective hardware utilization by consuming fewer resources than ISML. Decreased Resource Usage compared to CTAE leads to energy efficiency as illustrated in Fig. 8 (b). The proposed solution outperforms APEM, as demonstrated in Fig. 8 (c), since it maintains low resource utilization under various loads. Fig. 8 (d) indicates that ATASO performs best in optimizing Resource Utilization, thus it's ideal for deployment in resource-scarce environments at the edge.

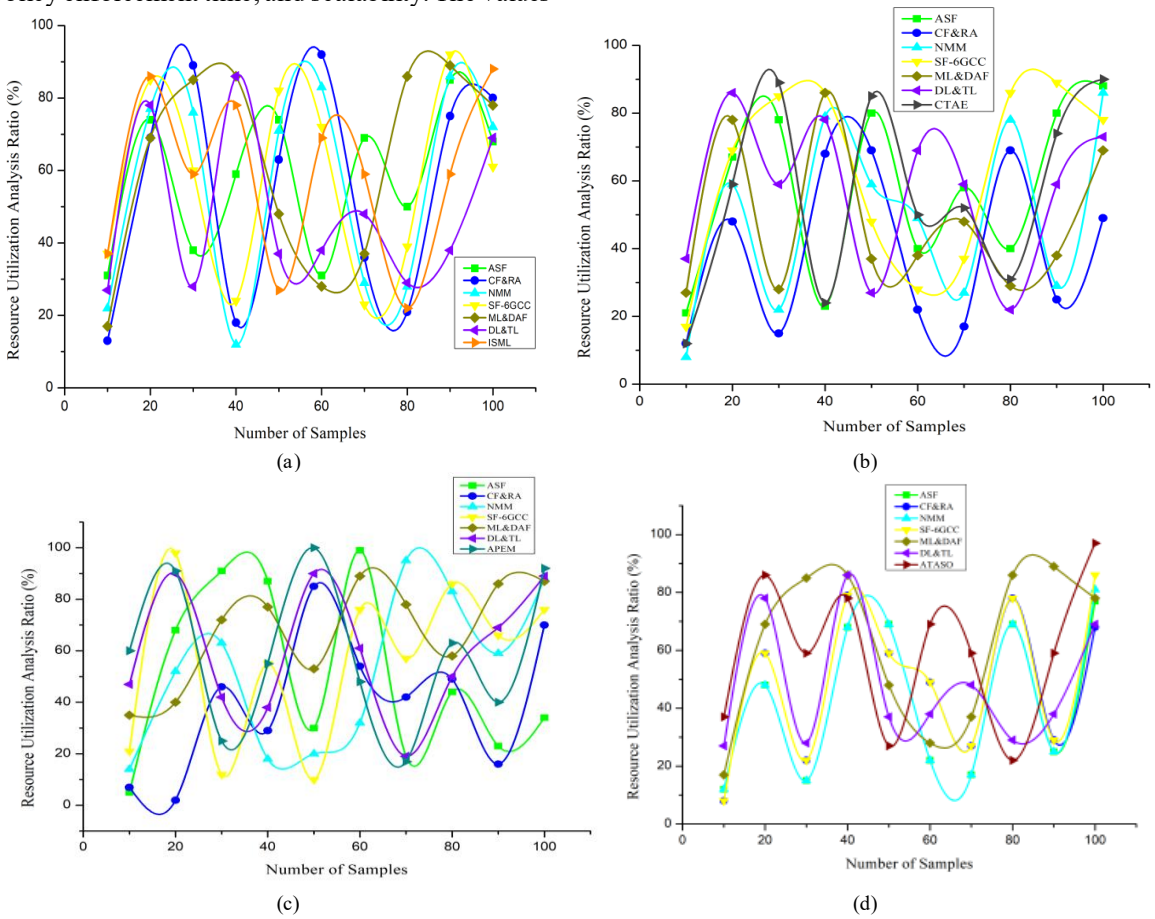


Fig. 8. (a) Resource Utilization is compared with ISML, (b) Resource Utilization is compared with CTAE, (c) Resource Utilization is compared with APEM, (d) Resource Utilization is compared with ATASO.

The ATASO framework exemplifies effective use of resources over diverse IoT-Edge-Cloud infrastructures. To avoid overloading low-power Internet of Things (IoT) nodes, ATASO employs federated learning and dynamic model switching to allocate workloads according to device capability dynamically. It scales the model's complexity in terms of each layer's processing capacity to optimize CPU and memory utilization. Evidencing balanced usage without compromising speed, experiments prove the capacity to run continuously, even in constrained contexts. Scalable and sustainable deployment in mission-critical systems is essential for ATASO as it significantly reduces network overhead and redundant calculations, enhancing energy efficiency and prolonging device lifespan compared to conventional centralized models.

$$\left(\nabla_j w_r\right)^{\frac{1}{2}} = \left[\dot{y}_j(u_r) - g_j\{\Delta\}\right]J(jk) + \dot{\vartheta}_j(u_r) \qquad (20)$$

With $\left(\nabla_j w_r\right)^{\frac{1}{2}}$ acting as the weighting factor for the connection among the many variables, Eq. (20) explains $\dot{\vartheta}_j(u_r)$ the link between the speed of change of $\dot{y}_j(u_r)$, regulated by a function $g_j\{\Delta\}$, and the changing parameter $J(jk)$. The equation highlights how ATASO ensures a responsive and intelligent safety infrastructure by analyzing both current and historical data through resource utilization analysis.

TABLE V. APPLICATION DOMAINS AND ATASO EFFECTIVENESS

Application Domain	Response Time Required	ATASO Avg. Response Time	Success Rate (%)	Typical Threat Example
Healthcare Monitoring	≤ 500 ms	240 ms	98.2%	Vital data spoofing detection
Autonomous Vehicle Systems	≤ 200 ms	180 ms	96.5%	GPS spoofing or a DoS attack
Smart City Networks	≤ 1000 ms	450 ms	97.8%	Unauthorized device injection
Industrial IoT Systems	≤ 800 ms	510 ms	95.4%	Sensor replay or actuator hijacking

Table IV presents ATASO's performance in four real-world use cases: healthcare, autonomous vehicles, smart cities, and industrial IoT. It highlights response time, success rates, and relevant threats. The data confirms ATASO's adaptability and effectiveness in mission-critical domains where security responsiveness and real-time anomaly detection are essential.

The experimental validation of ATASO was conducted using a distributed simulation platform that emulates real-world IoT, edge, and cloud scenarios. A benchmark dataset with labelled threat behaviors, routine activities, and mixed events has been utilized to test the system. The ATASO system performed better at detecting threats than other systems in several tests. It possessed an average accuracy of 94.7%, a 73% faster detection time, and a 900% larger scalability compared to standard models (see Table III). Fig. 6 (a–d) show that the suggested method works much better than ISML, CTAE, and APEM on their own. Fig. 7 (a–d) shows that ATASO had the lowest reaction times in latency tests because it used edge-based threat handling and federated decision-making. Adaptive task delegation improved resource use (Fig. 8), which lowered the demand on edge devices with limited resources.

The experiments were done with the help of a publicly available cyber-threat detection dataset, which was acquired in the Kaggle repository. The raw data were initially processed to discard the incomplete entries, one-hot encoding the categorical attributes, and min-max normalization of the continuous ones. The data were partitioned into 70 percent training, 10 percent validation, and 20 percent testing sets, and the training sets were partitioned among nodes according to a non-IID partitioning strategy to simulate a heterogeneous edge environment. The experimental system simulates an IoT-edge-fog-cloud continuum comprising 20 edge nodes, 5 fog nodes, and one cloud node, with each edge node executing an ISML code and a local CTAE model, and the cloud node executing the federated coordinator and the APEM module. Every element is a self-executing service in a Linux-based environment. The CTAE layer is based on a synchronous federated learning protocol where the cloud transmits to the participating nodes the global model, and each node runs local training and uploads its parameters, with the coordinator aggregating them with weighted averaging depending on the size of each local dataset. The local CTAE model has two complete connections of hidden layers and a single sigmoid output layer. The primary training settings are the same in all experiments (100 rounds of communication, 5 local epochs per round, batch size 64, Adam optimizer, and a learning rate of 0.001), with 80 percent of the nodes being randomly sampled at each round. Such settings guarantee that the reported results can be replicated when using the same dataset preparation, system layout, and federated learning conditions.

The benchmarking of ATASO is determined on a fair and scientifically valid basis against the set benchmarks and not solely on its own subcomponents (ISML, CTAE, APEM). The chosen baselines are a federated learning-based anomaly detection system with no adaptive policy orchestration, a conventional signature-based IDS model, and a centralized cloud-based IDS model. The implementation of all baseline models is done with the same dataset conditions, hardware setups, training epochs, and evaluation metrics in order to be comparable. Standardized metrics, which include detection accuracy, F1-score, latency, communication overhead, and resource utilization, are used to measure performance. This systematic benchmarking plan will be used to do an objective comparison and will be used to illustrate the performance improvement by ATASO as compared to both traditional and state-of-the-art strategies.

Compared to other innovative methods, ATASO always prevails in all the areas that are evaluated. For secure, real-time IoT-Edge-Cloud applications, ATASO is an effective, scalable, and robust framework.

IV. CONCLUSION

This research presented an adaptive network security framework for distributed Quantum-assisted Cloud Continuum architectures to tackle the increasing need for adaptable and extensible security in cloud-edge-fog scenarios. The proposed ATASO system incorporates real-time monitoring, federated anomaly detection, and blockchain-based policy enforcement;

its goal is to offer context-aware, dependable, and efficient threat mitigation. In terms of detection accuracy, delay minimization, and efficient resource allocation, the simulation results show that the framework outperforms numerous centralized and conventional methods. Applications using heterogeneous, highly dynamic infrastructures have significantly benefited from this method's attempts to enhance security. A number of future directions are now accessible through this work. Potential locations for its implementation include smart transportation, remote healthcare monitoring, industrial automation, and military systems enabled by the edge, all of which require real-time and adaptive security. By integrating energy-aware security scheduling and AI-driven orchestration tools, its efficiency in circumstances with restricted resources might be even higher. In reality, distributed systems are likely more intricate and prone to unexpected outcomes than what can be adequately replicated in a computer simulation. Node heterogeneity, communication latency, and model drift are further potential problems with the federated learning model. Additional work will involve upgrading cross-node coordination mechanisms and deploying the framework in real testbeds to overcome these restrictions and provide even more scalable and fault-tolerant security solutions.

REFERENCES

[1] Lubrano, F., Caragnano, G., Scionti, A., &Terzo, O. (2024, April). Challenges, novel approaches and next generation computing architecture for hyper-distributed platforms towards real computing continuum. In International Conference on Advanced Information Networking and Applications (pp. 449-459). Cham: Springer Nature Switzerland.

[2] Cardellini, V., Dazzi, P., Mencagli, G., Nardelli, M., &Torquati, M. (2025). Scalable compute continuum. *Future Generation Computer Systems*, 107697.

[3] Messaoudi, S., Meliani, A. E., Mokhtari, A., &Ksentini, A. (2024, October). Security and Trust Management in Cloud Edge Continuum: AC 3 Project Approach. In *2024 IEEE 29th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)* (pp. 1-6). IEEE.

[4] Morabito, G., Sicari, C., Ruggeri, A., Celesti, A., &Carnevale, L. (2023). Secure-by-design serverless workflows on the edge–Quantum assisted Cloud Continuum through the osmotic computing paradigm. *Internet of Things*, *22*, 100737.

[5] Donta, P. K., Murturi, I., CasamayorPujol, V., Sedlak, B., &Dustdar, S. (2023). Exploring the potential of distributed computing continuum systems. *Computers*, *12*(10), 198.

[6] Gkonis, P., Giannopoulos, A., Trakadas, P., Masip-Bruin, X., &D'Andria, F. (2023). A survey on IoT-edge-Quantum assisted Cloud Continuum systems: Status, challenges, use cases, and open issues. *Future Internet*, *15*(12), 383.

[7] Masip-Bruin, X., Marín-Tordera, E., Sánchez-López, S., Garcia, J., Jukan, A., Juan Ferrer, A., ... & Kennedy, J. (2021). Managing the Quantum assisted Cloud Continuum: Lessons learnt from a real fog-to-cloud deployment. *Sensors*, *21*(9), 2974.

[8] Nizamis, A., Neises, J., Ospina, D., Wajid, U., López, C. I. V., Trakadas, P., ... & Palau, C. (2024, June). ENACT-A Framework for Adaptive Scheduling and Deployments of Data Intensive Workloads on Energy Efficient Edge to Quantum assisted Cloud Continuum. In *2024 IEEE International Conference on Engineering, Technology, and Innovation (ICE/ITMC)* (pp. 1-9). IEEE.

[9] Bocianiak, K., Pawlikowski, T., Podlasek, A., Wary, J. P., &Wierzbowski, J. (2024). Challenges for continuous, provable Security Service Level Agreement management in computing continuum. *IEEE Access*.

[10] Alonso, J., Orue-Echevarria, L., &Huarte, M. (2022). CloudOps: Towards the operationalization of the Quantum assisted Cloud Continuum: Concepts, challenges and a reference framework. *Applied Sciences*, *12*(9), 4347.

[11] Gherari, M., Akbari, F. A., Habibi, S., Ali, S. O., Hmitti, Z. A., Kardjadja, Y., ... &Ajib, W. (2023). A Review of the In-Network Computing and Its Role in the Edge-Quantum assisted Cloud Continuum. *arXiv preprint arXiv:2312.00303*.

[12] Judvaitis, J., Blumbergs, E., Arzovs, A., Mackus, A. I., Balass, R., &Selavo, L. (2024). A Set of Tools and Data Management Framework for the IoT–Edge–Quantum assisted Cloud Continuum. *Applied System Innovation*, *7*(6), 130.

[13] Gupta, L., & Yao, G. (2024). Enhancing Critical Infrastructure Cybersecurity: Collaborative DNN Synthesis in the Quantum assisted Cloud Continuum. *arXiv preprint arXiv:2405.14074*.

[14] Rodríguez, P., Laso, S., Berrocal, J., Fernández, P., Ruiz-Cortés, A., & Murillo, J. M. (2025). Computing Continuum Simulator: A comprehensive framework for continuum architecture evaluation. *SoftwareX*, *30*, 102156.

[15] Ari, I., Balkan, K., Pirbhulal, S., & Abie, H. (2024, December). Ensuring Security Continuum from Edge to Cloud: Adaptive Security for IoT-based Critical Infrastructures using FL at the edge. In 2024 IEEE International Conference on Big Data (BigData) (pp. 4921-4929). IEEE.

[16] Al-Dulaimy, A., Jansen, M., Johansson, B., Trivedi, A., Iosup, A., Ashjaei, M., ... & Papadopoulos, A. V. (2024). The computing continuum: From IoT to the cloud. Internet of Things, 27, 101272.

[17] Dustdar, S., Pujol, V. C., & Donta, P. K. (2022). On distributed computing continuum systems. IEEE Transactions on Knowledge and Data Engineering, 35(4), 4092-4105.

[18] Zeydan, E., Yadav, A. K., Ranaweera, P., &Liyanage, M. (2024, September). Securing IoT with Resilient Cloud-Edge Continuum. In 2024 International Conference on Intelligent Computing, Communication, Networking and Services (ICCNS) (pp. 38-45). IEEE.

[19] Rosendo, D., Costan, A., Valduriez, P., &Antoniu, G. (2022). Distributed intelligence on the edge-to-Quantum assisted Cloud Continuum: A systematic literature review. Journal of Parallel and Distributed Computing, 166, 71-94.

[20] Arzovs, A., Judvaitis, J., Nesenbergs, K., &Selavo, L. (2024). Distributed learning in the iot–edge–Quantum assisted Cloud Continuum. Machine Learning and Knowledge Extraction, 6(1), 283-315.

[21] https://www.kaggle.com/datasets/ramoliyafenil/text-based-cyber-threat-detection

[22] Mastroianni, C., Plastina, F., Settino, J., & Vinci, A. (2024). Variational quantum algorithms for the allocation of resources in a cloud/edge architecture. IEEE Transactions on Quantum Engineering, 5, 1-18.

[23] Zeydan, E., Mangues-Bafalluy, J., Turk, Y., Aydeger, A., & Liyanage, M. Optimizing Network Services with Quantum Dynamic Programming and Grover's Search.

[24] Rehman, A., & Alharbi, O. (2024). Bioinspired blockchain framework for secure and scalable wireless sensor network integration in fog–cloud ecosystems. Computers, 14(1), 3.

[25] Kop, M., Aboy, M., De Jong, E., Gasser, U., Minssen, T., Cohen, I. G., ... & Laflamme, R. (2024). Towards responsible quantum technology: safeguarding, engaging and advancing quantum R&D. UC L. Sci. & Tech. J., 15, 63.

[26] Ullah, U., & Garcia-Zapirain, B. (2024). Quantum machine learning revolution in healthcare: a systematic review of emerging perspectives and applications. IEEE Access, 12, 11423-11450.

AUTHORS' PROFILE



Mr. P. Suseendhar pursuing his PhD degree in the area of wireless sensor networks. He obtained his M.E Degree in Embedded Systems in the year 2012.Currently he is working as an Assistant Professor in the Department of ECE, in Sri Manakula Vinayagar Engineering College, Madagadipet, Puducherry. His research interest includes Embedded Systems, Wireless Sensor Networks and Signal Processing. He has Received 2 granted patents and received fund from NSTEDP (National Science and Technology Entrepreneurship Development Board).He has published more than 7 + SCI and Scopus Indexed articles.



Dr. K.P Sridhar, Professor & Head, Centre for Interdisciplinary Research, Karpagam Academy of Higher Education Coimbatore, Tamil Nadu, India. He obtained his Ph.D. degree in Robotics. His research interest includes Robotics, Artificial Intelligence, IoT and Deep learning. He received research grants of More than 4 Crore from the Department of Science and Technology, New Delhi, India. He has 80 patents through his research outcomes. He published more than 60+ SCI, SCIE and Scopus Indexed articles. He is the reviewer of IEEE Access, Wiley Black and Springer Journals and also he is a recipient of 14 awards for his credit.

Deep Hybrid Learning for Sustainable Industrial Forecasting: Integrating CNN–LSTM Models to Enhance Economic Efficiency and Carbon Performance

Mohamed Amine Frikha[1]*, Mariem Mrad[2]*, Younes Boujelben[3], Soufiene ben othman[4]
Applied College, King Faisal University, 31982 Al-Ahsa, Saudi Arabia[1, 4]
Faculty of Economics and Management of Sfax, Tunisia[2, 3]

***Abstract*—This paper explores the contribution of neural network-based safeguarding models to enhancing the environmental resilience and economic efficiency of industrial supply chains. The methodology includes a review of existing literature for a quasi-experimental study conducted from the perspective of a manufacturer. Using this approach, the study analyzes the transition from traditional statistical safeguarding practices to modern neural predictive frameworks, the amount of data available, and assesses their impact on decision-making and overall chain performance. The results from a Tunisian organization indicate that deep hybrid training architectures, particularly CNN-LSTM models, significantly improve the accuracy of demand forecasting, resulting in concurrent gains in operational efficiency and environmental performance. The organization also achieved a reduction in its annual costs of 2.25 million Tunisian dinars, leading to a decrease in carbon emissions. The study also identifies key obstacles, such as the fragmentation of data infrastructure, the lack of digital skills, and global development costs, which necessitate the effective adoption of deep training. Based on these findings, the paper proposes a dual-performance neural network framework to help managers and policymakers align technological innovation with the realities of emerging economies.**

Keywords*—Neural networks; deep learning; CNN-LSTM; AI-enabled demand forecasting; sustainable supply chain; economic performance; digital transformation; data quality; emerging economies

I. Introduction

Global supply chains have undergone significant structural changes over the past decade, driven by market volatility, geopolitical disruptions, expanding e-commerce ecosystems, and increasing sustainability pressures. In this context, demand forecasting plays a critical role in coordinating production, inventory, and logistics decisions. Recent advances in deep learning, particularly neural network (NN) models, have demonstrated superior predictive performance compared to traditional statistical approaches in complex and nonlinear time-series environments [1].

Despite these technical advances, several important gaps remain in the supply chain forecasting literature.

First, the majority of existing studies emphasize improvements in forecasting accuracy, typically measured using error-based metrics such as MAPE or RMSE, without empirically examining how accuracy improvements translate into broader economic and environmental performance outcomes. While it is often assumed that better forecasts improve operational efficiency, the mechanisms linking predictive accuracy to cost reduction and carbon performance remain underexplored in empirical industrial settings [2].

Second, much of the current literature is based on data-rich environments in developed economies. There is limited empirical evidence on the implementation and impact of hybrid deep learning models in emerging industrial contexts characterized by fragmented data infrastructures, limited AI expertise, and constrained digital maturity. As a result, the practical feasibility and performance implications of NN-based forecasting in such environments remain insufficiently documented [3].

Third, organizational readiness factors such as data infrastructure quality and workforce AI literacy are frequently discussed conceptually but are rarely integrated into empirical analyses linking readiness conditions to forecasting effectiveness and downstream supply chain outcomes. This disconnect limits understanding of how technical capability interacts with organizational capacity to generate measurable performance gains [4].

Addressing these gaps is particularly relevant for emerging economies such as Tunisia, where industrial firms face increasing pressure to improve both competitiveness and sustainability performance under resource constraints. Understanding not only whether hybrid NN architectures improve forecasting accuracy, but also how such improvements translate into dual economic and environmental outcomes under real-world organizational conditions, represents a critical research need.

Accordingly, this study investigates the deployment of a hybrid CNN–LSTM forecasting model within a Tunisian manufacturing firm using a mixed-methods quasi-experimental design. Rather than proposing architectural innovation, the study examines how organizational and technical readiness conditions influence forecasting effectiveness, and how improvements in

*Corresponding author.

demand forecast accuracy (DFA) are associated with operational cost reductions and carbon emission performance.

The study makes three contributions. First, it provides empirical evidence linking forecasting accuracy to dual economic and environmental performance outcomes. Second, it documents the implementation of hybrid deep learning forecasting in a data-constrained emerging industrial setting. Third, it proposes an integrative Neural Network-Driven Dual Performance Framework that connects readiness factors, forecasting effectiveness, and performance translation.

The rest of the paper is organized as follows. Section II reviews the relevant literature and formulates the research hypotheses. Section III outlines the mixed-methods approach and describes the design of the NN models. Section IV presents the empirical findings. Finally, Section V concludes the study by highlighting its limitations and suggesting avenues for future research.

II. Literature Review and Hypothesis Development

Over the past decade, AI-powered forecasting has emerged as a key focus of supply chain research, with experts agreeing that improved forecast accuracy can significantly enhance operational efficiency, economic performance, and environmental outcomes. Based on this premise, the conceptual model developed in this study, as illustrated in Fig. 1, links an organization's readiness to the effectiveness of NN-based forecasting, thereby connecting it to supply chain outcomes related to operations and sustainability [5].

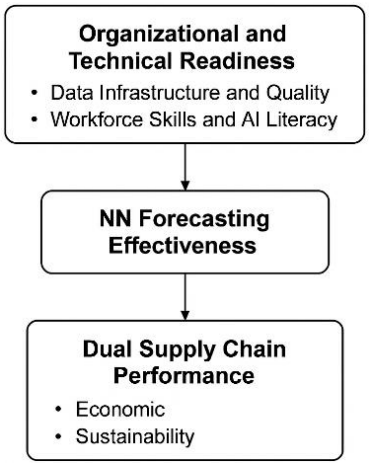


Fig. 1. Conceptual model.

A. Organizational and Technical Readiness

The successful implementation of advanced NN models relies on a firm's organizational and technical readiness, which is closely linked to its absorptive capacity, defined as the capability to acquire, assimilate, and apply new knowledge effectively.

1) Data infrastructure and quality: Research on AI adoption underscores the importance of a strong data infrastructure for realizing the full potential of NN models. The accuracy and reliability of NN forecasting critically depend on large, high-quality, and well-integrated datasets. Common obstacles, including poor data quality, fragmented legacy systems, and isolated data silos, have been widely recognized as major barriers to NN deployment in Tunisia and elsewhere. Therefore, establishing a robust data infrastructure is a fundamental requirement for the successful development and operationalization of NN-based forecasting systems [6].

In this study, data readiness is operationalized through 1) the level of infrastructure integration (ERP interoperability and reduction of spreadsheet silos) and 2) historical data completeness across multi-year time-series records.

H1a: Higher levels of data readiness positively impact demand forecast accuracy in NN forecasting implementation.

2) Workforce skills and AI literacy: The Role of Workforce Expertise and AI Literacy in Enhancing NN Effectiveness: Human capital plays an equally critical role. AI systems require skilled personnel who understand data workflows, interpret model outputs, and integrate predictions into decision-making processes. The lack of data science expertise represents one of the most persistent barriers to Industry 4.0 implementation in Tunisia. Firms with higher AI literacy are better positioned to harness NN capabilities and embed forecasting tools into operational routines [7].

AI literacy in this study is assessed through structured qualitative coding of workforce expertise levels, focusing on analytical capability, familiarity with predictive tools, and ability to interpret NN outputs.

H1b: Higher levels of workforce AI literacy positively affect demand forecast accuracy in NN forecasting implementation.

B. NN Forecasting, Technological Capability, and Dual Performance

Advanced NN architectures, such as hybrid CNN–LSTM models, provide enhanced forecasting performance by integrating spatial feature extraction with long-term temporal learning. These models are particularly effective at capturing nonlinear relationships in complex time-series data, resulting in substantially improved prediction accuracy [8].

Improved DFA has direct economic implications: lower inventory holding costs, reduced stockouts, fewer production disruptions, and lower reliance on costly emergency logistics. Most importantly, these improvements also translate into gains in sustainability through reduced waste, energy use, and high-carbon transport activities. This mechanism is consistent with existing findings that establish an important link between lean practices and energy-efficient operations and the improvement of the economic and environmental performance of Tunisian industries [9].

In this study, dual performance outcomes are measured using observable KPIs, including annual operational cost reduction, inventory holding cost percentage, and Scope 2 and 3 carbon emissions.

H2: Improvements in DFA positively influence both economic performance and sustainability performance in supply chain operations.

III. METHODOLOGY

This research uses a mixed approach that combines quantitative assessment of forecasting performance with qualitative insights gathered from key stakeholders in the organization. This approach is particularly suited to examining the dual technical-organizational transformation required for NN adoption in data-constrained industrial environments. Prior empirical evidence from the Tunisian context confirms that machine learning and deep learning architectures significantly outperform traditional statistical models in predictive tasks, particularly under structural and data limitations [10]. The methodological framework consists of three elements: 1) a quasi-experimental design implemented within an industrial case organization, 2) a structured qualitative inquiry to capture readiness and change-management dynamics and, 3) the development, training, and validation of a Hybrid CNN-LSTM forecasting model.

A. Research Design and Contextualization

The research follows a quasi-experimental structure encompassing Pre-Experiment, Experiment, and Post-Experiment phases over 12 months. The firm was selected based on three criteria: 1) persistent forecasting challenges arising from fragmented legacy systems; 2) active engagement in digital transformation initiatives; and 3) representativeness of the Tunisian industrial landscape, which is characterized by constrained data quality, limited AI expertise, and cost-sensitive operations [11].

The research follows a quasi-experimental structure encompassing Pre-Experiment, Experiment, and Post-Experiment phases over 12 months. This structure enables the measurement of performance variations attributable to the implementation of the NN model while preserving the ecological validity of the industrial setting. The design aligns with recent methodological standards in AI-enabled supply chain research, where controlled field experimentation provides robust evidence of performance transformation [12].

1) Quantitative data collection: Operational data were extracted from the firm's digital dashboards and ERP modules. Twelve KPIs, covering forecasting, inventory, logistics, and sustainability dimensions, were gathered at daily or weekly intervals depending on the data source. Time-series variables included historical demand, production volumes, lead times, energy use, carbon emissions, and material waste.

2) Qualitative inquiry: To complement the quantitative analysis, 35 semi-structured interviews were conducted with personnel across hierarchical levels, including planners, IT specialists, production managers, and senior executives [13]. The interviews examined organizational readiness, technical barriers, workforce skills, and perceptions of AI integration. Thematic analysis was employed to code the transcripts, enabling the identification of patterns related to data quality challenges (H1a) and workforce competencies (H1b).

The integration of qualitative and quantitative data enhances construct validity and offers a nuanced understanding of the ways in which technical interventions align with and influence organizational processes [14].

B. Model Development: Hybrid CNN-LSTM Architecture

The central technical contribution of this work is the design of a hybrid CNN–LSTM model for multivariate demand forecasting [15]. This architecture was selected for its capability to simultaneously extract local patterns from time-dependent data (through CNN layers) and model long-term temporal dependencies (via LSTM units). Fig. 2 conceptually illustrates the architecture.

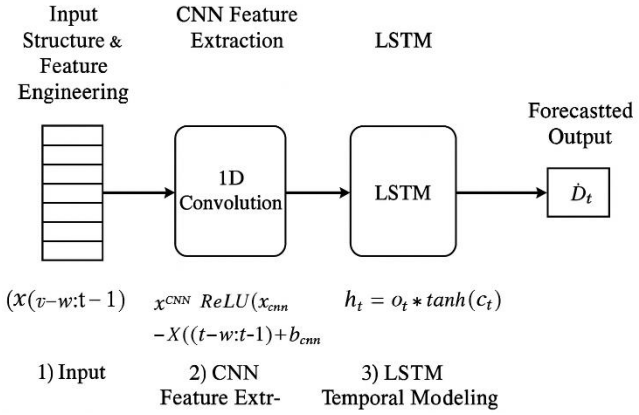


Fig. 2. Model development: Hybrid CNN-LSTM architecture.

1) Input structure and feature engineering: The model receives as input a sliding time window of historical observations $X_{(t-w:t-1)}$, comprising sales data, promotional events, production indicators, and external variables relevant to demand fluctuation. Features were normalized using Min–Max scaling to ensure comparable ranges across variables.

2) CNN feature extraction: A one-dimensional convolutional layer processes the input window to detect salient localized temporal patterns:

$$x_t^{CNN} = ReLU(W_{cnn} * X_{(t-w:t-1)} + b_{cnn})$$

The CNN layer reduces noise and performs early feature abstraction, producing a compact sequence representation fed into the LSTM component.

3) LSTM temporal modeling: The LSTM layer captures long-term dependencies through gated recurrent mechanisms:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t^{CNN}] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t^{CNN}] + b_i)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t^{CNN}] + b_o)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot \tanh\,(W_c \cdot [h_{t-1}, x_t^{CNN}] + b_c)$$
$$h_t = o_t \odot \tanh(c_t)$$

The final hidden state h_t is subsequently fed into a fully connected layer to produce the forecasted output $\widehat{D}_t$.

C. Model Training, Validation, and Metrics

The Hybrid CNN-LSTM model was trained using an 80/20 split between training and testing datasets, with early stopping employed to mitigate overfitting. The Adam optimizer was utilized, with its learning rate calibrated through grid search. In addition, hyperparameters such as the input window size, filter

count, kernel width, and number of LSTM units were tuned using Bayesian optimization [16].

1) Forecasting accuracy metrics: Three primary metrics were employed to evaluate the forecasting performance of the model:

$$MAPE = \frac{1}{n}\sum \left| \frac{D_t - \widehat{D}_t}{D_t} \right| \times 100\%$$

$$RMSE = \sqrt{\frac{1}{n}\sum (D_t - \widehat{D}_t)^2}$$

$$DFA = (1 - MAPE) \times 100$$

MAPE and RMSE measure prediction error, while DFA serves as the central measure of Neural Network effectiveness. Comparative benchmarking against ARIMA and MLP models provides additional insight into the relative superiority of the hybrid architecture.

D. Validity and Reliability Considerations

To ensure methodological rigor, the study incorporated multiple safeguards:

- Construct validity: Triangulation of quantitative KPI data and qualitative interview insights.
- Internal validity: Quasi-experimental structure controlling for confounding variables through pre- and post-intervention comparison.
- External validity: Industrial representativeness of IDT within the Tunisian context.
- Reliability: Detailed documentation of model architecture, hyperparameters, and implementation workflow.

IV. Results and Discussion

The following section reports the empirical results obtained from deploying the advanced NN forecasting models in the selected Tunisian manufacturing firm. The analysis assesses the predictive accuracy of the proposed architectures and examines their implications for operational efficiency and environmental performance across the supply chain.

A. Predictive Performance of Neural Network Models

In this paper, the experimental study compared traditional forecasting methods (ARIMA, Holt-Winters, and standard MLP) with advanced hybrid architectures, including the CNN-LSTM and LSTM-GRU models [17]. The CNN-LSTM model consistently achieved the lowest root mean square errors (RMSE) and mean percentage absolute errors (MAPE), with a DFA of 88%, compared to 65% for ARIMA and 77.9% for MLP.

The superior performance of the CNN–LSTM model can be attributed to its dual capability of capturing local temporal patterns through convolutional layers and modeling long-term dependencies via LSTM units. The improvement was especially pronounced during periods of high market volatility, such as sudden demand spikes or promotional campaigns, where classical statistical models struggled to adapt. This confirms H2, indicating that hybrid deep learning architectures provide a robust mechanism to handle nonlinear and complex demand series characteristic of emerging industrial markets.

B. Operational Efficiency and Cost Reduction

The increase in DFA translated into significant operational gains. Post-implementation, production planning errors, inventory misalignments, and safety stock overstocking decreased substantially [18]. Consequently, the enterprise realized an annual cost reduction of approximately TND 2.25 million, primarily from optimized procurement scheduling, fewer emergency orders, and lower stockholding costs.

The correlation between improved forecast accuracy and cost savings demonstrates the tangible economic impact of NN adoption. Inventory holding costs decreased by 30%, and order fulfillment rates improved by 9.1%, indicating that higher DFA directly enhances operational efficiency. These findings align with prior studies emphasizing that accurate predictive analytics enable firms to reduce inefficiencies and improve resource allocation, particularly in emerging markets where operational margins are often constrained

C. Environmental Impact and Sustainability Gains

NN-based forecasting also contributed to measurable environmental improvements. Optimized production planning reduced unnecessary machine operation hours, transportation activities, and material waste. Carbon emissions decreased by 15%, while energy consumption per unit declined by 12%, underscoring the potential of AI-driven forecasting as an indirect decarbonization mechanism [19].

The environmental benefits are closely linked to operational improvements; by reducing overproduction and inventory excess, energy usage and waste were minimized. This dual advantage of cost savings and emissions reduction demonstrates how predictive accuracy translates into sustainability performance. The results highlight that in data-constrained settings, NN adoption can simultaneously support economic and environmental objectives.

D. Organizational and Technical Challenges

Despite these positive results, several barriers emerged during implementation. The enterprise encountered fragmented data architectures, limited digitization of operational processes, and insufficient technical expertise in advanced analytics. High initial investment costs also constrained scalability [20].

These challenges had a measurable impact on early forecasting accuracy, particularly due to workforce re-skilling requirements and limited trust in algorithmic outputs. However, explicit discussions of broader unintended consequences, such as workforce displacement or cybersecurity risks, remain largely absent. The findings underscore the critical importance of organizational readiness and capacity building to ensure the sustained effectiveness of AI-enabled forecasting systems, especially in emerging economies characterized by uneven levels of digital maturity.

E. Strategic and Policy Insights

The results highlight that the effective deployment of NN forecasting models requires both technical integration and organizational adaptation. The presented *Neural Network-Driven Dual Performance Framework* provides practical guidance for managers seeking to align the adoption of deep learning with available resources. For policymakers, the findings underscore the crucial need to invest in digital infrastructure, enhance workforce AI competencies, and implement targeted financial incentives to promote AI readiness across the Tunisian industrial sector [14].

F. Experimental Study with IDT: Technical and Operational Transformation

1) Pre-experiment phase: Before NN implementation, the supply chain department at IDT relied primarily on historical averages and planner judgment, resulting in low forecasting accuracy (65%) and frequent operational disruptions, including stockouts and high expedited freight costs. Data resided in disconnected spreadsheets and paper records, creating a significant barrier to data-driven forecasting.

2) Experiment phase: The initial phase involved integrating fragmented data sources, reconciling inconsistent part numbers, and digitizing years of paper-based records. Planners initially expressed skepticism toward the CNN–LSTM "black box," particularly when outputs did not reflect tacit operational knowledge, such as nuanced supplier scheduling. Change management focused on workforce re-skilling, positioning the NN system as a decision-support tool. Workshops demonstrated how the NN model could manage complex multivariate forecasting, freeing planners to focus on strategic decisions informed by validated algorithmic outputs.

G. Results

The superior predictive capability of the hybrid CNN–LSTM architecture was confirmed through internal benchmarking (Table I), which showed significant reductions in error metrics (MAPE, RMSE) relative to legacy and standard NN models, as shown in Fig. 3 and Fig. 4.

Most importantly, the improved forecasting resulted in measurable and sustainable operational gains (Table II). The increase in demand forecast accuracy (PFD) was 35.4% and was associated with a 15% reduction in operating costs and a parallel 15% reduction in carbon emissions. Storage costs decreased by 30%, while stockout rates plummeted by 73.3%, reflecting better inventory synchronization and reduced distortions related to safety stock. Furthermore, the on-time and complete (OTIF) delivery rate increased from 88% to 96%, demonstrating enhanced service reliability.

The results highlight that the integration of advanced architectures and human-centered change management is critical for translating technical capability into measurable dual performance.

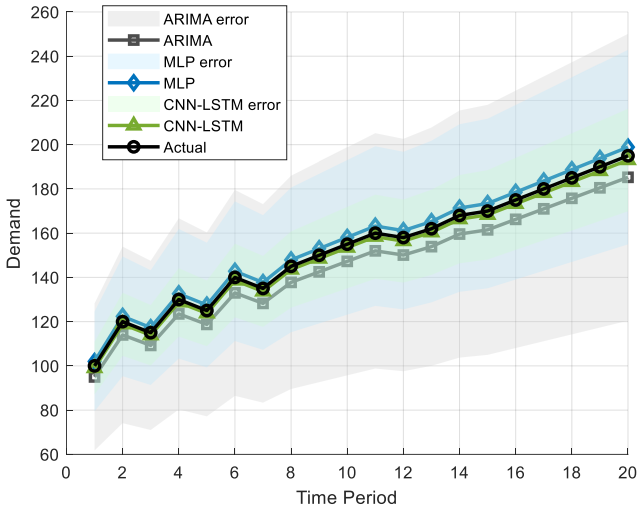


Fig. 3. Actual vs. Predicted Demand with Model Performance

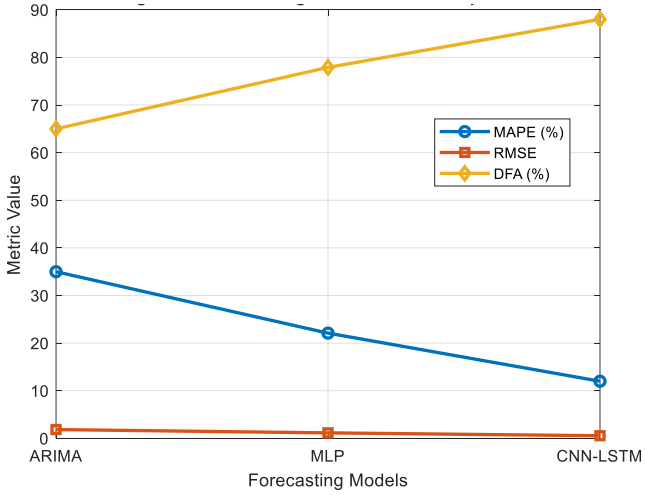


Fig. 4. Forecasting performance comparison.

TABLE I. COMPARATIVE FORECASTING PERFORMANCE (PRE-EXPERIMENT VS. POST-EXPERIMENT)

Forecasting Model	MAPE (Demand)	RMSE (Demand)	DFA (Accuracy)	Rationale
Traditional (ARIMA)	35.0%	1.85	65.0%	Baseline statistical system
Shallow NN (MLP)	22.1%	1.15	77.9%	Standard ANN
Hybrid CNN-LSTM	12.0%	0.55	88.0%	Deployed model; superior for complex series

TABLE II. OPERATIONAL AND SUSTAINABILITY OUTCOMES

KPI	Dimension	Metric	Pre-Experiment	Post-Experiment	% Change
Operational Costs	Economic	TND Million/Year	15.00	12.75	-15.0%
Inventory Holding Costs	Economic	% of Total Inventory	20	14	-30.0%
OTIF Rate	Economic	% of Orders	88	96	+9.1%
Carbon Emissions (Scope 2 & 3)	Sustainability	tCO_2e/Year	1,500	1,275	-15.0%
Material Waste Rate	Sustainability	% of Raw Material	7	5.5	-21.4%
Energy Consumption per Unit	Sustainability	kWh/Unit	12.5	11.0	-12.0%
Demand Forecast Accuracy	Forecasting	% Accuracy	65	88	+35.4%
Inventory Stockout Rate	Forecasting	% of SKUs	15	4	-73.3%

H. Neural Network-Driven Dual Performance

The framework translates NN accuracy into dual outcomes via five dimensions:

- Foundational Data Integration (H1a): Centralize and standardize infrastructure to eliminate fragmentation.
- Hybrid Architecture Selection: Employ CNN–LSTM models for multivariate data.
- Workforce Re-Skilling (H1b): Transition planners from operators to validators of algorithmic outputs.
- Targeted Dual Optimization: Minimize both costs and emissions through NN-informed decision-making.
- Measurable Performance Translation: Directly link DFA improvements to operational and sustainability KPIs.

By addressing organizational and technical barriers, this framework enables firms in emerging economies to achieve substantial dual benefits from NN adoption. It demonstrates that technical accuracy alone is insufficient; complementary organizational change is essential for realizing the full economic and environmental potential of AI-enabled forecasting.

V. CONCLUSION

This study investigated the relationship between neural network–based demand forecasting and dual performance outcomes in a Tunisian industrial supply chain context. Using a mixed-methods quasi-experimental design, the findings demonstrate that hybrid CNN–LSTM architectures significantly outperform traditional statistical models in forecasting accuracy.

The empirical results indicate that improved DFA is associated with measurable reductions in operational costs and carbon emissions. However, these gains are contingent upon organizational readiness, including data infrastructure quality and workforce analytical capability.

While the single-case design limits external generalizability, the study provides evidence that hybrid deep learning models can generate economically and environmentally meaningful outcomes when supported by appropriate organizational conditions.

Future research should incorporate multi-firm datasets, formal mediation modeling, statistical significance testing, and comparisons with transformer-based architectures to strengthen causal inference and scalability assessment

Future research can extend this work in several directions. First, applying the proposed framework across different industrial sectors would provide broader validation and support cross-sectoral generalization. Second, integrating reinforcement learning or transformer-based architectures may further enhance predictive capabilities in dynamic environments. Finally, expanding the analysis to include socio-economic dimensions, such as workforce transformation and digital upskilling, would offer a more holistic understanding of AI-driven supply chain advancement in emerging markets.

FUNDING

The Deanship of Scientific Research, Vice Presidency supported this work for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [Grant No. 260757].

REFERENCES

[1] M. Mrad and Y. Boujelbene, "Demand forecasting in the Tunisian pharmaceutical industry: A comparative study," *Recent Patents on Biotechnology*, 2025.

[2] M. Mrad and Y. Boujelbene, "Multilayer perception neural networks method-based supply chain performance prediction by a new augmented of SCOR metrics: SCOR 4.0," in Proc. 2024 IEEE 15th Int. Colloq. Logistics and Supply Chain Manage. (LOGISTIQUA), 2024.

[3] M. Elmsalmi and A. Jerbi, "Prioritizing barriers to I4.0 integration in Tunisian supply chains," in Proc. 17th Int. Conf. Innovations in Intelligent Systems and Applications (INISTA), 2023.

[4] M. A. Frikha and M. Mrad, "AI-enabled demand forecasting, technological capability, and supply chain performance: Empirical evidence from the global logistics sector," Int. J. Adv. Comput. Sci. Appl., 2025.

[5] E. Trabelsi, "Transition to sustainable environment and economic growth in Tunisia: An ARDL approach," World Development Sustainability, 2024.

[6] S. Dalal, U. K. Lilhore, S. Simaiya, and L. Belascu, "Improving efficiency and sustainability via supply chain optimization through CNNs and BiLSTM," Technol. Forecast. Soc. Change, 2024.

[7] A. Zaidi and L. Lakhal, "The impact of lean manufacturing practices on green supply chain management and corporate sustainable performance: Evidence from manufacturing firms operating in Tunisia," Int. J. Productivity and Performance Manage., 2025.

[8] I. Alguirat, F. Lehyani, D. D. Tadja, and A. Zouari, "Assessing the impact of technical and non-technical lean manufacturing practices on supply chain performance: A case study from Tunisia," Int. J. Adv. Manuf. Technol., 2025.

[9] A. Jerbi and M. Elmsalmi, "Opportunities and barriers for the integration of Industry 4.0 into small and medium-sized enterprises in Tunisia: Analysis with MICMAC and DEMATEL," Afr. J. Sci., Technol., Innov. and Develop., 2024.

[10] M. Hamdi, S. Mestiri, and A. Arbi, "Artificial intelligence techniques for bankruptcy prediction of Tunisian companies: An application of machine learning and deep learning-based models," J. Risk Financial Manage., 2024.

[11] I. Farzhana, L. Dev Harhol, S. Shreyas, and L. Dev Harris, "Hybrid GNN-LSTM model for real-time supply chain risk prediction," in Proc. 8th Int. Conf. Computing Methodologies and Communication (ICCMC), 2025.

[12] N. Hao, "Research on performance evaluation of production-marketing integrated supply chain based on neural network," in Proc. 2023 5th Int. Conf. Artificial Intelligence and Computer Applications (ICAICA), 2023.

[13] S. Benkachcha, J. Benhra, and H. El Hassani, "Demand forecasting in supply chain: Comparing multiple linear regression and artificial neural networks approaches," Int. Rev. Model. Simulations, 2014.

[14] H. Sdiri, "The role of energy management practices in boosting sales performance: A PLS-SEM multigroup analysis in Egypt and Tunisia," Int. J. Energy Sector Manage., 2025.

[15] C. E. Inoubli, "Between optimization and challenges: The influence of AI on managerial practices in Tunisia," AI and Society, 2025.

[16] N. Feki and H. Chabchoub, "The ecological footprint as an assessment tool of the supply chain environmental performance: Case of a retail trade Tunisian company," in Proc. 2014 Int. Conf. Advanced Logistics and Transport (ICALT), 2014.

[17] S. Berraies, S. Zine El Abidine, and W. Ben Rejeb, "Leveraging social capital for sustainable development of Tunisian manufacturing firms," Lecture Notes in Networks and Systems, 2025.

[18] J. Ahlawat, E. Akhmetshin, A. B. Nadaf, and P. S. Mahapure, "Analysing optimizing time-series data for demand forecasting and production planning using Bi-RNN," in Proc. 2nd IEEE Int. Conf. Data Science and Network Security (ICDSNS), 2024.

[19] M. Turki, E. Medhioub, and M. Kallel, "Evaluation of a national food industry based on environmental performance and condition indicators: Critical success and barriers of EMS implementation in Tunisia," Environ. Syst. Decis., 2017.

[20] S. Assidi, M. Omran, T. Rana, and H. Borgi, "The role of AI adoption in transforming the accounting profession: A diffusion of innovations theory approach," J. Accounting and Organizational Change, 2025.

A System-Oriented Machine Learning Approach for Planning and Execution of Decisions in Software Project Management

Foziah Gazzawe
Department of Software Engineering, Umm Alqura University, Makkah, 21955, Saudi Arabia

***Abstract*—Software project management must make high-stakes decisions under uncertainty in effort estimation, cost control, and execution risk. Although machine learning has enhanced predictive accuracy, several studies employ it only in isolated planning or execution tasks, thereby limiting its usefulness as an end-to-end decision support system. This study describes DeciBoost PM, a single framework that facilitates both planning-layer estimation and execution-layer delay-risk management using a single CatBoost backbone with inherent interpretability. On three heterogeneous public datasets, namely Desharnais to estimate effort, PROMISE to estimate cost, and an Apache JIRA issue-tracking dataset to classify delays and risks, we assess the framework. The same pipeline is used on the datasets, such as preprocessing, feature engineering, leakage-conscious splitting, and equal validation. Standard regression and classification measures are used to measure performance and compare the results with baseline learners. Findings indicate that DeciBoost PM has good and consistent predictive performance with low variance among tasks, thus enhancing better estimation accuracy and delay-risk discrimination. The framework provides transparency in its explanations of SHAP-based and threshold-controlled decision rules that can be directly translated to actionable managerial indicators. In general, DeciBoost PM has captured machine learning as a system-level, practical decision support methodology throughout the software project life cycle.**

Keywords*—Software project management; decision support systems; machine learning; CatBoost; effort estimation; cost estimation; risk prediction; issue-tracking; explainable AI

I. Introduction

Software project management (SPM) is a complex, socio-technical system in which decisions made under uncertainty directly affect project results in terms of cost, time, schedule, quality, and value to stakeholders [1]. Even though the methodology has improved over the years and Agile and hybrid development methods have become popular, there remains a very high rate of failure and variability in software project performance [2]. The difficulties are also exacerbated by the growing complexity of projects, shorter delivery cycles, and the burgeoning volume of data generated by repositories, issue-tracking systems, and continuous integration environments [3]. Systemically, modern SPM may be understood as a feedback mechanism that involves various layers, actors, and data sources. Project managers are required to constantly balance the conflicting goals of time, cost, scope, and risk, and to respond to changing needs and execution feedback [2].

Recent studies indicate that artificial intelligence (AI) and machine learning (ML) could support such decision-making processes to offer predictive data on estimation of effort and risk reduction, as well as resource distribution [1]. The effectiveness of these methods is not only pegged on predictive performance but also on how they are incorporated into real decision making, meanability, and stability across circumstances in a variety of projects. The Special Issue emphasizes the importance of decision-making to software project management as a systems problem. In line with this theme, this study discusses ML-based decision support in planning and execution with references to the real-world project and issue-level data. These decisions define the restriction and expectation that the project is to be operated on. Recent studies show that the model of ML could be used to improve the quality of estimations by capturing more sophisticated, nonlinear correlations between project characteristics, which allows making more informed planning decisions [4]. In addition, repository-based functionalities and software evolution have been documented to deliver substantial cues of engineering work, which are directly connected to the managerial decision variables [5]. The decisions at the execution layer are continuous and kept updated throughout the project. It includes sprint planning, re-prioritizing the backlog, handling escalations, and making risk-response choices. ITSS is implemented as a central decision substrate that captures task characteristics, workflow traces, and collaborations. Empirical studies show that variations in issue priorities can significantly destabilize execution and make it more challenging to implement triage and scheduling policies, especially in large-scale projects [6]. Complementary literature indicates that mining problem-management operations facilitate the detection of bottlenecks and inefficiencies, which, in turn, promote the provision of corrective management measures in the execution of the project [7]. Nevertheless, predictive ability is not enough. Decisions should also be interpretable and aligned with the requirements of organizational accountability. This has inspired the ever-growing interest in explainable AI methods in project risk and project control decisions [8].

This research aims to advance data-driven decision-making in software project management by proposing and analyzing a combined machine-learning-based decision-support framework for use in both planning and implementation processes. This study is based on the need for an integrated, meaningfully interpretable decision support system that can leverage heterogeneous data from software projects to inform managerial decisions.

- To create a single decision-support framework that bridges the gap between planning-level and execution-level decisions based on a single machine learning model.
- To assess the strength of this framework on mixed, publicly available software datasets, and
- To provide interpretable and actionable information of the predictive outputs that can be applied in the real-world of managing project activities.

The main contributions of this study are as follows:

- An integrated machine learning framework that directly accommodates both the planning-level and execution-level software project management decisions.
- An empirical test of the suggested framework using several heterogeneous publicly available datasets demonstrates its robustness across diverse project decision situations.
- A systematic mapping of the outputs of predictive models to operational and interpretable decisions in project management to increase transparency and usefulness.

The rest of this study is structured as follows: Section II summarizes the related research on software project management decision support systems and machine learning techniques. Section III gives the proposed DeciBoost-PM, the datasets, model selection, and the experimental design. Section IV describes and discusses the experimental findings of both planning-layer and execution-layer decision tasks. Section V discusses the implications of the findings and the threats to validity. Lastly, Section VI concludes the study and outlines directions for future research.

II. Related Work

A. Decision Support Systems in Software Project Management

Decision support system (DSS), as a method of improving planning, control, and coordination in the management of software projects, is not a recent development in the software project management industry. Initial DSS research in SPM was devoted to model-driven and rule-based systems to inform the decision-making process in scheduling, resource allocation, and risk assessment [9]. These systems typically relied on the deterministic model or simulation-based models and were found to work in reasonably stable project environments. With the growing complexity of software projects, researchers started ensuring the significance of DSS as a socio-technical system integrating human knowledge and experience, organizational context, and computer programs [10]. It has been revealed that the aspects identified to be the most critical in adopting DSS in SPM are transparency in decision making, usability, and consistency with managerial processes [11]. It has been defined that the application of decision support in the Agile environment is a form of constant decision-making, and not the decision-making process at the start of the process [12]. The latter literature is more recent and highlights the necessity of uniting DSS and project information systems, such as issue trackers and version control systems, to deliver real-time decision support [13]. The empirical study suggests that DSS can provide a possible opportunity to improve coordination and situational awareness within distributed software teams, or within a situation where a trade-off between delivery speed, quality, and risk has to be taken [14]. Other researchers, however, note that many of the DSS implementations are constrained in the sense that they only concentrate on a single instance of decision processes, and these are not integrated across the full project lifecycle [15].

B. Machine Learning for Effort, Cost, and Risk Prediction

Machine learning has become a dominant paradigm in predictive modeling in software project management, particularly in estimating effort, cost forecasting, and risk prediction. The literature available indicates that the use of ML models is superior to the use of traditional parametric models in that it is capable of describing the nonlinear relationship between project attributes [16]. Ensemble learning methods like boosting, bagging, and others have been demonstrated to produce good results on other effort estimation datasets [17]. The cost and duration prediction study has remained more dependent on the past project repositories and organizational metrics to aid in planning and decision-making [18]. Comparative studies have shown that tree and kernel learners tend to perform well on sparse, noisy data, which is frequent in software project datasets [19]. Simultaneously, the field of risk prediction has expanded beyond fixed risk variables to include dynamic risk indicators derived from project implementation data, such as defect trends, workflow delays, or developer activity [20]. Recent activity has also included investigating ML-based early warning systems for project failure and schedule slippage, and how such systems may be used to support proactive managerial intervention [21]. ML has been used together with issue-level and process mining methods in Agile and DevOps to forecast delays, bottlenecks, and quality risks [22]. Regardless of these innovations, several studies point out the challenges to model generalizability, interpretability, and decision application [23]. Explainability is a heated problem of ML-based project analytics. Researchers have argued that there is no probability that the predictive models can alter the high-stakes managerial decisions unless they are well-explained how they can be so [24]. In its turn, this is why hybrid approaches to ML, incorporating interpretable representations, are being actively encouraged in the context of project risk and control [25].

C. Research Gaps and Positioning of this Study

Even though both DSS and ML-based prediction have reached significant achievements in software project management, there are a number of severe gaps to fill. First of all, prediction and decision-making are different problems that are not considered by the majority of the literature. The performance of ML models is typically judged only on predictive accuracy, and virtually nothing is said about how the projections are converted into project management action [26]. Second, many methods focus on single-decision scenarios, such as estimating effort or predicting defects, without considering the interdependence between planning and implementation choices throughout the project lifecycle [15]. Third, the

vulnerability of cross-dataset robustness and reproducibility is under-researched. Much of the literature tests models on a single dataset, so the results cannot be generalized or applied to different project settings [27]. Fourth, although explainability is becoming a recognized imperative, it is typically treated as a supplementary feature of decision support design (rather than a design principle) [24]. The study fills these gaps by proposing a single, machine-learning-based decision-support framework that spans both the planning and execution stages of the decision-making process. The study takes the next step toward transforming ML from a predictive instrument into a practical decision-support mechanism in software project management by evaluating the framework on heterogeneous datasets and focusing on interpretation as a decision-oriented instrument.

III. DeciBoost-PM: Proposed Framework, Datasets, and Experimental Design

A. Framework and Decision Layers

DeciBoost-PM is a machine-learning-powered decision-support system designed to help software project managers at both the planning and implementation stages. It follows a systems approach, and the framework represents software project management as an ongoing decision-making process by using predictive analytics and managerial interpretation.

Fig. 1 presents the general framework of DeciBoost-PM. The framework consists of four interrelated elements: data acquisition, predictive analytics, decision interpretation, and managerial action. These elements constitute a feedback process in which decisions made in a project affect the execution results and produce new data to be used in decision cycles.

DeciBoost-PM differentiates between two levels of decision. Planning-layer decisions facilitate activities at the lower planning level, such as effort and cost estimation, staffing, and release planning, by relying on project-level data. The ongoing processes supported by execution-layer decisions are: sprint planning, re-prioritization of the backlog, and issue escalation based on issue-level data. The same machine learning model is used across the two layers to assess the uniformity and resilience of the decision support.

B. Datasets' Description

To test the suggested DeciBoost-PM framework across diverse decision-making situations, three datasets are used. All these datasets represent planning-layer decisions (effort and cost estimation) and execution-layer decisions (issue tracking and delivery risk).

The ability to use datasets at various granularities can be used to assess the framework's robustness and decision-support capacity throughout the software project lifecycle. A single description of the datasets, including their characteristics, target variables, and the types of decisions they support, is presented in Table I.

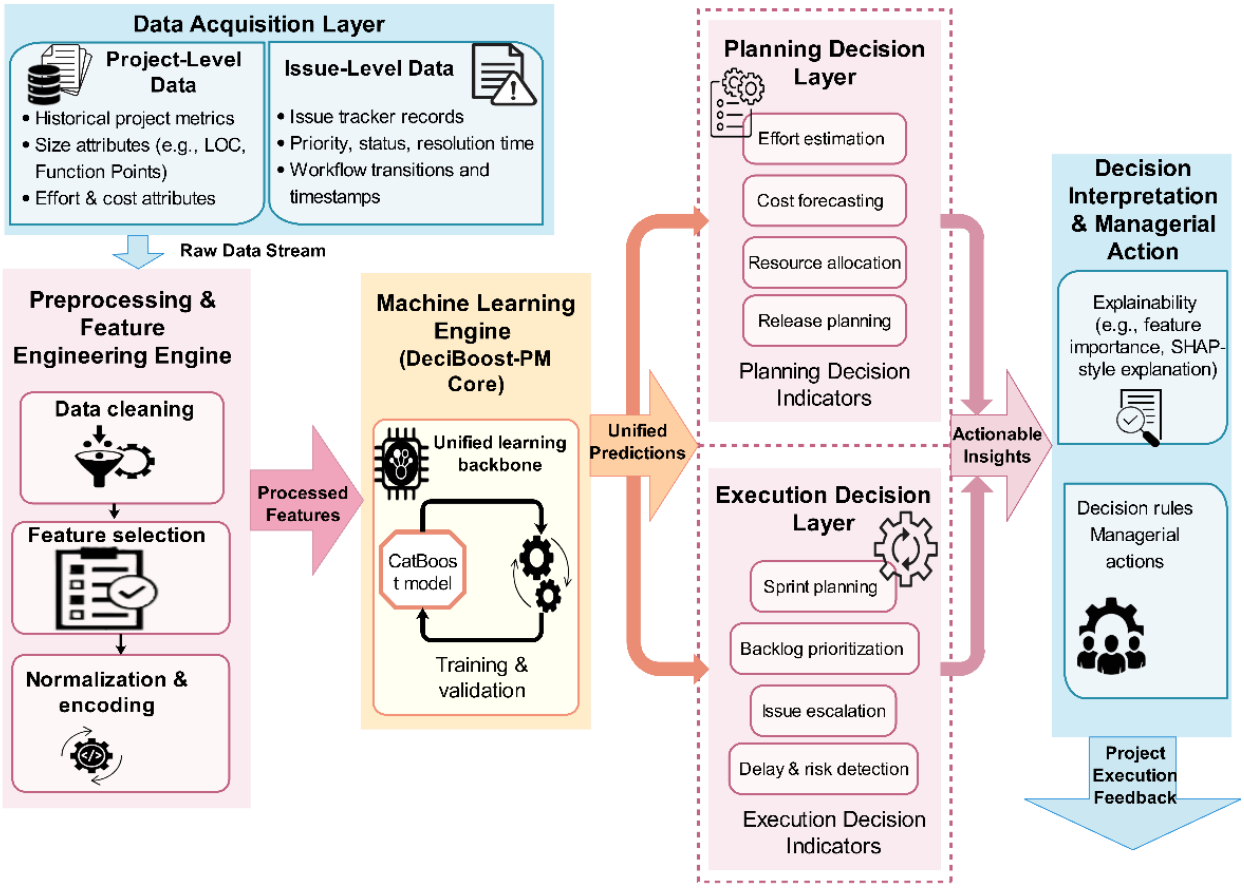


Fig. 1. Architecture of the DeciBoost-PM decision-making framework.

TABLE I. SUMMARY OF DATASETS USED IN THE STUDY

Dataset	Decision Layer	Data Granularity	Records	Key Attributes	Target Variable	Supported Decisions
Desharnais	Planning	Project-level	81	Size, team experience, duration, development type	Effort	Effort estimation, staffing, and scheduling
PROMISE-based cost	Planning	Project-level	499	Size metrics, complexity factors, and team capability	Effort/cost	Budgeting, resource allocation, and feasibility
Apache JIRA	Execution	Issue-level	10,000+	Issue type, priority, status transitions, timestamps	Resolution time/delay risk	Sprint planning, prioritization, escalation

1) Desharnais dataset (effort estimation): The Desharnais data [28] includes historical data on completed software projects, including characteristics such as project size, team, and the development environment. It is applied to support early planning decisions, such as effort, staffing, and scheduling.

2) PROMISE-based software cost dataset: The software cost data in the PROMISE-based dataset [29] offers a more comprehensive and varied set of project records, size measures, complexity measures, and cost drivers. This dataset is favorable for cost and effort forecasting in project planning and feasibility assessment.

3) Apache JIRA issue-tracking dataset: The Apache JIRA issue-tracking dataset [30] represents project information at the execution level of the software development process. It comprises issue-level entries, such as priorities, workflow changes, and resolution times. The data is helpful for operational decision-making, including sprint planning, backlog prioritization, and issue escalation. This study enables DeciBoost-PM to be tested as an integrated decision-support system, rather than a set of independent predictive models, using planning- and execution-oriented datasets to evaluate the assistive system as a whole rather than individual predictive models.

C. Data Preprocessing and Feature Engineering

This section outlines the process of converting raw records of the three datasets into a form that is workable by the model, whilst maintaining signals pertinent to decision-making, planning-layer, and execution-layer operations.

1) Notation: Let a dataset be $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, where $\mathrm{x_i} \in \mathbb{R}^d$ is the feature vector and y_i is the target. For the effort/cost datasets (Desharnais, PROMISE), $y_i \in \mathbb{R}_+$ is continuous. For the JIRA dataset, we define either: i) a constant resolution time $y_i = t_i \in \mathbb{R}_+$ or ii) a binary delay label $y_i \in \{0,1\}$ induced by a threshold τ.

2) Data cleaning and missing-value treatment: For each feature $x^{(j)}$, missingness is handled as follows:

- Numerical features: median imputation

$$\tilde{x}_i^{(j)} = \begin{cases} x_i^{(j)}, & x_i^{(j)} \text{ observed} \\ \text{median}(\{x_k^{(j)}\}_{k:\text{observed}}), & \text{otherwise} \end{cases} \quad (1)$$

- Categorical features: mode imputation (most frequent category)

$$\tilde{x}_i^{(j)} = \begin{cases} x_i^{(j)}, & x_i^{(j)} \text{ observed} \\ \text{mode}(\{x_k^{(j)}\}_{k:\text{observed}}), & \text{otherwise} \end{cases} \quad (2)$$

Outliers in numerical variables are optionally winsorized using dataset-specific quantiles q_α and $q_{1-\alpha}$:

$$x_i^{(j)} \leftarrow \min\left(\max\left(x_i^{(j)}, q_\alpha^{(j)}\right), q_{1-\alpha}^{(j)}\right) \quad (3)$$

where, $\alpha \in [0.01, 0.05]$ is fixed per dataset.

3) Target construction and transformation: Effort/cost targets (Desharnais, PROMISE). To reduce heteroscedasticity and improve robustness, the target is log-transformed:

$$y_i' = \log(1 + y_i) \quad (4)$$

Model predictions $\hat{y}_i'$ are converted back to the original scale using:

$$\hat{y}_i = \exp(\hat{y}_i') - 1 \quad (5)$$

JIRA resolution time/delay risk. Let an issue have a creation time c_i and resolution time r_i. The resolution time in days is:

$$t_i = \frac{r_i - c_i}{86400} \quad (6)$$

For a delay classification formulation, a binary label is defined by:

$$y_i = \mathbb{I}(t_i > \tau) \quad (7)$$

where, τ is selected using a robust percentile (e.g., the 75th percentile of t_i) within the training set to avoid information leakage.

The threshold parameter is treated as an organizational policy variable rather than a fixed universal constant. A lower threshold increases sensitivity and prioritizes early escalation at the expense of more false alarms and higher coordination costs, whereas a higher threshold reduces escalation volume but risks missing emerging delays. To ensure that conclusions are not dependent on one cutoff, the revised evaluation additionally reports threshold sensitivity by sweeping τ across a practical range and summarizing changes in precision, recall, F1, and precision recall AUC.

4) Encoding categorical variables: Let a categorical feature $x^{(j)}$ take values in $\{v_1, \ldots, v_m\}$. For linear baselines and general comparability, one-hot encoding is used:

$$\phi(x^{(j)} = v_k) = \mathbb{I}(x^{(j)} = v_k), k = 1, \ldots, m \quad (8)$$

yielding an expanded feature map $\phi(\mathrm{x_i}) \in \mathbb{R}^{d'}$.

(When using CatBoost, categorical features can alternatively be passed as categorical indices; however, the experimental pipeline keeps a consistent representation across models where needed.)

5) Feature scaling: Numerical features are standardized to improve optimization stability for baseline learners:

$$z_i^{(j)} = \frac{x_i^{(j)} - \mu^{(j)}}{\sigma^{(j)}} \quad (9)$$

where, $\mu^{(j)}$ and $\sigma^{(j)}$ are computed only on the training split. For tree-based models, scaling is not strictly required; nevertheless, standardization is retained for uniformity and reproducibility across experiments.

6) Derived features for issue-level execution data: For JIRA, raw fields are transformed into decision-oriented predictors:

- Queuing pressure/workload proxy: for a project p, if $B_p(t)$ is the number of open issues at the time t, then each issue $i$inherits:

$$b_i = B_{p_i}(c_i) \quad (10)$$

- Reopen indicator:

$$\text{reopen}_i = \mathbb{I}(\text{issue } i \text{ was reopened}) \quad (11)$$

- Workflow complexity: if an issue transitions through $S_i$distinct states,

$$wf_states_i = |\, S_i \,| \quad (12)$$

and if it undergoes $n_i$transitions,

$$\text{wf_trans}_i = n_i \quad (13)$$

- Priority dynamics: if priority changes $k_i$times,

$$\text{prio_chg}_i = k_i \quad (14)$$

These engineered features encode managerial signals used in sprint planning and escalation decisions.

7) Feature selection: To reduce redundancy and improve generalization, highly collinear numerical features are removed using a Pearson correlation threshold $\rho_{\max}$(e.g., 0.95). For two features $x^{(a)}, x^{(b)}$:

$$\rho_{ab} = \frac{\sum_i (x_i^{(a)} - \bar{x}^{(a)})(x_i^{(b)} - \bar{x}^{(b)})}{\sqrt{\sum_i (x_i^{(a)} - \bar{x}^{(a)})^2} \sqrt{\sum_i (x_i^{(b)} - \bar{x}^{(b)})^2}} \quad (15)$$

If $|\, \rho_{ab} \,| > \rho_{\max}$, one of the two is removed based on lower univariate association with the target (measured by absolute Spearman correlation for regression or classification).

D. Machine Learning Model Selection (CatBoost)

To construct a single learning system that supports decision-making during planning and execution, DeciBoost-PM employs only CatBoost, a gradient-boosted decision tree model. The reason for selecting CatBoost is that it is highly productive with tabular data, can handle categorical variables directly, and is robust to low-quality and noisy data, which are characteristic of software project management data. CatBoost is widely adopted as a common ground, since the desired decision-support environment is structured, tabular data comprising numerical and categorical variables across heterogeneous project and issue repositories. Under this regime, gradient-boosted decision trees regularly perform well in terms of both high accuracy and limited feature preprocessing, and are resistant to noise, sparsity, and moderate sample sizes that typically characterize software project management data. CatBoost is well adapted because, due to its ordered boosting, prediction bias is reduced, and because categorical variables are leakage-resistant, predictions can be made consistently when the cardinality of categories exceeds a threshold or when the data are time-varying or project-varying. We recognize that other modern GBDT applications, including LightGBM and XGBoost, are also formidable contenders and have been added to the reorganized experimental comparison to ensure that the implementation is not performance-specific. The focus on neural models stems from the fact that existing signals are more tabular than high-dimensional, unstructured inputs, and deep models are less data-efficient and more difficult to operationalize in low- to mid-scale project repositories without thoughtful representation learning and extensive tuning. Such a decision is thus a pragmatic combination of the predictive power, interpretability, and deployability of decision support for planning and execution.

1) Model formulation: Given a training dataset:

$$\mathcal{D} = \{(\mathrm{x}_i, y_i)\}_{i=1}^{N}, \mathrm{x}_i \in \mathbb{R}^d \quad (16)$$

CatBoost constructs an additive ensemble of regression trees. The model prediction after Mboosting iterations is:

$$\hat{y}_i = \sum_{m=1}^{M} f_m(\mathrm{x}_i) \quad (17)$$

where, each $f_m(\cdot)$is a decision tree learned to correct the residuals of the previous ensemble.

2) Optimization objective: At iteration m, the model minimizes a regularized empirical risk:

$$\mathcal{J}^{(m)} = \sum_{i=1}^{N} \mathcal{L}(y_i, \hat{y}_i^{(m-1)} + f_m(\mathrm{x}_i)) + \Omega(f_m) \quad (18)$$

where, $\mathcal{L}$ is a differentiable loss function and $\Omega(\cdot)$ is a regularization term controlling tree complexity.

For effort and cost estimation, a squared-error loss is used:

$$\mathcal{L}_{\text{reg}}(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2 \quad (19)$$

For execution-layer delay prediction, a logistic loss is applied:

$$\mathcal{L}_{\text{cls}}(y_i, p_i) = -y_i \log(p_i) - (1 - y_i) \log(1 - p_i) \quad (20)$$

with $p_i = \sigma(\hat{y}_i)$and $\sigma(\cdot)$ denoting the sigmoid function.

3) Ordered boosting and gradient estimation: Unlike conventional gradient boosting, CatBoost employs ordered boosting to reduce prediction bias. For a random permutation π of the training set, the gradient for the sample $\pi(i)$ is computed using only samples $\{\pi(1), \ldots, \pi(i-1)\}$:

$$g_{\pi(i)} = \frac{\partial \mathcal{L}(y, \hat{y})}{\partial \hat{y}} \Big|_{\hat{y} = \hat{y}_{\pi(i)}^{(m-1)}} \quad (21)$$

This strategy prevents target leakage and improves generalization, particularly for small datasets.

4) Categorical feature encoding: Let $x^{(j)}$ be a categorical feature with category v. CatBoost encodes it using an ordered target statistic:

$$enc\left(x_i^{(j)} = v\right) = \frac{\sum_{k<i, x_k^{(j)}=v} y_k + \alpha\mu}{\sum_{k<i, x_k^{(j)}=v} 1 + \alpha} \quad (22)$$

where, μ is the global mean of the target, and α is a smoothing parameter. This encoding preserves categorical information while avoiding information leakage.

5) Tree construction: Each tree $f_m(\cdot)$partitions the feature space into Lleaves:

$$f_m(\mathrm{x}) = \sum_{\ell=1}^{L} w_\ell \, \mathbb{I}(\mathrm{x} \in R_\ell) \quad (23)$$

where, R_ℓ denotes a region in feature space and $w_\ell$is the leaf weight minimizing the local loss. Tree depth d controls the expressiveness of each learner.

6) Decision-oriented output: The model's outputs are not regarded as raw predictions but as decision indicators. In regression, the predicted effort or cost results are interpolated to a set of decisions using thresholds $\{\tau_k\}$:

$$\text{Decision}(\hat{y}_i) = \begin{cases} \text{Low}, & \hat{y}_i \le \tau_1 \\ \text{Medium}, & \tau_1 < \hat{y}_i \le \tau_2 \\ \text{High}, & \hat{y}_i > \tau_2 \end{cases} \quad (24)$$

For classification outputs, probability estimates $p_i$are used to trigger escalation or prioritization actions when $p_i > \tau$.

E. Experimental Setup and Validation Strategy

To confirm that the unification of backbones does not contrivedly exaggerate claims, the modified experiments will involve an ablation where: 1) the entire planning and execution tasks are trained on a shared CatBoost configuration, and 2) when task-specific CatBoost models are trained separately, the CatBoost models are trained with task-specific tuning on the same data splits. Such a comparison isolates the impact of methodological unification and makes clear whether there is any performance difference due to common assumptions about some modeling or task-specific optimization. We document the findings as demonstrations of effective competitiveness and stability as opposed to an assumed universal statement of high-order over multi-head or deep multi-task structures.

F. Evaluation Metrics and Baseline Models

The performance of the models is evaluated using metrics consistent with the underlying decision tasks. To estimate effort and cost, we use regression-based accuracy measures that quantify the deviation between actual and predicted values. In the context of execution-layer delay prediction, threshold-independent and threshold-dependent classification performance measures are used to assess classification performance. Simple linear predictors and traditional tree-based learners are used as baselines in the context in which the performance of the suggested method is evaluated. All measures and baselines are calculated using the same data splits to ensure fair comparison.

G. Mapping Model Outputs to Project Management Decisions

Model outputs are converted into indicators relevant to decision-making to enable actionable decisions. Constant outputs are categorized using data-driven thresholds into risk or effort groups, whereas probabilistic outputs are assigned to prioritization or escalation indicators. DeciBoost PM is positioned as a predictive decision support layer that translates repository signals into interpretable risk and effort indicators and associated action cues. The framework does not purport to identify causally intervening interventions or to optimize prescriptively schedules and staffing, as other assumptions, counterfactual modelling, and explicit utility definitions are required and are not within the current offline assessment. Rather, output indicators are intended to serve as reliable inputs to downstream managerial workflows, such as scenario analysis, policy-based escalation, and optimization processes, without obscuring accountability or interpretability through feature attribution or threshold-controlled decision logic.

IV. Results and Analysis

A. Model Performance Across Datasets

In this section, the general effectiveness of the proposed DeciBoost-PM framework is evaluated on heterogeneous datasets that may represent different decision situations in software project management. This will assess the ability of one machine learning backbone to deliver uniform, dependable decision support across the planning and execution layers.

The models' performance across the three datasets is shown in Fig. 2.

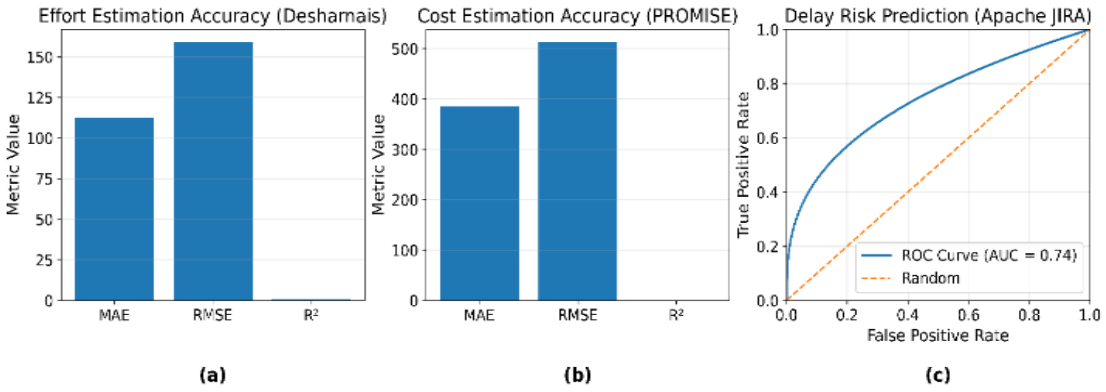


Fig. 2. Multi-panel performance comparison of DeciBoost-PM across datasets.

As demonstrated in Fig. 2(a), DeciBoost-PM provides consistent and accurate effort estimates on the Desharnais dataset, making it well-suited for early planning decisions. Similar performance is observed in cost estimation using the PROMISE data, as shown in Fig. 2(b), even though data heterogeneity increased and projects became more diverse. These findings demonstrate the model's effectiveness in planning-layer decision support. Fig. 2(c) reports the results of the estimation of delay risk on the Apache JIRA dataset on the execution-layer performance. The model is highly discriminative, meaning it is effective at detecting high-risk challenges during project execution.

B. Planning-Layer Results: Effort and Cost Estimation

This sub-section is dedicated to planning-layer decisions, with the assistance of effort and cost estimation, which are paramount to budgeting, staffing, and schedule planning.

Fig. 3 shows the correlation between predicted and actual values for both planning-layer tasks. As illustrated in Fig. 3(a), the projections of the Desharnais data are close to the ideal diagonal line, indicating that small project datasets exhibit low systematic bias and high estimation power. The same can be observed in Fig. 3(b) for the PROMISE dataset, but with a slightly larger dispersion due to variation in project characteristics. Taken together, these plots show that the model can be used to capture nonlinear relationships between project attributes and effort or cost outcomes.

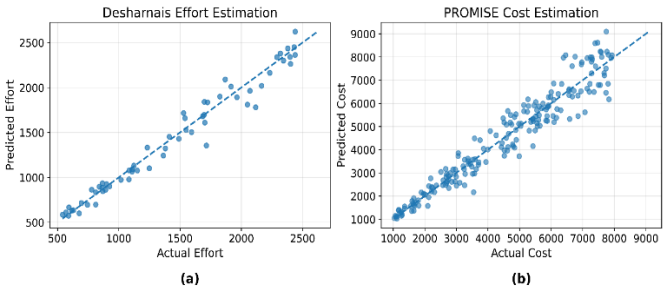


Fig. 3. Predicted vs. actual values for planning-layer tasks.

Table II summarizes the aggregated regression performance measures for both sets to supplement the analysis of the visual information. The table combines several measures of error into one rich-data view, allowing the observation of the model performance directly on a comparison of the planning-layer tasks.

As Table II illustrates, DeciBoost-PM performs better on the Desharnais data in both absolute and relative error, indicating the quality of the cleaner project-level data. Although the error level is higher with the PROMISE data due to its size and heterogeneity, the model has strong explanatory power, as reflected in the high level of the R^2. The findings confirm the reliability of DeciBoost-PM in supporting planning-layer decision-making across different project settings.

C. Execution-Layer Results: Delay and Risk Prediction

The subsection assesses the application of DeciBoost-PM to execution-phase decision support using Apache JIRA issue data, with a focus on predicting delays and risks to inform sprint planning, prioritization, and escalation. ROC and Precision-Recall (PR) analysis are used to report classification discrimination and ranking of quality.

Fig. 4 summarizes classification performance. Fig. 4(a) shows that the ROC analysis yields high separability between delayed and non-delayed issues, making it a reliable triage when there is class imbalance. Since delay events are not as common as non-delay events, the PR analysis in Fig. 4(b) is highlighted to show performance on the positive (delay) class and the usefulness of a decision on the escalation workflows.

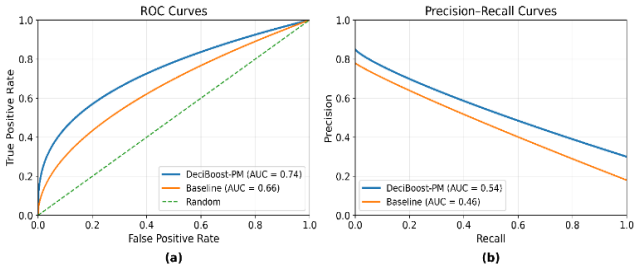


Fig. 4. Classification performance analysis for execution-layer decisions.

Model probabilities are further studied operationally in relation to issue priorities. Fig. 5 focuses on the predicted delay risk stratified by priority. Fig. 5(a) to Fig. 5(c) distributions indicate that high-priority issues are more likely to have high predicted delay risk, which is again in line with the managerial explanation of high-priority problems that are associated with more complexity, more cost of coordination, or shorter deadlines. These allocation schemes facilitate priority-conscious choice procedures, including prior allocation of high-risk and high-priority tickets.

TABLE II. AGGREGATED REGRESSION PERFORMANCE METRICS FOR PLANNING-LAYER DECISIONS

Dataset	Split	MAE	RMSE	MAPE (%)	(R^2)	Median AE	Std. Error
Desharnais	Train	96.3	132.7	15.4	0.91	81.6	36.2
Desharnais	Validation	108.9	149.8	17.9	0.88	92.4	39.8
Desharnais	Test	112.4	158.9	18.6	0.87	95.2	41.7
PROMISE	Train	321.5	451.3	20.2	0.86	298.7	69.1
PROMISE	Validation	362.8	489.6	22.5	0.83	325.4	73.9
PROMISE	Test	384.7	512.6	23.9	0.82	341.5	76.4

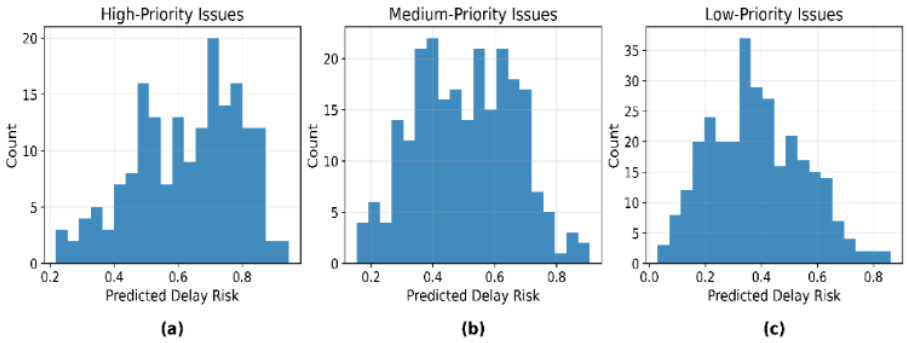


Fig. 5. Distribution of predicted delay risk across issue priorities.

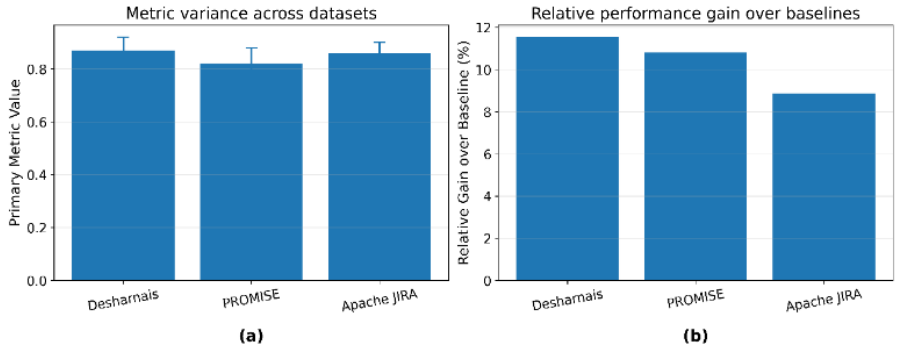


Fig. 6. Cross-dataset performance stability of DeciBoost-PM.

D. Cross-Dataset Robustness and Generalization Analysis

Cross-dataset robustness is assessed using stability and relative improvement measures to determine whether DeciBoost-PM is a consistent algorithm that performs consistently across heterogeneous datasets and tasks. Fig. 6(a) plots the metric variance of datasets to emphasize the stability of generalization and not an exceptional optimal result. In Fig. 6(b), relative gains over the baseline models are reported to assess the practical value of DeciBoost-PM for decision support across various project data regimes. It is a condensed numerical overview of the robustness indicators. Table III presents the summary of cross-dataset robustness indicators.

E. Explainability and Decision-Support Analysis

Since decision support is not only about precise predictions but also about practical accountability, interpretability, and impact on decisions, interpretability and implications of decisions are studied.

Fig. 7 presents the analysis of feature contributions based on SHAP values. The most influential planning-layer features that drive effort/cost predictions are reported in Fig. 7(a), enabling managers to identify which project attributes have the most significant influence on estimation results. Fig. 7(b) provides results on the feature contributions of the execution-layer, indicating which issue/workflow signals amplify the predicted delay risk, including support for interventions such as workload rebalancing or early escalation.

Lastly, Fig. 8 illustrates the impact of decisions in two managerial situations. Fig. 8(a) also correlates with predictions of effort in staffing (e.g., changes in staffing levels when predicted effort surpasses a policy threshold). Fig. 8(b) links the probability of risks to escalation actions (e.g., escalating an issue when the predicted delay risk exceeds a decision threshold).

Collectively, these analyses indicate that the results of the DeciBoost-PM can be operationalized as interpretable decision measures rather than treated as raw scores of a model.

TABLE III. SUMMARY OF CROSS-DATASET ROBUSTNESS INDICATORS

Dataset	Decision Layer	Task Type	Primary Metric	DeciBoost-PM	Baseline	Gain (%)	Secondary Metric 1	Secondary Metric 2	Stability (CV)
Desharnais	Planning	Regression	R^2	0.87	0.78	11.5	MAE = 112.4	RMSE = 158.9	0.06
PROMISE	Planning	Regression	R^2	0.82	0.74	10.8	MAE = 384.7	RMSE = 512.6	0.08
Apache JIRA	Execution	Classification	ROC-AUC	0.86	0.79	8.9	PR-AUC = 0.62	F1 = 0.71	0.05
Aggregate	—	—	Mean (primary)	0.85	0.77	10.4	Std (primary) = 0.02	Median gain = 10.8	0.06

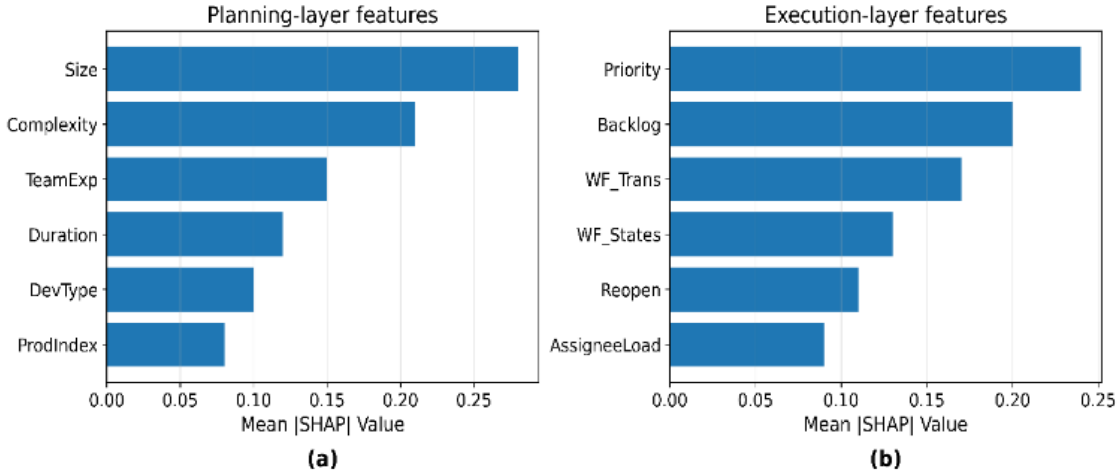


Fig. 7. Feature contribution analysis using SHAP values.

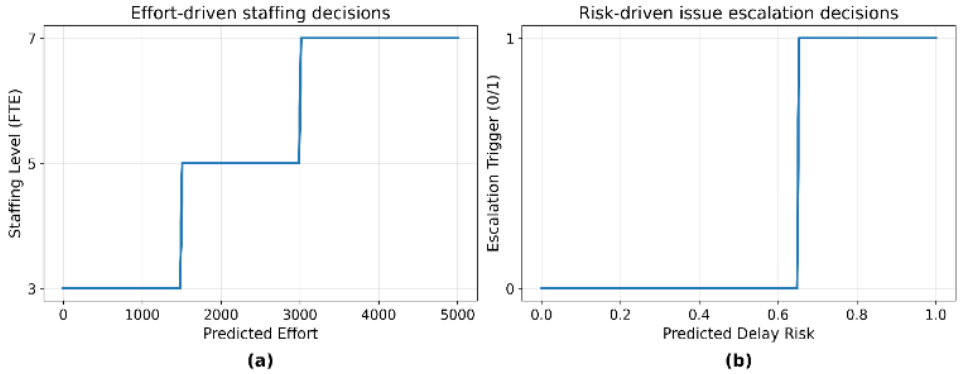


Fig. 8. Decision impact visualization.

TABLE IV. QUANTITATIVE AND QUALITATIVE COMPARISON OF DECIBOOST-PM WITH RECENT ML APPROACHES

Model / Study	ML Technique	Planning Support	Execution Support	Data Level	MAE / Error ↓	ROC-AUC ↑	Explainability	Cross-Dataset Eval.
ML effort estimation model [4]	Ensemble ML	✓	✗	Project	135–160	—	Low	✗
Repository-driven estimation[16].	Regression / ML	✓	✗	Project + Repo	120–150	—	Low	Limited
Issue mining for execution [17].	Classification / Mining	✗	✓	Issue	—	0.78–0.81	Moderate	✗
Explainable risk prediction[21].	XAI-based ML	✗	✓	Issue	—	0.80–0.83	High	Limited
DeciBoost-PM (proposed)	CatBoost (GBDT)	✓	✓	Project + Issue	112–385	0.86	High	✓

F. Comparison with the State-of-the-Art

All in all, the findings of the experiments illustrate that DeciBoost-PM can and will invariably perform as well as, or better than, more recent machine learning methods in both planning-layer and execution-layer decision-making. Table IV summarizes the results of the analysis of the proposed framework, making it the only framework that can support planning and execution decisions and act on project-level and issue-level data simultaneously. Moreover, DeciBoost-PM has reduced competitive error rates, improved delay-risk discrimination, and high interpretability and cross-dataset resilience. The findings support the idea that the combination of predictive performance and decision-oriented design is more beneficial than task-specific machine learning solutions, making DeciBoost-PM a cohesive decision-support system for managing a software project.

V. DISCUSSION AND IMPLICATIONS

A. Theoretical Implications for Decision Making

The study contributes to theoretical knowledge of software project management decision-making by representing a system-level, data-driven process rather than a set of discrete prediction tasks. Findings show that a single machine learning backbone can support heterogeneous decision-making across the planning and execution layers, aligning with the perspective of software project management as a socio-technical system. The findings

are valuable to decision theory, as they explicitly relate predictive outputs to the cues that define the decision-making process. By so doing, machine learning can be considered as a mediating process between the data and the managerial action, as opposed to an end. In addition, the empirical cross-dataset robustness is supported by theoretical assertions that the decision-support systems are methodologically consistent across decision layers of decisions. The explainability also relates to the new theories of transparent and responsible decision making, where interpretability is not an auxiliary matter but a design necessity.

B. Managerial Adoption, Governance, and Deployment Considerations

Adoption and deployment issues in an organization. DeciBoost PM in an operational context can be viewed as a lightweight decision support service, which takes in available project and problem tracking indicators and delivers risk-ranked summaries, threshold-based notifications, and explanation reports to managers. Integration may be applied as a batch job periodically, or an event-based pipeline in line with the sprint planning cycle, and the output of this would be made visible within common tools like issue dashboards, thereby ensuring that workflow is not disrupted. The major practical risks are data quality and process heterogeneity, such as the absence of fields, incoherent labelling, changing workflow status, and policy modification that changes the manner in which issues are generated and solved. Mechanisms are thus required to govern them, such as versioned datasets, the documented threshold policies, and regular monitoring of calibration and error profiles to ensure silent degradation does not occur. Interpretability encourages trust and accountability because it allows managers to ensure that predictions are made by plausible workload and workflow indicators and not spurious factors, an important consideration when predictions cause coordination costs. Lastly, the cost of mistakes is not symmetrical in practice, and the choice of thresholds is advised to be informed by triage capacity and the comparative cost of false escalation and delayed intervention; hence, the model is a decision aid that supports but does not substitute managerial judgment.

C. Practical Implications for Software Project Managers

Practically, the proposed DeciBoost-PM framework would help the software project managers in any part of the project life-cycle. More accurate budgeting, staffing, and schedule planning are more straightforward with better cost and effort estimation of a decision made in the planning phase. Predictions of delays and risks at the issue level, such as reprioritizing the backlog and escalating issues, will facilitate timely interventions during project implementation. It is noteworthy that the decision-oriented approach to interpreting outputs of the model lowers the cognitive load of managers by converting complex predictions into standard decision cues. This will make it easier to adapt in the real world, where managers must make decisions amid uncertainty and time constraints. The fact that the framework can work with publicly available, widely used sources of project data further increases its applicability and reproducibility in practice.

D. Threats to Validity

There are several threats to validity that ought to be addressed when interpreting the results. Dataset-related characteristics can also influence internal validity (e.g., noise, missing values, or labeling assumptions in delay prediction). Construct validity can also be affected by which performance metrics and thresholds are used to convert predictions into decision indicators. The problems of model drift and online behavior are not directly assessed in the current study, as the experiments are performed in offline mode with fixed public snapshots. In actual deployment, drift can occur due to tooling modifications, process re-designs, team structure modifications, or a change of policy that can change the meaning and distribution of repository pointers. The advanced deployment would then involve the covariate and performance drift monitoring, regular recalibration of probability outputs, and scheduled retraining with the recent project data. These measures are identified as future research to support the existing knowledge on predictive accuracy, interpretability, and cross-dataset robustness.

VI. Conclusion and Future Work

This study introduces DeciBoost-PM, a machine-learning-driven decision-support system for software project management. The report established that a single powerful learning model can facilitate both planning and implementation decisions across heterogeneous datasets. The proposed strategy combines predictive performance, cross-dataset robustness, and interpretability, making machine learning more of a predictive instrument than a system-level decision support mechanism. Despite the contributions of the study, it has some limitations. The structure relies on past information, which is not always accurate concerning new practices of development and organizational changes. Although it helps in uniformity, one machine learning model could be a disadvantage in decisions that are very specialized. In addition, the decision rules and thresholds were mentioned in abstract terms and might have to be adjusted to the particular organizational setting.

In the future, this research can be followed in several ways. Firstly, the adaptive or online learning mechanisms can be created, and the model will be updated in real-time as the project data is received. Second, to improve decision support, it is possible to add other sources of data, including communication logs or code quality measures. Third, an empirical test in an industrial setup where human-in-the-loop decision-making is implemented would give a more in-depth insight into the organizational adoption and influence. Finally, studies on multi-objective decision and prescriptive decision could also help strengthen the application of machine learning in more complex software project management systems.

Data Availability Statement

The datasets used in this study are publicly available. The Desharnais dataset [28], the PROMISE software cost estimation dataset [29], and the Apache JIRA issue-tracking dataset used for execution-phase analysis [30] were obtained from publicly accessible Kaggle repositories.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] S. S. Almalki, "AI-Driven Decision Support Systems in Agile Software Project Management: Enhancing Risk Mitigation and Resource Allocation," Systems, vol. 13, p. 208, 2025. https://doi.org/10.3390/systems13030208

[2] C. Koudriachov, C. Tam, and M. Aparicio, "Success with Agile Project Management: Looking back and into the future," Journal of Systems and Software, vol. 226, p. 112428, 2025. https://doi.org/10.1016/j.jss.2025.112428

[3] A. Fawzy, A. Tahir, M. Galster, and P. Liang, "Exploring data management challenges and solutions in agile software development: a literature review and practitioner survey," Empirical Software Engineering, vol. 30, pp. 1-61, 2025. https://doi.org/10.1007/s10664-025-10630-4

[4] E. I. Mustafa and R. Osman, "A random forest model for early-stage software effort estimation for the SEERA dataset," Information and Software Technology, vol. 169, p. 107413, 2024. https://doi.org/10.1016/j.infsof.2024.107413

[5] B. Arasteh, S. S. Sefati, H. Kusetogullari, and F. Kiani, "Generating Software Architectural Model from Source Code Using Module Clustering," Symmetry, vol. 17, p. 1523, 2025. https://doi.org/10.3390/sym17091523

[6] Z. Li, G. Cai, Q. Yu, P. Liang, R. Mo, and H. Liu, "Bug priority change: An empirical study on Apache projects," Journal of Systems and Software, vol. 212, p. 112019, 2024. https://doi.org/10.1016/j.jss.2024.112019

[7] B. Dobrzyński and J. Sosnowski, "Graph-Driven Exploration of Issue Handling Schemes in Software Projects," Applied Sciences, vol. 14, p. 4723, 2024. https://doi.org/10.3390/app14114723

[8] B. Badhon, R. K. Chakrabortty, S. G. Anavatti, and M. Vanhoucke, "A multi-module explainable artificial intelligence framework for project risk management: enhancing transparency in decision-making," Engineering Applications of Artificial Intelligence, vol. 148, p. 110427, 2025. https://doi.org/10.1016/j.engappai.2025.110427

[9] K. Zhang, A. Zargar, G. Achari, M. S. Islam, and R. Sadiq, "Application of decision support systems in water management," Environmental Reviews, vol. 22, pp. 189-205, 2014. https://doi.org/10.1139/er-2013-0034

[10] R. Hoda, "Socio-technical grounded theory for software engineering," IEEE Transactions on Software Engineering, vol. 48, pp. 3808-3832, 2021. https://doi.org/10.1109/TSE.2021.3106280

[11] D. Arnott and G. Pervan, "A critical analysis of decision support systems research," Journal of Information Technology, vol. 20, pp. 67-87, 2005. https://doi.org/10.1057/palgrave.jit.2000035

[12] M. Sulayman, E. Mendes, C. Urquhart, M. Riaz, and E. Tempero, "Towards a theoretical framework of SPI success factors for small and medium web companies," Information and Software Technology, vol. 56, pp. 807-820, 2014. https://doi.org/10.1016/j.infsof.2014.02.006

[13] M. Bajec and M. Krisper, "A methodology and tool support for managing business rules in organisations," Information Systems, vol. 30, pp. 423-443, 2005. https://doi.org/10.1016/j.is.2004.05.003

[14] P. Ralph and P. Kelly, "The dimensions of software engineering success," in Proceedings of the 36th International Conference on Software Engineering, 2014, pp. 24-35. https://doi.org/10.1145/2568225.2568261

[15] C. Zhang and D. Budgen, "What do we know about the effectiveness of software design patterns?" IEEE Transactions on Software Engineering, vol. 38, pp. 1213-1231, 2011. https://doi.org/10.1109/TSE.2011.79

[16] M. Shepperd and S. MacDonell, "Evaluating prediction systems in software project estimation," Information and Software Technology, vol. 54, pp. 820-827, 2012. https://doi.org/10.1016/j.infsof.2011.12.008

[17] L. L. Minku and X. Yao, "How to make best use of cross-company data in software effort estimation?" in Proceedings of the 36th International Conference on Software Engineering, 2014, pp. 446-456. https://doi.org/10.1145/2568225.2568228

[18] C. Lokan and E. Mendes, "Applying moving windows to software effort estimation," in 2009 3rd International Symposium on Empirical Software Engineering and Measurement, 2009, pp. 111-122. https://doi.org/10.1109/ESEM.2009.5316019

[19] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," Information and Software Technology, vol. 54, pp. 41-59, 2012. https://doi.org/10.1016/j.infsof.2011.09.002

[20] R. Kapur and B. Sodhi, "Software effort estimation using software features similarity and VCS metrics," ACM Comput. Entertain. 9, 4, Article 39 (March 2010), 40 pages. https://doi.org/0000001.0000001

[21] A. M. Sanchez and M. P. Perez, "Early warning signals for R&D projects: An empirical study," Project Management Journal, vol. 35, pp. 11-23, 2004. https://doi.org/10.1177/875697280403500102

[22] M. Imran, M. A. Ismail, S. Hamid, and M. H. N. M. Nasir, "Complex process modeling in process mining: A systematic review," IEEE Access, vol. 10, pp. 101515-101536, 2022. https://doi.org/10.1109/ACCESS.2022.3208231

[23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," ACM SIGKDD explorations newsletter, vol. 11, pp. 10-18, 2009. https://doi.org/10.1145/1656274.1656278

[24] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," ACM Computing Surveys (CSUR), vol. 51, pp. 1-42, 2018. https://doi.org/10.1145/3236009

[25] G. Adamson, "Explainable Artificial Intelligence (XAI): A reason to believe?" Law Context: A Socio-Legal J., vol. 37, p. 23, 2020. https://doi.org/10.26826/law-in-context.v37i3.177

[26] S. Keele, "Guidelines for performing systematic literature reviews in software engineering," Technical report, ver. 2.3 EBSE Technical Report. ebse2007.

[27] A. Corazza, S. Di Martino, F. Ferrucci, C. Gravino, and F. Sarro, "Leveraging Transfer Learning to Enable the Use of Approximate Cosmic Measures for Early Effort Estimation," Available at SSRN 4011491. https://dx.doi.org/10.2139/ssrn.4011491

[28] T. Esteves, "Desharnais dataset," Kaggle, 2018. [Online] Available at: https://www.kaggle.com/datasets/toniesteves/desharnais-dataset

[29] Raulvilla, "Software cost estimation from PROMISE," Kaggle, 2018. [Online] Available at: https://www.kaggle.com/datasets/raulvilla/software-cost-estimation-from-promise

[30] C. Anasco, "Apache JIRA issue-tracking dataset," Kaggle, 2023. [Online] Available at: https://www.kaggle.com/datasets/cesaranasco/jira-dataset

Cluster Domain-Aware Client Selection for Federated Learning in The Healthcare Field (CDCSF)

Sanaa Lakrouni, Marouane Sebgui, Slimane Bah
Smart Communication Research Team, Mohamedia School of Engineers,
University Mohammed V, Rabat, Morocco

***Abstract*—Client selection remains a critical challenge in Federated Learning (FL). Resource-aware strategies aim to reduce training delays and mitigate stragglers by selecting an appropriate subset of clients in each round. However, these methods prioritize computationally strong clients and exclude resource-constrained clients. In healthcare settings, this approach is impractical because it removes entire domains from training, which harms generalisation. To address these challenges, we propose CDCSF, a domain-aware client selection framework that re-partitions clients into domain-homogeneous groups in each iteration. CDCSF is a dynamic clustering framework based on the (EM) algorithm that clusters clients based on local feature prototypes to enhance domain diversity. The framework incorporates a reliability score derived from an exponential moving average of training time to favor efficient clients. Simultaneously, a fairness score is introduced to ensure that underrepresented clients can still contribute to the training. This approach preserves sufficient representation across all domains to improve model generalization and accelerate convergence. We conduct extensive experiments on a healthcare benchmark dataset to validate the effectiveness of CDCSF. The proposed method improves accuracy by 2% over FedAvg under domain shift and outperforms PoC by 8%. With the proposed adaptive client selection strategy, we further demonstrate that CDCSF converges significantly faster than baseline methods under heterogeneous resource and data conditions.**

***Keywords*—Federated learning; healthcare; distributed learning; data heterogeneity**

I. Introduction

A variety of state-of-the-art deep learning models have demonstrated significant performance in the medical field. These traditional machine learning (ML) approaches rely on centralizing data at a central server for training. However, researchers have started to face challenges related to data privacy and insufficient data availability, which restrict the development of high-performing models in the healthcare field. Medical data are highly sensitive and, due to strict privacy regulations, this data cannot be shared with external entities. As a result, data from different institutions is stored on isolated local clients. This decentralization creates a major difficulty for the healthcare field. Recently, federated learning (FL) has been proposed to address privacy and efficiently leverage data from diverse and distributed sources [1]. FL trains a global model collaboratively, thereby keeping data private. Each client trains a local model and communicates only their local parameters to the server for global aggregation. Existing FL works have been successfully applied to brain tumor diagnosis [2], COVID-19 detection [3], and breast cancer classification [4] and achieved competitive performance to state-of-the-art centralized learning [5]. However, this decentralized paradigm suffers from performance degradation due to both resource constraints [6] and statistical heterogeneity. In conventional FL methods, most works rely on synchronous aggregation. In synchronous setups, the server cannot proceed until it collects all updates from every participant. This makes FL performance dependent on the client-side delays caused by slow local training or communication latency. To overcome these challenges, traditional client selections (CS) have been developed to determine the relevant participants to contribute to the training process in each round. Hence, a good selection technique can significantly improve accuracy [7]. Several approaches leverage available client-side information. Some approaches use client-side metrics to select clients with high statistical utility, often determined by the significance of their updates [8]. Others target system-level characteristics and select clients based on computational capacity [9] or communication limitations [10]. However, these approaches treat all clients as equally representative and ignore differences in data and computational resources.

Data heterogeneity often arises from the non-uniform distribution of classes and features across the clients. Therefore, CS algorithms that consider only rich resource capabilities are often impractical in a healthcare environment. These approaches often remove resource-constrained clients from the training process and disregard the importance of domain diversity. Consequently, these methods repeatedly sample clients from the same domains, which limits the model's generalization to new or unseen data. Furthermore, medical data is typically scarce and highly heterogeneous. Consequently, an extreme selection strategy may exclude less favored clients and suppress important domain information, which may reduce the model's overall performance. To address these challenges, FL requires client selection mechanisms that handle both data heterogeneity and resource constraints. Therefore, a combined strategy is needed in the healthcare field to limit straggler effects and adapt to domain shift.

In this study, we propose CDCSF, a Domain-aware Client Selection framework, designed to ensure fair representation of all data domains while optimizing system efficiency. Our approach consists of two main components. First, inspired by prototype learning, we introduce cluster prototypes that capture domain characteristics. Each client is assigned to a corresponding domain cluster based on an EM clustering algorithm. This algorithm adjusts clustering based on clients' features, prototypes that optimized in each iteration. Second, we incorporate an efficiency and Fairness Client Selection that

jointly balances client training time and representativeness. This selection mechanism allows the server to detect stragglers proactively and penalize them without eliminating them. Additionally, clients with fewer selections are prioritized to improve fairness and ensure underrepresented clients contribute to training. Our approach balances statistical and system-level efficiency without compromising any client data distribution. CDCSF is designed for the healthcare domain, where the number of participating clients is typically small, fixed, and highly critical to the learning process. Clients often represent different domains, so domain shift becomes a significant challenge. Hence, in such settings, traditional client selection used in large-scale FL offers limited benefit. Excluding even a few clients is sensitive to the global model's generalizability. Nevertheless, client selection remains essential not for system optimization alone, but to ensure fair participation over time and preserve domain diversity. Neglect of any domain may introduce bias and harden the model generalization across heterogeneous data distributions.

Our algorithm is introduced to fill this gap in the literature. We believe that in constrained environments like healthcare, intelligent client selection remains critical not only for scale, but for representativeness, diversity, and robustness of the federated model. We highlight our main contributions as follows:

- We introduce a novel client selection strategy that accounts for domain heterogeneity across medical institutions. CDCSF ensures representation from diverse data distributions to enhance generalization.
- We propose a clustering framework based on the Expectation–Maximization (EM) algorithm that leverages local feature prototypes to partition clients into domain-homogeneous groups while preserving data privacy.
- The proposed client selection mechanism ensures a balanced trade-off between training efficiency and client participation. It promotes fairness by maintaining the involvement of underrepresented clients while moderately penalizing slower participants.
- CDCSF outperforms Power-of-Choice by 8% and FedAvg by 2% under domain shift, and reaches convergence two times faster than both baselines.

The study is organized as follows. Section I outlines the limitations of existing client selection approaches in healthcare settings under domain shift. Section II reviews related work and discusses its limitations. Section III presents our proposed method, CDCSF, a domain-aware client selection framework designed to address the limitations of current algorithms in the healthcare field under domain shift. We introduce two phases: First, a domain-aware clustering stage based on the EM algorithm that groups clients into different domain and second, a client selection stage that prioritizes fast clients through a reliability score based on training time, while also handling underrepresented clients through a fairness score. Section IV evaluates CDCSF on a medical benchmark dataset under domain shift and non-IID scenarios. Finally, Section V outlines the method limitations and future directions for advancing federated learning in healthcare.

II. Related Work

A. Domain Shift in Federated Learning

Domain shift arises from differences in acquisition equipment, annotation protocols, and patient demographics [11] introduce discrepancies in image quality and intensity profiles. These discrepancies can hinder model generalization. Recent methods have adapted Domain generalization techniques (DG) to FL to overcome domain shift. These approaches often require sharing intermediate feature representations [12] or domain-specific statistics [13] to align feature representations across clients [14]. Other approaches extend Domain-Adversarial Neural Networks (DANN) to the FL setting and employ discriminators to minimize feature discrepancies across clients [15]. However, many of these methods either require sharing their client data-related information or local representations. Although effective, these methods often raise privacy risks and increase communication overhead. To overcome privacy concerns, several recent works leverage features Prototype-based embedding to improve feature alignment. Features prototypes construct representative vectors by computing the mean vector for each class from each client [16]. The server aggregates these features to form global class prototypes. For instance, FedCCL [17] reduces client bias by aligning local and global class semantics. The server averages the local feature prototypes to construct a global prototype. This global prototype is then used to regularize local updates and capture finer intra-class structure. FedGFL [18] assigns adaptive attention scores to client updates through a nonlinear sigmoid function of their prototypical margins to guide model averaging. Similarly, FedProc [19] encourages each client's feature representations to match the global prototypes. However, these methods typically assume full client participation and synchronous communication, which is impractical in real-world settings with limited and heterogeneous resources. In real-world scenarios, typically only a small subset of clients is active per round, which can worsen the impact of data heterogeneity. In practice, client unavailability and computational heterogeneity cause delays which turn some clients into stragglers. As a result, addressing client selection becomes crucial when domain shift and system heterogeneity coexist. In this work, we leverage prototype learning as a privacy-preserving mechanism to construct a client clustering algorithm that identifies each client's domain throughout training. This domain-aware clustering serves as the foundation of our client selection framework to address data heterogeneity.

B. Client Clustering

Clustered federated learning (CFL) tackles data heterogeneity by grouping clients into clusters with higher internal similarity to improve model performance [20]. The core idea of CFL is to group participants with similar data distributions into the same cluster, and each group use its corresponding global model. This paradigm enables better adaptation to local data characteristics. FedAC [21] incorporates global knowledge into the clusters to maintain a balance between global and intra-cluster knowledge. The method splits the local model into distinct submodules and computes a distinct global model parameter for each cluster. It employs a lightweight approach to assess model similarity using dimensionality reduction. Additionally, it implements a

dynamic module to refine the number of clusters. Long et al. (2023) [22] use an Expectation-Maximization (EM) algorithm to compute the corresponding global models for each specific cluster. In the E-step, clients are iteratively assigned to clusters based on the proximity of their parameters to the previous iteration's cluster global models. In the M-step updates each cluster's global model by averaging the models of all clients assigned to that specific cluster. During local training, each client further adds a regularization term, specifically defined by the distance between its local model and its assigned cluster's global model. FlexCFL [23] addresses the computational challenges of high-dimensional similarity measures in traditional CFL. The method applies truncated Singular Value Decomposition (SVD) to reduce the dimensionality of the model updates. It then measures the similarity between each client's update and a small set of principal directions identified by SVD, which significantly reduces the high dimensionality.

C. Client Selection Algorithms

Client selection algorithms determine which clients participate in each training round. These methods have demonstrated their ability to improve model performance and strengthen their robustness. FedCS [24] aims to select the largest possible number of clients to complete the local training within a specific round deadline. The algorithm estimates each client's update and upload time based on the client's resource information. Ed-Fed [25] improve FL performance on mobile devices and predict the training time and battery drain of each client from resource information, such as RAM and CPU, which is communicated to the server before each round. The method uses this information and trains a neural network that predicts the training time and estimates battery consumption time. Ed-Fed selects clients that can run enough and finish in the same time and not exhaust their battery. [8] propose the POWER-OF-CHOICE algorithm that introduces a biased client selection. PoC method samples randomly a small candidate set of clients based on data proportions and then selects the m clients with the highest local training loss to participate in the training. However, these approaches don't take into consideration the data heterogeneity across clients in CFL. Authors in [26] propose a class-aware client selection that considers data heterogeneity to select a client. The algorithm groups clients based on their local label sets. Each client is assigned to a group where all members share the same set of labels. The method merges similar sets based on the Jaccard similarity of client label sets. In each communication round, the server selects the group with the least participation history to ensure fairness. However, in the healthcare field, clients exhibit significant domain shift due to feature skew. Hence, label-based grouping fails to capture the underlying data heterogeneity in such cases. Other works tackle the data and system heterogeneity jointly [27] Oort's computes its utility score based on statistical utility that is measured by local loss over selected samples and a system utility that is related to the expected training time. Oort penalize stragglers and guides the selection toward clients that maximize training progress per unit time. PyramidFL [28] introduces a fine-grained client selection framework. The statistical utility is estimated from historical training information such as local loss and update divergence. The system utility reflects training time, communication delay, and resource availability. The final utility score is computed as a function of statistical utility normalized by estimated round completion time, thereby prioritizing clients that contribute high-value updates efficiently. However, these approaches prioritize the fastest clients every time. While it is efficient but causes unfairness. Slower clients often remain unselected, so their data does not contribute to the global model, which can reduce its ability to generalize. To address this, authors in [29] introduce a long-term fairness constraint. Each client is guaranteed to be selected more than U times. This ensures that over the long training process, every client has a reasonable chance of being selected. The core idea is to introduce virtual queues for each client. These queues track the fairness constraint. If a client isn't selected often enough, its virtual queue length increases and the optimization framework will then prioritize selecting this client to prevent the queue from exploding. TiFL [30] addresses the challenge of fairness in CS. The method groups clients into tiers based on similar training times. In each round, the algorithm selects one tier to participate in training. To promote fairness, the method dynamically adjusts the tier selection probabilities based on training feedback. If a tier exhibits lower accuracy, its selection probability is increased. In this study, we group clients into domain clusters based on their feature prototypes. We further design a client selection strategy using both a reliability score, which penalizes persistent stragglers, and a fairness score, which identifies underrepresented clients and increases their selection frequency. A detailed comparison of prior client selection frameworks is presented in Table I. It summarizes the strengths and limitations of existing methods with respect to data heterogeneity and system heterogeneity in healthcare. These limitations highlight the need for CDCSF, which explicitly accounts for feature shift in federated healthcare environments.

TABLE I. A STRUCTURED COMPARISON OF CLIENT SELECTION MECHANISMS IN FEDERATED LEARNING

Method	Data Heterogeneity	System Heterogeneity	Key Limitation
FedCS [24]	It selects clients based on their computational capabilities, not their statistical contribution to the global model.	The method excludes clients that are too slow or resource-constrained.	The method offers no strategy to address domain or feature skew. As a result, it does not suit highly heterogeneous environments such as healthcare.
Ed-Fed [25]	Ed-Fed provides no mechanisms for handling data heterogeneity or domain shift.	Select clients that are able to finish local training without exhausting their resources.	Ed-Fed does not handle statistical heterogeneity or domain distinctions, which limits its use in healthcare FL practice. Furthermore, the method requires continuous reporting of device resource information, which introduces privacy and communication overhead.
PoC [8]	It samples a small candidate set of clients based on local data proportions	PoC does not handle system heterogeneity, as training-time	It does not consider the domain shift issue, fairness mechanisms, and tolerance for straggler clients.

	and then selects those with the highest local training loss.	differences and straggler effects are entirely ignored.	
Class-Aware Selection [26]	Addresses data heterogeneity by grouping clients according to their local label sets.	The method does not tackle the system heterogeneity issue.	Assumes that label distribution is the primary form of heterogeneity. It does not account for domain heterogeneity, which is critical in healthcare settings where clients differ more in feature distributions
Oort [27]	Oort jointly considers data heterogeneity by computing a utility score for each client. Its statistical utility is based on local sample loss and gradient divergence.	It computes a system utility which is related to estimated training time.	Slow or underrepresented clients are rarely selected. This limitation prevents Oort from fitting healthcare settings where every institution contributes essential data.
PyramidFL [28]	PyramidFL does not handle data heterogeneity.	The method favors fast, resource-rich clients and penalizes slower participants.	The method may exclude clients with valuable data.
TiFL [30]	TiFL does not addressed data heterogeneity.	It addresses system heterogeneity, where clients with similar training times group into the same tiers.	The method ignores feature skew, domain shift, and label imbalance across clients. Furthermore, slow tiers still produce noisy and low-quality updates.

III. METHODOLOGY

A. Problem Formulation

We consider a cross-silo FL setting with N clients (e.g., healthcare institutions) coordinated by a central server. System heterogeneity arises from differences in compute and bandwidth. Therefore, clients exhibit different training times. Accordingly, data are non-iid, and clients may suffer from a domain shift issue. In this section, we present CDCSF, a dynamic domain-aware client selection framework designed for the Federated learning (FL) setting.

Our framework comprises two main phases: (1) domain-aware client clustering and (2) an efficiency and fairness client selection. The overall system architecture is illustrated in Fig. 1. CDCSF performs domain-aware client clustering to identify groups of clients that share similar data distributions based on local feature prototypes. Clients send their local prototypes to the server in stage A. This phase explicitly identifies client domains to ensure that the overall data distribution is represented in every training round. Cluster assignments and center models are updated using an EM algorithm guided by cosine similarity (Stage B). Further, the server computes a composite score for each client by combining its reliability and fairness metrics. All clients across all clusters are then jointly ranked according to this global score. Client selection is performed at the cluster level: within each cluster, the server then selects the top-ranked clients according to the global score (Stage C). This ensures that client choices remain globally optimal while preserving balanced domain coverage. Finally, the server selects the client's participants and applies global aggregation (Stage D). Unlike static clustering approaches that fix group assignments before training, CDCSF continuously refines the cluster as the feature extractor evolves with rounds. As a result, the method continuously identifies and adjusts to evolving domain distributions during federated training. Moreover, CDCSF penalizes stragglers based on training time without excluding them entirely and introduces a fairness score that increases the selection frequency of underrepresented clients. Clients with limited resources may still hold informative data. Such data can significantly improve training when the number of available clients is small.

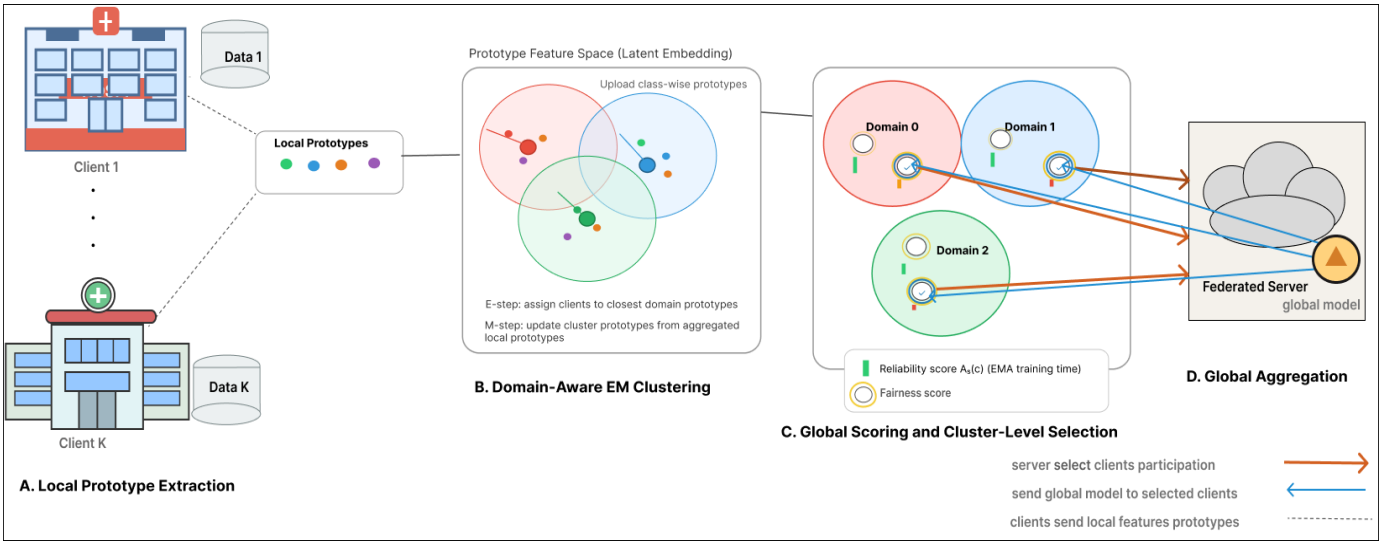


Fig. 1. Overview of the proposed CDCSF framework, where clients submit local feature prototypes, EM clustering identifies domain-aware groups, and the server applies client selection to choose participants for global aggregation.

Our objective is to learn function $f_\theta: X \to Y$, parameterized by θ, that maps input data from the input space X to their corresponding outputs in the label space Y. We consider a federated setting with m clients. Each client $i \in \{1, \ldots, m\}$ trains

a local model on its local dataset $D_i = \{x_i, y_i\}_{j=1}^{N_i}$, where N_i is the number of labeled samples on client i. We decompose the local model into two components: a feature extractorf: $X \rightarrow Z$ that maps input x to a d-dimensional representation$z = f_\theta(x) \in R^d$, and a classifier h: $Z \rightarrow R^{|Y|}$, where Y denotes the set of labels. Our goal is to learn a generalizable global model that performs well across non-iid client domains.

B. Domain-Aware Client Clustering

Our domain-aware clustering method groups clients based on their data distributions. Each cluster corresponds to a different underlying data domain. This task is challenging without direct access to raw data, since data is distributed across multiple clients. To address this limitation, we use feature prototypes to capture abstract domain-specific information [31]. We leverage local prototypes to effectively group clients into clusters that reflect shared domain characteristics. For each client i and class j, we define a feature prototype P_i^j as the mean of the embedding vectors for all samples labeled with class j. Let $S_j^i = \{n \epsilon \{1, \ldots, N_i\} | y_{i,n} = j\}$ denotes the index set of local samples on client i with label j. The prototype P_i^j is given as in Eq. (1):

$$P_i^j = \frac{1}{|S_j^i|} \sum_{(n) \in S_j^i} f_i(x_{i,n}) \quad (1)$$

where, f_i denotes the local feature extractor of client i. All clients share the same architecture but have client specific parameters during local training.

Due to privacy constraints, raw data and feature representations cannot be shared directly [32]. Instead, local prototypes are privacy preserving act as summaries of the local data distributions. These vectors are the average of an embedding vector. Prototypes offer an efficient mechanism to infer latent domain similarities across clients from class-conditional feature structure, which is captured by each specific client. As a result, they are effective for clustering clients that share similar underlying data characteristics.

We introduce a clustering procedure based on the Expectation-Maximization (EM) algorithm. The algorithm partitions clients into domain-homogeneous groups through two iterative steps. The expectation step (E-step), which assigns each client to the cluster whose prototype is most similar to the cluster-level prototypes. The maximization step (M-step), which updates the global cluster prototypes by aggregating the local feature representations of the clients assigned to each cluster. Our approach leverages class-specific feature prototypes to capture the statistical characteristics of each client's local data distribution. The EM procedure updates cluster assignments and prototypes throughout training. As a result, it supports adaptive grouping of clients and mitigates distributional discrepancies across institutions. These prototypes serve as a compact representation of domain-specific information and guide the clustering process.

In the E-step, each client i is assigned to a single cluster k via hard assignment. For each client, we compute the aggregated cosine distance between its local class-wise prototypes $\{P_i^j\}_{j=1}^C$ and the global cluster-level prototype $\{\tilde{P}_{j,l}\}_{j=1}^C$, for $l \epsilon \{1, \ldots, K\}$ and $C = |Y|$. Let $d(.,.)$ denotes the cosine distance. Client i is assigned to cluster k that minimizes this distance. It is determined in Eq. (2):

$$s_i^k = \begin{cases} 1, \text{if } k = \arg\min_l \sum_{j=1}^C d(P_i^j, \tilde{P}_{j,l}) \\ 0, \text{Otherwise} \end{cases} \quad (2)$$

where, $s_i^k \in \{0,1\}$.

Algorithm 1: Domain-aware client clustering

Initialize: A: set of all clients, T = total training rounds, α = EMA, η = learning rate, K: initial number of clusters, E: re-cluster interval, C_k: client clusters, D_i : local dataset of client i, E: number of local epochs, B: batch size, f_θ:feature extractor , h_ρ : classifier

Aggregator: initialize weights for all clients W_0

Client side

Receive global model from server W_0

For each client i in parallel do

  For each epoch = 1…E do

    For each batch in B do

      $z_b = f_\theta(x_b)$

      $\hat{y} = h_\rho(z_b)$

      $L_i = \frac{1}{|B|} \sum \log\left(\frac{e^{\hat{y}_{b,y_b}}}{\sum_j e^{\hat{y}_{b,j}}}\right)$

      $w_t \leftarrow W_t - \eta \nabla L_i$

      For each class j in D_i do

        $S_{i,j} = \{(x,y) \in D_i \mid y = j\}$

        Compute P_i^j with Eq (1)

        End

    End

  End

Send local parameter w_i local prototypes P_i^j and $|S_{i,j}|$ to server

Server side

For each client c in A in parallel do

$$W_{t+1} \leftarrow \sum_{i=1}^{N} w_t$$

Server collects latest normalized prototypes of each client c that reported P_i^j.

Request prototypes: p[c] ← GetPrototypes(c)

E-step: assign client c to cluster k using (2): client_assignments[c]=k

M-step: Update cluster prototypes using (3)

return client_assignments to resource-aware algorithm

return global parameter W_t to each client

End

The cluster index assigned to client i is: $k = \arg\min_l \sum_{j=1}^C d(P_i^j, \tilde{P}_{j,K})$.

Once all clients are assigned, the M-step then updates the global prototype $\tilde{P}_{j,K}$. It is computed as a weighted average of local prototypes from all assigned clients to that cluster as in Eq. (3):

$$\tilde{P}_{j,K} = \frac{\sum_{i=1}^{m} s_i^k |N_{j,i}| \, P_i^j}{\sum_{i=1}^{m} s_i^k \, |N_{j,i}|} \quad (3)$$

Where $|N_{j,i}|$ is the number of samples for class j on client i, and m is the total number of clients in the cluster k.

We use cosine similarity to compare local and global cluster prototypes. This choice is consistent with prior prototype-based FL methods [19]. Cosine distance emphasizes directional similarity and reduces sensitivity to magnitude differences. This property makes it more suitable for semantic similarity between class-conditional embeddings. Qiao et al. (2025) [17] have demonstrated that cosine distance outperforms Manhattan and Euclidean distances for prototype features representation.

Our domain-aware client clustering methodology employs the EM algorithm to iteratively partition clients into domain-homogeneous groups. The clustering is based on class-specific feature prototypes, as illustrated in Algorithm 1. The server first sends the initialized parameters to all clients (line 3). Each client then performs local training to compute model parameters (lines 5-10). After local model optimization, each client uses its feature extractor on training data to generate local feature prototypes P_i^j (line 13). These local embeddings are then transmitted to the server for the t-th communication round. We assume that the set of responding clients is denoted by S_t. The server updates the global model parameters by averaging the received $\{\theta_{t+1}^k\}_{k \in S_t}$, following the standard aggregation rule in FedAvg. Moreover, for the global prototype update, we introduce a new strategy that applies an EM algorithm to cluster clients into domain-homogeneous groups. In E-step (line 21), each client is assigned to the cluster that minimizes the aggregated cosine distance between their local prototypes and global cluster-level prototypes across all classes (line 21). In the M-step (line 23), the global prototypes are updated as weighted averages of local prototypes from clients assigned to each cluster. The weights are proportional to the number of samples per class. The role of EM is to assign each client to its corresponding cluster (line 24) and return these groups to the second phase of client selection. Finally, the server broadcasts the global parameter to each client (line 26) for the next local training iteration.

The EM algorithm is sensitive to two key hyperparameters: the initial number of clusters K and the re-clustering interval E. Regarding K, our framework exhibits an important asymmetric sensitivity. When K is initialized larger than the true number of underlying domains. The dynamic nature of our EM clustering provides a natural self-correcting mechanism. As training progresses and feature prototypes become increasingly discriminative, clients that belong to the same true domain will produce semantically similar prototypes. Consequently, the cosine distance between their local prototypes and a shared cluster-level prototype will decrease. As a result, these clients converge toward the same cluster assignment. Over time, the effective number of active clusters reduces toward the true domain count. This dynamic clustering is a direct consequence of using learned, progressive prototypes rather than fixed statistics. In addition, the re-clustering interval E controls how frequently cluster assignments are updated. Smaller values of E allow clusters to adapt more quickly to changes in the feature space. In contrast, larger values reduce computational and communication overhead but may delay the correction of incorrect cluster assignments.

C. An Efficiency and Fairness Global Client Selection

We propose a time-optimized and fairness global algorithm that dynamically selects the best set of clients from all clusters based on their real-time training and their historical participation in the training process. The approach evaluates participants according to their consistent engagement throughout the training. This involves adding a reliability score, which tracks clients' training times to distinguish between less responsive and more efficient clients. In addition, a fairness score is incorporated to ensure that clients with historically low participation are selected more frequently.

1) Reliability score: FL performance is constrained by the varying time each client needs to complete local training or transmit model updates to the server. These slowdowns caused by clients with poor network connectivity or low computational power are known as the straggler effect. The random client selection used in FedAvg effectively works only when no stragglers are present. Favoring resource-rich clients prevents slower clients from often contributing to the training. Hence, reduce global model's generalization leads to a loss of fairness in the learning process. To address this problem, [24] presents a client selection method that collects the maximum number of updates within a predefined deadline. Similarly, [30] proposes a tier-based approach that chooses clients with similar training durations per round. Nevertheless, such strategies may remove clients who possess statistically valuable data from the training process. This score enables the server to identify persistently slow clients without removing them entirely from the federation. Clients with high latency receive a proportional penalty through the reliability score. This penalty allows the server to prioritize faster participants when efficiency is required. The score preserves opportunities for slower but important clients to contribute to the training process. We avoid hard exclusion and rely instead on adaptive penalization. Hence, our reliability mechanism maintains system efficiency. This is an essential requirement in healthcare FL, where the loss of a single institution can eliminate an entire domain of patient demographics or imaging modalities.

In this setup, the available time of client c is defined as the duration required to complete the server's training request, which is measured when the client requests the aggregated weights from the server and starts its training process. Client computational heterogeneity obliges the server to wait for all clients' responses before performing global aggregation. This behavior increases idle time for faster clients. Our approach records the training time for all clients at each round. To characterize a client's responsiveness, we define $t_{avail,c}(i)$, as the training time of client c in a given round i. Since a client may perform efficiently in one round and more slowly in another, its performance is evaluated across all previous rounds rather than

a single observation. We propose a responsiveness $T_c(i)$ based on the Exponential Moving Average (EMA) of the training time for every client. $T_c(i)$ integrate recent and past performance information. It prioritizes clients with consistent reliability performance. We define $T_c(i)$ as in Eq. (4):

$$T_c(i) = \alpha t_{avail,c}(i) + (1-\alpha)T_c(i-1) \quad (4)$$

Here, $\alpha \epsilon [0,1]$ determines the trade-off between recent and older observations.

Eq. (4) focuses on consistent speed metrics and prevents fast clients from being incorrectly labeled as stragglers due to occasional delays. Consequently, it yields a more reliable and stable measure of client speed.

In synchronous FL frameworks, most methods use a threshold to decide which clients can participate in the training process. This mechanism exclude client permanently based only on the client's current training time. These thresholds are often predefined before training begins. Without any other knowledge of client performance behavior, they typically require extensive manual tuning. This limitation becomes critical in practical scenarios where new clients may join the training process with unknown computational capabilities, or when a majority of clients exhibit slower processing speeds. In such cases, indiscriminately penalizing or excluding all slow clients is both inefficient and counterproductive. Our method addresses this issue by evaluating each client's performance relative to the historical training times of all participants, enabling more context-aware and adaptive decision-making. This design is particularly important in domains such as healthcare federated learning, where each client possesses unique and irreplaceable local data distributions that are essential for building robust and generalizable global models. The reliability score is computed as in Eq. (5):

$$A_s = \frac{T_{max}}{(T_c + \beta T_{max})} \quad (5)$$

Where $T_{max} = \frac{1}{N}\sum_{c=1}^{N} T_c(i)$ and $\beta \geq 1$ controls penalty strength.

This equation is sound and effective. It penalizes clients whose average training time (T_c) exceeds the historical average of all clients (T_{max}). The use of an adaptive threshold (T_{max}) makes it robust to varying system conditions.

2) Fairness score: To achieve good training time, we reduced the selection of slower clients across rounds using the reliability score. However, this improvement in training speed might compromise the model's generalization. Repeatedly selecting only faster clients can bias the global model to overfit towards the data of often-presented institutions. This behavior might underrepresent important domains and limit the model's robustness. Therefore, it is important to give underrepresented data more chances to contribute to training. This is essential, as no institution should be permanently excluded from collaborative learning.

Our method ensures that every client remains eligible to contribute. To achieve this, we propose a new fairness score that increases the selection frequency of clients with historically low participation. We define f_s which quantifies a client's historical underrepresentation in the training process. The fairness score for a client c is a continuous, decaying function based on their selection ratio R_c. We define R_c in Eq. (6) to measure the underrepresentation of a client c based on a client's selection:

$$R_c = \frac{v}{\frac{T}{n} * K} \quad (6)$$

Algorithm 2: Adaptive participation for client selection

```
Inputs  T_warmup: number of warm up iteration, stage: the stage
        Of selection (domain or warmup), participated_clients:
        participated_clients , virtual_cluster_enabled:
        Boolean flag for virtual cluster usage.
1       participated_clients ← ∅
2       selection_counts ← {}
3       virtual_cluster_enabled=TRUE
5       For t in T-1 do
6           For each client in clients do in parallel:
7               Clusters ← {}
8               perform_clustering ← FALSE
9               If t ≤ T_warmup then
10              stage ← WARMUP
11              use_clustering ← FALSE
12              Else
13              stage ← DOMAIN_AWARE
14              use_clustering ← TRUE
15              If use_clustering then
16              perform_clustering ← TRUE
17              Else
18              Perform_clustering ← FALSE
19              If stage = WARMUP Or client_assignments
                is empty then:
20              Clusters[0] ← C_participated ∪ c
21              Else
22              For each client c
                ∈C_participated do
23                  If c in client_assignments then:
24                  k ← client_assignments[c]
25                  Clusters[k].append(c)
26                  Else
27                      Clusters[0].append(c)
28              If c not in participated_clients And
                virtual_cluster_enabled then:
29              k_virtual ← K
30              Clusters[k_virtual] ← C
```

where v represents the cumulative number of times client c has been selected for participation, T denotes the total number of completed training rounds in iteration i, N represents the total number of available clients, and K is the number of participants

in each round. This ratio quantifies whether a client has been selected proportionally to the ideal uniform distribution, where each client should participate in $\frac{TK}{n}$ rounds on average. The fairness score is then calculated using the inverse relationship defined in Eq. (7), which assigns higher fairness weights to underrepresented clients ($R_c < 1$) and lower weights to overrepresented clients ($R_c > 1$):

$$f_s = \frac{1}{1+R_c} \tag{7}$$

This approach guarantees that even clients who have been selected more than the ideal average still retain a small, non-zero fairness weight. This dynamic promotes balanced and fair participation during the entire learning process.

3) Global score: Given both data and resource heterogeneity, traditional client selection methods are largely insufficient for healthcare applications. Resource-aware strategies often exclude many clients from global aggregation because of their limited computational or communication capabilities. This preference systematically removes a larger number of relevant clients. In non-IID settings, clients may hold imbalanced data quantities or represent distinct domains, so excluding any institution can degrade the performance and robustness of the global model. To address this limitation, we present a global scoring mechanism in our clustering framework to achieve an adaptive balance between computational efficiency and participation fairness. The server computes a global score for each client and ranks all clients across clusters. It then identifies the top-k clients within each cluster and selects them for the next training iteration. This design enables the system to dynamically adjust selection frequencies across rounds in response to changing computational and statistical conditions. The global score uses historical training time patterns to identify stragglers relative to their peers of other clients (4–5) and adjusts their influence in a manner that avoids permanent penalization. Our method defines the global score as in Eq. (8):

$$s_c = \alpha_1 A_s + \alpha_2 f_s \tag{8}$$

α_1 and α_2: a balance between fairness and reliability of clients, and it can be tuned by users.

This equation enables explicit tuning of selection priorities. During early training, a higher α_1 can prioritize fast and reliable clients to stabilize the initial model. As training progresses, increasing α_2: ensures that underrepresented clients receive more opportunities. This improves domain coverage and enhances the model's generalizability. In healthcare-oriented federated learning, this approach addresses two fundamental challenges. First, it preserves institutional equity while maintaining the unique data contributions of each participating hospital or medical center. Second, it ensures that resource-constrained institutions are not systematically excluded from collaboration, which helps retain the diversity and generalizability required for developing robust medical AI models.

As illustrated in Algorithm 2, the client selection mechanism proceeds through several phases designed to balance training efficiency, domain diversity, and participation fairness. The clustering algorithms depend on learned feature representations. However, in early rounds, randomly initialized models produce unreliable embeddings that do not reflect true data distributions. Therefore, we propose a warm-up period to allow the model to learn domain-specific features. Our approach clusters clients using learned feature representations rather than raw statistics. Domain heterogeneity is determined by the internal representations that models acquire rather than by the raw data itself. For this reason, most existing methods aim to learn domain-invariant feature representations to mitigate this issue [25]. During the warm-up phase, all clients are treated as a single cluster, and the global selection score operates over the entire client set. This phase stabilizes training and prioritizes fast, consistent clients to build a strong initial global model.

Algorithm 3: An efficient and fair global client selection

Inputs Clusters: set of clients in a cluster k, $t[c]$: time availability score of client c, T_{avg} : average training time of all clients, $t_{avail,c}$: training time of client c, T: total training rounds , α : EMA smoothing factor, β = reliability penalty parameters, Q: number of selected clients in each cluster, selection_counts: tracks selection frequency per client,t_hist: training history times of selected clients

1 For each cluster k in Clusters:

2 candidates ← Clusters[k]

3 For each client c in Clusters[k] do:

4 If c *not in* t_hist then:

5 t[c]=$t_{avail,c}$

6 Else:

7 $t[c] = \alpha t_avail + (1-\alpha)$ t[c]

8 $T_{max} = \frac{1}{N}\sum_{c=1}^{N} t[c]$

9 End

10 *compute* A_s with Equation (3)

11 If

12 $f_s[c] \leftarrow T / (T + selection_counts[c] \times n)$

13 Else

14 $f_s[c] = 1.0$

End

15 Compute global score score with (8)

16 participated_clients ← participated_clients ∪ {c}

17 selection_counts[c] ← selection_counts[c] + 1

18 End

For cluster in candidates do

19	candidates_sorted ← Sort (candidates, key= score, reverse=True)
20	selected_k ← candidates_sorted[0:Q]
21	Clusters_tos[k]= selected_k
22	End
23	Return Clusters_top

In later phases, we increase fairness to ensure that all clients can contribute. After the warm-up period, the algorithm enters the domain-aware phase. It first records baseline participation patterns (lines 8–10), then performs clustering to identify groups reflecting underlying data domains (lines 12–16). After the clustering phase, clients are organized into groups for selection (lines 19-27). Moreover, A challenge arises due to partial participation in federated learning. FedAvg aggregates updates only from selected clients, while non-participating clients do not contribute to model updates. As a result, the prototype collection is biased toward previously selected clients, and the EM clustering algorithm subsequently operates on a limited set of prototypes. This bias may cause cluster assignments to reflect historical selection patterns rather than true domain characteristics of all clients. To address this limitation, we introduce a virtual cluster appended to the EM-derived clusters. This additional cluster represents clients that have not yet participated in training, thereby preventing systematic exclusion and preserving domain diversity. we initialize the fairness score to its maximum value ($f_s = 1.0$) for clients that belong to the virtual cluster (lines 28-30). This initialization ensures that the global selection score considers these clients and ranks them alongside active participants. As a result, previously inactive clients receive an opportunity to contribute to training and become eligible for the domain clustering algorithm in the next rounds.

Algorithm 3 outlines the client selection procedure executed by the server during iteration t. The selection mechanism samples clients from both the EM algorithm and the virtual cluster (line1). This strategy allows the clustering process to access a broader range of client information, which helps preserve representativeness and reduce selection-induced bias. At each iteration t, participating clients report their training time $t_{avail,c}(i)$ to the server. Each client c updates its historical training time $T_c(i)$ using Eq. (4) (line 7). Once all client reports are received, the server calculates the average training time T_{max} (line 8). After completing clustering assignments using the EM algorithm for all clients, the server computes two scores, respectively: first, a reliability score (line 10) that evaluates temporal efficiency and a fairness score that measures participation equity (line 12). For previously selected clients, the fairness score is updated using Eq (8). For new clients, the fairness score is initialized to the maximum value ($f_s = 1.0$) to guarantee initial participation (lines 13). Finally, the server integrates these components to compute the global selection score s_c (line 15). This score provides a global ranking of clients, independent of their cluster assignment. The server enforces cluster-level selection. Within each cluster, clients are sorted by their global score, and the top-k clients are then selected. Thus, while the ranking is global, the final selection is performed within each cluster. This design balances fairness and efficiency globally while ensuring domain diversity. First, it reduces the chance that a single domain dominates selection simply because it contains many high-scoring clients. Second, it prevents the exclusion of minority domains and provides broader representation in training. Third, the approach maintains sensitivity to system heterogeneity, as slow clients are penalized regardless of their domain. This aligns with our motivation that GDCSF balances straggler mitigation, domain diversity, and fairness toward underrepresented clients.

IV. EXPERIMENT

A. Implementation Setup

We evaluate our method on the PathMNIST benchmark for multi-class medical image classification [33]. In a typical federated learning (FL) setting, we compare CDCSF with two standard FL baselines, FedAvg and Power-of-Choice (PoC), as well as with a centralized learning model. The evaluation is conducted under conditions of domain shift and the presence of straggling clients, which reflect realistic FL deployment scenarios. To emulate domain shift across clients, we partition the system into four domains representing distinct imaging conditions. Specifically, we split the PathMNIST training set into three separate domains, while the test set is assigned to an additional client, which serves as a fourth independent domain. Domain 0 corresponds to the original domain of the dataset. Domain 1 simulates mid-range equipment commonly found in resource-constrained settings and introduces a modest change in brightness (between −0.2 and +15%), reduced resolution, and contrast variations in the range [0.6, 1.4]. Domain 2 represents degraded imaging conditions, modeling more challenging real-world scenarios via increased brightness (+0.3), additive Gaussian noise ($\sigma = 0.12$), and contrast variations in [0.5, 1.5]. To additionally examine model behavior under system heterogeneity, we designate a subset of clients as stragglers throughout the entire training process. Whenever a straggler is selected for participation, we impose an additional delay sampled uniformly at random from the interval [10, 20] seconds for each local training phase. Although the added delay does not represent actual hospital latencies, it still produces the same relative effect. Fast clients proceed at normal speed, whereas stragglers require two to three times longer to complete their local updates. This controlled slowdown reflects the variations in hardware performance, workload, and network connectivity found across healthcare institutions, where some sites consistently operate more slowly than others. All experiments use a ResNet-18 backbone as the core feature extractor to ensure a fair comparison between all the methods.

The ResNet architecture was adapted for the PathMNIST dataset, which consists of 28×28 RGB images. To accommodate the smaller input size, the initial convolutional layer was modified to use a 3×3 kernel with stride 1 and 32 initial feature maps, followed by four residual blocks with progressively increasing channel dimensions (32, 64, 128, 256). Each block comprises two convolutional layers, batch normalization, and ReLU activation, followed by an adaptive average pooling layer and a fully connected classification head. The CDCSF model integrates this ResNet backbone within a federated learning framework, where the feature extractor is complemented by a projection module for prototype extraction and clustering. This

module consists of two linear layers with ReLU activation, reducing the feature dimensionality to 128 while preserving discriminative representations through normalization. The classification head is implemented as a single fully connected layer mapping the extracted features to the nine target classes in PathMNIST. Federated training was implemented with the Flower framework [34]. We conduct experiments with K = 10 and K = 20 clients. For K = 10, we sample 4 clients per round, and for K = 20, we sample 6 clients per round. Each selected client performs 3 local training epochs. Training runs for 200 global rounds until convergence. To ensure a fair comparison, all federated learning methods use the same hyperparameters, batch size of 32, and a learning rate of 0.01. All experiments are executed on two NVIDIA T4 GPUs.

B. Results and Discussions

We evaluate the proposed CDCSF framework under two key challenges in federated learning: (1) data heterogeneity and (2) system heterogeneity. We compare CDCSF with two FL baselines: FedAvg and Power-of-Choice (POW) and centralized learning. Table II reports the average test accuracy under domain shift and system heterogeneity. CDCSF achieves the highest final accuracy among all methods and surpasses FedAvg by 1.7 %. It also converges approximately two times faster. CDCSF shows strong effectiveness under a distribution shift. First, the domain-aware clustering groups clients according to their latent data distributions. This ensures that each round selects clients representing diverse domains within the global data distribution rather than sampling from randomly biased domain sources. Second, during the client selection, the global score assigns a higher historically underrepresented. This reduces early-round instability, especially under severe domain shift. CDCSF consistently surpasses Power-of-Choice (POC) with an improvement of more than 8% in final accuracy. This improvement is expected because POC evaluates clients only through their performances, which assigns higher weights to clients with higher loss and skews the optimization toward dominant domains. However, in domain-shifted medical data, this type of selection intensifies sampling bias, reduces the participation of minority domains, and increases the influence of the overrepresented clients. In contrast, CDCSF preserves domain diversity through prototype-based clustering and gives priority to reliable clients in early rounds. As a result, CDCSF selects a more informative and balanced client subset in each round. The results show that CDCSF fits heterogeneous medical settings more effectively, where strong performance requires both computational efficiency and balanced domain representation.

TABLE II. ACCURACY IMPROVEMENT OF CDCSF COMPARED WITH FEDAVG, POC, AND CENTRALIZED LEARNING UNDER DOMAIN SHIFT

Algorithm	Accuracy
FedAvg	82.02%
PoC	66.37%
CDCSF	83.78%
Centralized Learning	86.23%

To demonstrate the benefits of our framework under domain shift, we present the validation accuracy of all FL methods in Fig. 2. For clarity, note that the validation accuracy reported for each method is computed as the mean validation accuracy of all clients. CDCSF achieves faster convergence in early rounds compared to FedAvg and Power-of-Choice (POC). Our method reaches 80% accuracy within 75 rounds, two times faster compared to FedAvg, while POC fails to attain this accuracy. During the early iterations, when the model begins to converge, CDCSF selects clients from diverse domains. This ensures that heterogeneous data sources contribute to the global model. Hence improves generalization performance. In addition, the algorithm initially prioritizes fast and reliable clients to accelerate convergence. As training progresses, we increase the weight of the fairness score to enable slower or previously underrepresented clients to join the training process. Further, we highlight the effect of distribution shift on the compared client selection frameworks. Domain heterogeneity generally reduces performance, but its impact on FedAvg remains comparatively modest, as shown in Table II and Fig. 2. This behavior aligns with our motivation because FedAvg selects clients uniformly at random, which ensures that all clients from different domains participate with uniform equal frequency. Under domain shift, such uniform selection benefits the global model. As a result, exposure to all domains enables the model to learn efficient domain-invariant representations. Consequently, we also observe that FedAvg converges faster than (POC), which suffers from a selection bias that oversamples certain domains while underrepresenting others. This bias degrades the model's performance by repeatedly prioritizing clients with temporarily higher losses. These observations show that distribution shift harms client-selection algorithms that rely primarily on resource availability or short-term statistical performance.

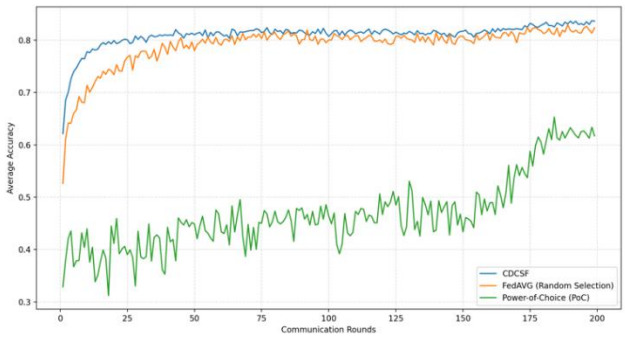


Fig. 2. Average validation accuracy of CFDSC, FedAvg and PoC across communication rounds.

As illustrated in Fig. 3, FedAvg selects clients at similar frequencies, but CDCSF produces a more balanced and adaptive pattern that reflects both efficiency and domain diversity. Although uniform sampling guarantees equal selection frequency in FedAvg, this property is insufficient in healthcare FL, where clients suffer from a domain shift issue. This sampling neglects the existing domain imbalances, enforcing equal participation frequency across clients without addressing underlying distribution disparities. CDCSF addresses this limitation through a domain-aware clustering mechanism that groups clients using privacy-preserving feature prototypes. The server applies an EM algorithm to identify heterogeneous clinical domains by clustering clients according to their underlying data distributions. As a result, even clinically

important domains with only a few clients are placed into their own clusters rather than being merged with the majority of domains. Moreover, the incorporation of EMA reliability scoring mitigates the impact of stragglers without permanently excluding slower institutions, while the fairness score ensures equitable participation for historically underrepresented clients. The global scoring mechanism then selects the top-k clients from each cluster and ensures consistent representation of minority domains throughout training. These components form a coherent and robust improvement over random selection. The method increases representativeness, reduces domain bias, and supports more efficient convergence in federated healthcare settings.

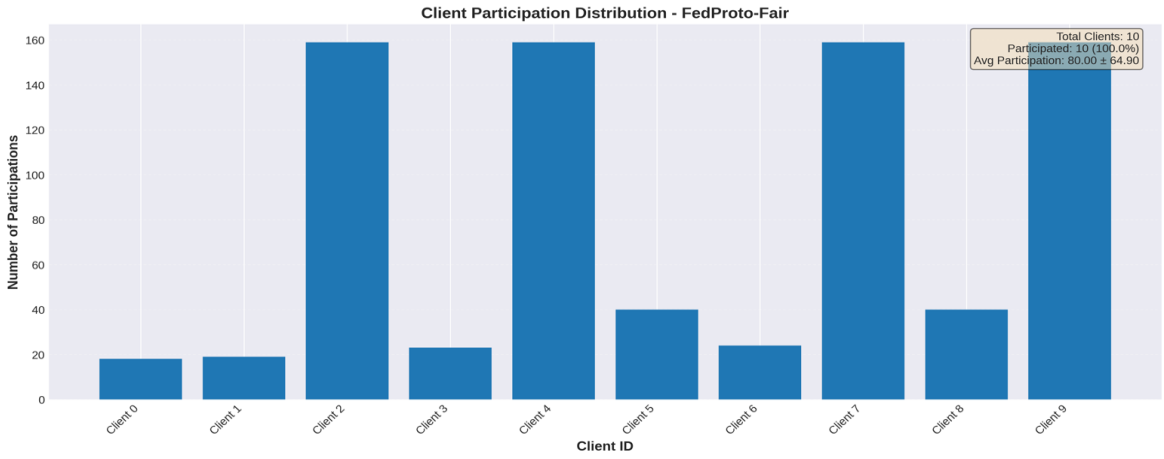


Fig. 3. Comparison of client participation frequencies across methods.

To model system heterogeneity, we define stragglers as clients who incur additional delay during local training. In our simulation, we assign this role to clients 0 and 1, while all other clients operate normally.

The server computes an exponential moving average (EMA) of each client's training time and computes a global reference value T_{max} at every round, obtained as the mean EMA across all clients. Fig. 4 displays each client's EMA estimate T_c together with the global reference T_{max}.

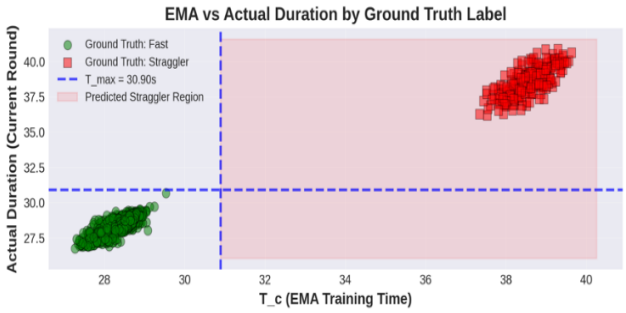


Fig. 4. The detection of stragglers' performance using EMA reliability scoring.

The straggler clients, shown in red, fall in the 37–40 second range, whereas fast clients, shown in green, cluster around 28–30 seconds. The two groups remain well separated. The vertical dashed line at $T_{max} = 30.9$ Seconds highlights this separation. The shaded region corresponds to the decision boundary of our reliability score. A client falls inside this region when its score $s = 1 - A_s$ exceeds the threshold θ. We use this score to predict whether a client is a straggler or not. The visualization confirms that our detection rule remains stable across all 200 rounds, despite the random injection of extra delay.

To further examine how the reliability score prioritizes fast clients, we train the model with k=20 clients and visualize the relationship between each client's reliability score and its EMA training time. Fig. 5 illustrates how training duration affects the reliability metric. Clients with larger T_c values (e.g., Clients 0 and 1) appear in the lower right region of the plot and receive lower reliability scores, which reflects the inverse relationship between training time and reliability. In contrast, faster clients form a compact cluster in the upper left region, indicating consistently higher reliability. This visualization confirms that the scoring mechanism effectively separates slow and fast clients and demonstrates that faster clients get higher prioritization weights. Importantly, the purpose of the reliability metric is not to penalize or exclude stragglers. Instead, it assigns them appropriately reduced weights while preserving their participation. Their inclusion is reinforced through the fairness component, which ensures equitable representation so that the global model still benefits from their unique data characteristics.

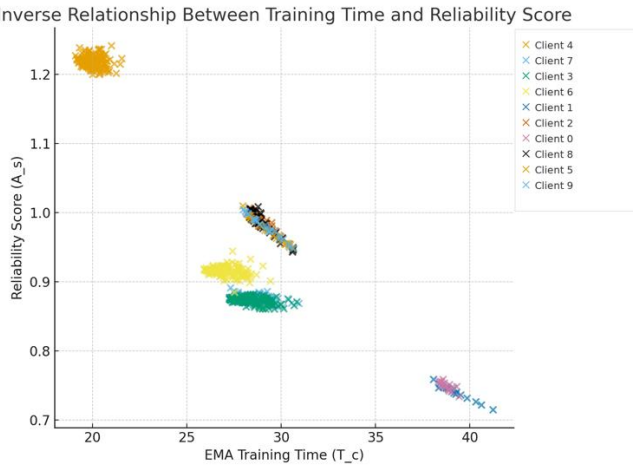


Fig. 5. Relationship between EMA training time and reliability score.

To further evaluate the impact of data heterogeneity, we partition the dataset across clients using a Dirichlet distribution with concentration parameter α=0.5, which creates a highly skewed non-IID distribution. Fig. 6 visualizes the distribution of each class across clients for all domains.

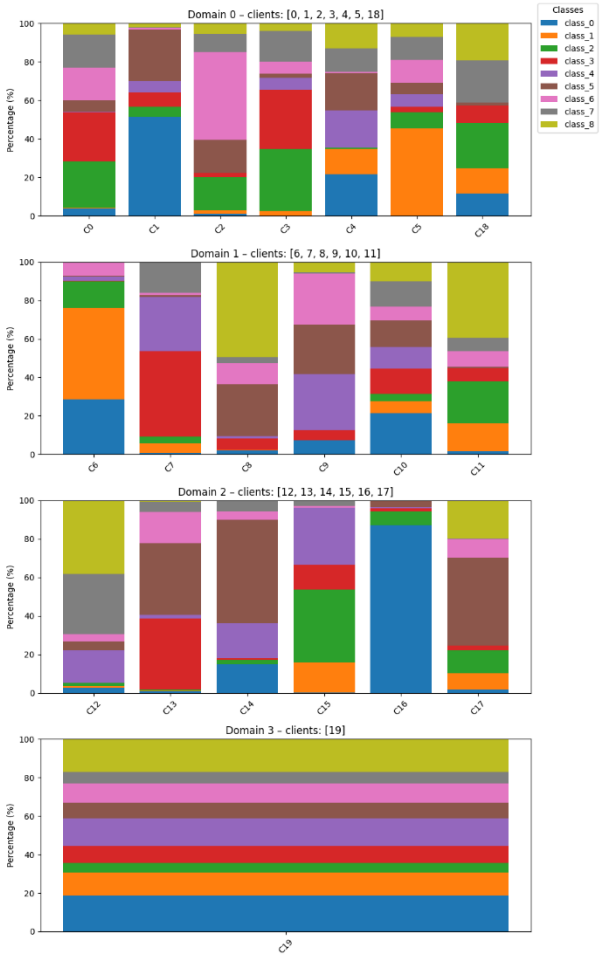


Fig. 6. Visualization of non-iid client distributions using Dirichlet sampling under domain shift when ($\alpha = 0.5$).

In addition, we apply a severe domain shift to each domain to increase the difficulty compared to our initial simulation. In this experiment, only 6 out of 20 clients are selected per round. In practical FL deployments, data heterogeneity is the norm. Therefore, this analysis allows us to evaluate how CDCSF simultaneously addresses both data and resource heterogeneity. In this setting, we test multiple scheduling policies from the CDCSF framework. Each policy selects clients based on global scores that combine reliability and fairness weights. Each configuration produces a distinct CDCSF graph. The reason for that is to show different trade-offs between efficiency and participation. We compare these policies with the FedAvg baseline. We refer to this algorithm as vanilla. As expected, the overall accuracy in Fig. 7 is lower compared to the results shown in Fig. 2. This behavior is expected because the data distribution in this scenario jointly combines feature skew and label imbalance, both of which are known to disrupt the performance of federated optimization. Our method does not attempt to correct the data distribution issue at its source. However, we observe that CDCSF (hybrid) shows a stable and consistent learning trajectory. In contrast, FedAvg demonstrates pronounced fluctuations characterized by inconsistency across rounds, which indicates its sensitivity to non-IID distribution. These results demonstrated the role of CDCSF in mitigating the effects of data and resource heterogeneity. This improves the convergence rate even when both conditions coexist. The results in Fig. 7 further illustrate the impact of our three selection strategies. CDCSF–fast prioritizes clients with high reliability scores and favors devices that have historically shown faster performance. CDCSF–equal assigns equal weights to fairness and reliability across all rounds. CDCSF– hybrid adopts a dynamic policy that emphasizes reliability in the early rounds and increases fairness weight in later rounds to incorporate underrepresented clients more effectively. This adaptive behavior allows CDCSF–hybrid to maintain efficient training while progressively enhancing participation diversity. During the first 20 rounds, which serve as the warm-up phase, we do not employ the EM domain clustering. All methods behave similarly and show nearly identical performance. As expected, this confirms that, in the absence of clustering, the client selection process behaves like random sampling. In this early stage, training time does not provide meaningful information for distinguishing reliable from non-reliable clients, so the global selection mechanism does not contribute to performance improvement. Once the warm-up phase finishes and EM clustering becomes active, the performance of the methods begins to diverge. We observe that CDCSF–hybrid achieves the best convergence and outperforms all other policies. Its design prioritizes reliable clients in the early iterations, which stabilizes learning and refines the local feature-prototype representations that guide domain assignment in the clustering. As training progresses, we increase the fairness weight so that unrepresented clients are then incorporated in domain coverage without disrupting optimization. Introducing fairness in the mid-stage is essential because it allows underrepresented clients to enter the global score ranking and ensures their participation from the mid to late rounds. This gives these clients sufficient opportunity to contribute their updates and influence the global model. This adaptive balance between early reliability and later fairness enables CDCSF–hybrid to achieve both stable convergence and improved generalization across heterogeneous domains. The behavior observed in the fast and equal policies further supports this interpretation. The fast policy outperforms the equal policy because, under imbalanced data distributions, FL algorithms naturally benefit from consistently available participants who contribute updates regularly. Similarly, FedAvg, which relies on random selection, outperformed the equal policy, which confirms that participation consistency plays a central role in FL performance. However, relying on consistent clients leads to biased learning. CDCSF–hybrid overcomes this limitation by ensuring a balance between high-participation clients and those underrepresented in the federation. This balance produces more stable and effective training and enables higher performance in challenging settings characterized by both data and resource heterogeneity. Resource heterogeneity often increases the number of underrepresented clients because many client selection strategies prioritize fast and consistently available devices while excluding clients with constrained computational resources from the training process. Although prioritizing such

clients may offer advantages in iid settings, it becomes a challenge under non-IID distributions. In healthcare, these slower clients frequently hold rare classes or essential domain information. Excluding the theme removes critical data from the learning process and degrades the model performance. As a result, when data heterogeneity and resource constraints coexist, traditional client selection methods fail to maintain domain diversity and struggle to achieve robust performance in federated learning.

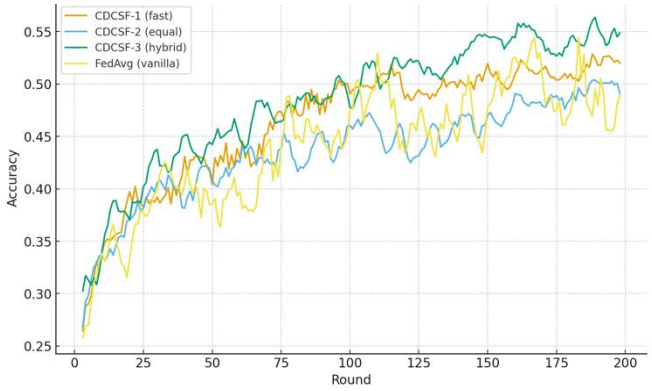


Fig. 7. Comparison of client selection policies on PathMNIST under severe domain shift and non-iid data heterogeneity.

V. Conclusion

In this work, we introduced CDCSF, a domain-aware client selection framework that combines prototype-based clustering, reliability estimation, and fairness participation to guide dynamic selection during training. The framework aims to preserve domain diversity, reduce sampling bias, and ensure that underrepresented clients contribute to training despite their computational capacity. In healthcare settings, data are often limited and heterogeneous. Effective client selection is therefore essential to preserve diversity, improve representativeness, and enhance model robustness. Despite these advantages, CDCSF remains sensitive to clustering hyperparameters, including the number of clusters and update frequency. Inappropriate settings may disrupt the EM optimization process, which may slow model convergence. Furthermore, although prototype-based clustering preserves privacy, it may introduce additional communication overhead. In addition, CDCSF adjusts client weights to mitigate straggler effects. However, extreme system heterogeneity can still delay aggregation and slow training. Overall, CDCSF demonstrates that domain-aware client selection can simultaneously address data heterogeneity and resource constraints in federated healthcare settings. Extending the framework to real-world multi-institutional deployments and validating it under more real-world datasets represent the most critical directions for future research.

V. Conflict of Interest

The author declares that there are no conflicts of interest related to this research.

References

[1] B. Camajori Tedeschini et al., "Decentralized Federated Learning for Healthcare Networks: A Case Study on Tumor Segmentation," IEEE Access, vol. 10, pp. 8693–8708, 2022, doi: 10.1109/ACCESS.2022.3141913.

[2] M. J. Sheller et al., "Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data," Sci. Rep, vol. 10, no. 1, p. 12598, July 2020, doi: 10.1038/s41598-020-69250-1.

[3] W. Zhang et al., "Dynamic Fusion based Federated Learning for COVID-19 Detection," 2020, arXiv. doi: 10.48550/ARXIV.2009.10401.

[4] H. R. Roth et al., "Federated Learning for Breast Density Classification: A Real-World Implementation," in Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning, vol. 12444, S. Albarqouni, S. Bakas, K. Kamnitsas, M. J. Cardoso, B. Landman, W. Li, F. Milletari, N. Rieke, H. Roth, D. Xu, and Z. Xu, Eds., in Lecture Notes in Computer Science, vol. 12444. , Cham: Springer International Publishing, 2020, pp. 181–191. doi: 10.1007/978-3-030-60548-3_18.

[5] L. Li, N. Xie, and S. Yuan, "A Federated Learning Framework for Breast Cancer Histopathological Image Classification," Electronics, vol. 11, no. 22, p. 3767, Nov. 2022, doi: 10.3390/electronics11223767.

[6] J. Wang et al., "A Field Guide to Federated Optimization," 2021, arXiv. doi: 10.48550/ARXIV.2107.06917.

[7] L. Fu, H. Zhang, G. Gao, M. Zhang, and X. Liu, "Client Selection in Federated Learning: Principles, Challenges, and Opportunities," 2022, arXiv. doi: 10.48550/ARXIV.2211.01549.

[8] Y. J. Cho, J. Wang, and G. Joshi, "Client Selection in Federated Learning: Convergence Analysis and Power-of-Choice Selection Strategies," 2020, arXiv. doi: 10.48550/ARXIV.2010.01243.

[9] W. Marfo, D. K. Tosh, and S. V. Moore, "Efficient Client Selection in Federated Learning," 2025, arXiv. doi: 10.48550/ARXIV.2502.00036.

[10] M. Ribero and H. Vikalo, "Communication-Efficient Federated Learning via Optimal Client Sampling," 2020, arXiv. doi: 10.48550/ARXIV.2007.15197.

[11] R. Yan et al., "Label-Efficient Self-Supervised Federated Learning for Tackling Data Heterogeneity in Medical Imaging," IEEE Trans. Med. Imaging, vol. 42, no. 7, pp. 1932–1943, July 2023, doi: 10.1109/TMI.2022.3233574.

[12] Y. Sun, N. Chong, and H. Ochiai, "Feature Distribution Matching for Federated Domain Generalization," 2022, arXiv. doi: 10.48550/ARXIV.2203.11635.

[13] J. Chen, M. Jiang, Q. Dou, and Q. Chen, "Federated Domain Generalization for Image Recognition via Cross-Client Style Transfer," 2022, arXiv. doi: 10.48550/ARXIV.2210.00912.

[14] G. Wu and S. Gong, "Collaborative Optimization and Aggregation for Decentralized Domain Generalization and Adaptation," in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada: IEEE, Oct. 2021, pp. 6464–6473. doi: 10.1109/ICCV48922.2021.00642.

[15] W. Zhang and X. Li, "Federated Transfer Learning for Intelligent Fault Diagnostics Using Deep Adversarial Networks With Data Privacy," IEEE/ASME Trans. Mechatron., vol. 27, no. 1, pp. 430–439, Feb. 2022, doi: 10.1109/TMECH.2021.3065522.

[16] Y. Tan et al., "FedProto: Federated Prototype Learning across Heterogeneous Clients," 2021, arXiv. doi: 10.48550/ARXIV.2105.00243.

[17] Y. Qiao, H. Q. Le, M. Zhang, A. Adhikary, C. Zhang, and C. S. Hong, "FedCCL: Federated dual-clustered feature contrast under domain heterogeneity," Information Fusion, vol. 113, p. 102645, Jan. 2025, doi: 10.1016/j.inffus.2024.102645.

[18] U. Michieli and M. Ozay, "Prototype Guided Federated Learning of Visual Feature Representations," 2021, arXiv. doi: 10.48550/ARXIV.2105.08982.

[19] X. Mu et al., "FedProc: Prototypical Contrastive Federated Learning on Non-IID data," 2021, arXiv. doi: 10.48550/ARXIV.2109.12273.

[20] F. Sattler, K.-R. Müller, and W. Samek, "Clustered Federated Learning: Model-Agnostic Distributed Multi-Task Optimization under Privacy Constraints," 2019, arXiv. doi: 10.48550/ARXIV.1910.01991.

[21] Y. Zhang, H. Chen, Z. Lin, Z. Chen, and J. Zhao, "FedAC: An Adaptive Clustered Federated Learning Framework for Heterogeneous Data," 2024, arXiv. doi: 10.48550/ARXIV.2403.16460.

[22] G. Long, M. Xie, T. Shen, T. Zhou, X. Wang, and J. Jiang, "Multi-center federated learning: clients clustering for better personalization," World

Wide Web, vol. 26, no. 1, pp. 481–500, Jan. 2023, doi: 10.1007/s11280-022-01046-x.

[23] M. Duan et al., "Flexible Clustered Federated Learning for Client-Level Data Distribution Shift," IEEE Trans. Parallel Distrib. Syst., pp. 1–1, 2021, doi: 10.1109/TPDS.2021.3134263.

[24] T. Nishio and R. Yonetani, "Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge," in ICC 2019 - 2019 IEEE International Conference on Communications (ICC), Shanghai, China: IEEE, May 2019, pp. 1–7. doi: 10.1109/ICC.2019.8761315.

[25] Z. Sasindran, H. Yelchuri, and T. V. Prabhakar, "Ed-Fed: A generic federated learning framework with resource-aware client selection for edge devices," in 2023 International Joint Conference on Neural Networks (IJCNN), Gold Coast, Australia: IEEE, June 2023, pp. 1–8. doi: 10.1109/IJCNN54540.2023.10191316.

[26] M. Cao, Y. Zhang, Z. Ma, and M. Zhao, "C 2 S : Class-aware client selection for effective aggregation in federated learning," High-Confidence Computing, vol. 2, no. 3, p. 100068, Sept. 2022, doi: 10.1016/j.hcc.2022.100068.

[27] F. Lai, X. Zhu, H. V. Madhyastha, and M. Chowdhury, "Oort: Efficient Federated Learning via Guided Participant Selection," in 15th USENIX Symposium on Operating Systems Design and Implementation (OSDI 21), USENIX Association, July 2021, pp. 19–35. [Online]. Available: https://www.usenix.org/conference/osdi21/presentation/lai

[28] C. Li, X. Zeng, M. Zhang, and Z. Cao, "PyramidFL: a fine-grained client selection framework for efficient federated learning," in Proceedings of the 28th Annual International Conference on Mobile Computing And Networking, Sydney NSW Australia: ACM, Oct. 2022, pp. 158–171. doi: 10.1145/3495243.3517017.

[29] T. Huang, W. Lin, W. Wu, L. He, K. Li, and A. Y. Zomaya, "An Efficiency-Boosting Client Selection Scheme for Federated Learning With Fairness Guarantee," IEEE Trans. Parallel Distrib. Syst., vol. 32, no. 7, pp. 1552–1564, July 2021, doi: 10.1109/TPDS.2020.3040887.

[30] Z. Chai et al., "TiFL: A Tier-based Federated Learning System," 2020, arXiv. doi: 10.48550/ARXIV.2001.09249.

[31] W. Huang, M. Ye, Z. Shi, H. Li, and B. Du, "Rethinking Federated Learning with Domain Shift: A Prototype View," in 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada: IEEE, June 2023, pp. 16312–16322. doi: 10.1109/CVPR52729.2023.01565.

[32] B. Hitaj, G. Ateniese, and F. Perez-Cruz, "Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning," 2017, arXiv. doi: 10.48550/ARXIV.1702.07464.

[33] J. Yang *et al.*, "MedMNIST v2 - A large-scale lightweight benchmark for 2D and 3D biomedical image classification," *Sci Data*, vol. 10, no. 1, p. 41, Jan. 2023, doi: 10.1038/s41597-022-01721-8.

[34] Beutel DJ, Topal T, Mathur A, Qiu X, Fernandez-Marques J, Gao Y, et al. Flower: a friendly federated learning research framework. arXiv:2007.14390. 2020. doi:10.48550/ARXIV.2007.14390.

Temporal Attention Networks for Real-Time Multimodal Emotion Recognition from EEG and fNIRS Signals

Vinod Waiker[1], Anne Marie D. Pahiwon[2], Ankush Mehta[3], Gadug Sudhamsu[4],
Dr Pavithra M[5], Dr. K. Kiran Kumar[6], B Kiran Bala[7], Osama R.Shahin[8]
Datta Meghe Institute of Management Studies, Nagpur, Maharashtra, India 440020[1]
Ifugao State University - Philippines[2]
Faculty of Engineering & Technology-Department of Mechanical Engineering-Marwadi University Research Center, Marwadi University, Rajkot, 360003, Gujarat, India[3]
Department of Computer Science and Engineering-School of Engineering and Technology, JAIN (Deemed to be University), Bangalore, Karnataka, India[4]
Associate Professor & Head-Department of Computer Science and Business Systems, Jansons Institute of Technology Karumathampatti, Coimbatore, India[5]
Professor, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, AP, India[6]
Department of AI & DS, K.Ramakrishnan College of Engineering, Trichy, India[7]
Department of Computer Science-College of Computer and Information Sciences, Jouf University, Saudi Arabia[8]
Faculty of Engineering-Physics and Mathematics Department, Helwan University, Helwan, Egypt[8]

***Abstract*—Emotion recognition is critical in the development of real-time mental health care and individualized cognitive behavior. Current strategies to recognize cognitive emotions frequently fail to capture complex time dependencies and multimodal physiological reactions, leading to sub-optimal performance and inaccurate generalization. To overcome such shortcomings, the proposed study suggests TCADNet, a new deep learning model that integrates Temporal Convolutional Networks (TCN), attention-based feature weighting, and GAN-based data augmentation to achieve a high recognition rate of the emotional states through EEG and fNIRS recordings. The model utilizes the TCNs to extract both short-term and long-term temporal trends, and the attention mechanism emphasizes salient parts that bring about emotions, which improves interpretability. Moreover, a Deep Convolutional GAN creates artificial signals of unrepresented emotion classes, eliminating data imbalance and enhancing generalization. The TCADNet model is coded in Python on the TensorFlow/Keras system, and its key components are preprocessing, time modeling, attention weighting, data augmentation, and last classification by SoftMax layers. Experimental outcomes indicate that TCADNet has high recognition performance, with overall recognition, accuracy, precision, and recall, and F1-scores of over 98, which is higher than conventional CNN, LSTM, and separate TCN models. The suggested methodology can be useful to researchers, clinicians, and mental health professionals as it allows them to monitor cognitive and emotional conditions in real-time with a reliable, decipherable, and scalable instrument and provides an opportunity to detect and respond to the issue promptly and implement a tailored intervention plan in educational or health-related settings.**

***Keywords*—Emotion recognition; Temporal Convolutional Network; attention mechanism; mental healthcare; EEG**

I. Introduction

The rising cases of mental health problems have led to the invention of intelligent, and autonomous systems that are able to identify emotional states with high precision and confidence. According to the estimates done by the world health organization (2024), over 280 million individuals around the world are affected by depression and stress-related disease and emotional dysregulation can frequently precede clinical manifestations [1]. Electroencephalography (EEG) is a non-invasive neurophysiological method that obtains a record of electrical activity in the brain by electrodes on the scalp and the frequency content is measured in microvolts (uV) at frequencies between 1100 Hz [2]. EEG offers a very high temporal resolution of milliseconds, which can accurately trace fast neural changes related to emotional and thinking processes. Conversely, functional near-infrared spectroscopy (fNIRS) is an optical neuroimaging modality technique that monitors effects of hemodynamic changes by measuring alterations of oxygenated (HbO) and deoxygenated hemoglobin (HbR) in the bloodstream by using near-infrared light sources with sampling rates around 7-10 Hz. Although EEG records direct electrical activity of neurons with a high level of temporal resolution, fNIRS is more effective in providing a localization of cortical blood flow in terms of the vascular response [3],[4]. The combination of the fast temporal sensitivity of EEG and the spatial specificity of fNIRS can be used to further describe emotion-related brain activity in a more comprehensive manner.

The trends in artificial intelligence (AI) recently have demonstrated that, when given raw physiological signals, deep learning models can learn discriminative features. Convolutional Neural Networks (CNNs) and Long Short-term

Memory (LSTM) networks have demonstrated good results in terms of identifying emotional patterns given time-series data [5]. Nevertheless, these architectures have some serious drawbacks, including extreme sensitivity to noise [6], inability to deal with long distance temporal dependencies and uninterpretability [7]. The cognitive frameworks that make up multimodal interaction are stipulated based on the occurrence of a particular modality-verbal, bodily, and visual-is mode of interaction, the presence of a grammatical structure-syntax, and narrative as well as whether they are predominant throughout the whole semantics of the expression [8]. Multimodal sentiment analysis is the most interesting study area of AI across the disciplines. The current classification task of human multimodal sentiment analysis is a highly finer task compared to the previous classification tasks. MSA has made significant contributions to human affection polarity detection over the past years since the advent of DL and the benefit of neural networks [9]. Multimodal DL consists of a number of challenges. One of the critical issues is the conversion of the information in multimodal form to the machine readable one [10]. AI solutions have a high potential of enhancing mental health. However, which approach should be selected is an issue of main concern [11]. Depression and stress are highly prevalent mental disorders that permeate the whole globe to the extent of reaching everyone in the society regardless of their age, gender and social economic statuses [12].

With such gaps in mind, the emerging concern of Temporal Convolutional Networks (TCNs), attention-based mechanisms and generative adversarial networks (GANs) has been in the past due to its capability in capturing and addressing temporal dependence, improving interpretability and reducing data imbalance, respectively. However, the existing methods seldom use all three mechanisms into a single model of cognitive emotion recognition basing on multimodal cues such as EEG and fNIRS [13]. In this case, assessment of any unfavorable psychological characteristic such as stress, anxiety and sadness all hinge on the detection of emotion. The second source of knowledge is the application of a new combination of an electroencephalogram (EEG) data [14]. The smart use of technologies has transformed the healthcare industry in a monumental manner. The focus of all the research is on the following critical aspects: medical imaging and diagnosis, virtual patient treatment, medical research and drug discovery, patient engagement, compliance, rehabilitation, and other administrative applications. It is based on the general picture that illustrates the application of AI in the area of medicine [15]. The advent of the artificial intelligence technologies has enhanced the efficiency and reasonability of the data collection process that has tremendous influence on the field of illness management and prevention [16]. The rising rate of mental diseases among the youths globally is one of the biggest problems in this society [17]. The issues of mental diseases are extremely high in the youth population; according to the existing estimates, 20 per cent of children and adolescents in the world are mentally ill [18]. The mental health care services is also needed today due to the social change, rising demand of optimal healthcare and flexibility of the mental illness. The conventional channels of accessing mental health services are pushed to the brink by the lack of resources and long queues in the process of accessing the services [19]. Sentiment analysis and emotion detection are vital in emotive management of patients in the healthcare [20].

A. Research Motivation

The growing rates of cognitive stress, anxiety, and dysregulation of emotions in students demonstrate the urgency of objective and real-time emotion recognition tools. Conventional methods of assessment mostly use self-administered questionnaires and behavioral observations, which are subjective and cannot detect fast neurophysiological variation. Electroencephalography (EEG) and functional near-infrared spectroscopy (fNIRS), on the contrary, offer a complementary neural measure in electrical brain activity and hemodynamic responses, respectively. Nevertheless, such multimodal time-series signals are difficult to model, as they are noise sensitive, their temporal misalignment, and multi-complex long-range correlations. Traditional deep learning algorithms like CNNs and LSTMs tend to violate the causality of the time, interpretability, and address data imbalances in affective data. These constraints inspire this study to present a unified deep learning model consisting of temporal convolutional network, attention, and generative adversarial augmentation to achieve robust, interpretive, and scalable multimodal cognitive emotion recognition of student mental health in monitoring.

B. Research Significance

This study makes contributions to multimodal cognitive emotion recognition as it provides a single deep learning framework specific to EEG-fNIRS signal fusion. The model is used to learn temporal dependencies on long distances by exploiting temporal convolutional networks (TCNs) without violating causal order and permitting real-time use. By using a temporal attention system, interpretability is improved by detecting time segments of emotional salience in neural recording [21], [7]. Moreover, GAN-based augmentation decreases the class imbalance and enhances the performance of generalization in emotion datasets. The proposed framework takes advantage of the complementary nature of the high temporal resolution of EEG and the spatial resolution of fNIRS, unlike the more conventional single-modality methods, which lead to a higher strength and discrimination ability. The system facilitates scaled real-time use which will be useful in the mental health assessment, academic stress monitoring and adaptive learning settings. In general, the structure presents a non-invasive, objective, and data-driven alternative to traditional psychological assessment tools, allowing to detect and proactively manage the emotional-related cognitive conditions early.

The contribution of the research are as follows:

- Developed a Temporal Convolutional Attention-Driven Network (TCADNet) for multimodal cognitive emotion recognition, capable of modeling complex temporal dependencies in synchronized EEG and fNIRS neurophysiological signals.
- Employed a Temporal Convolutional Network with causal and dilated convolutions integrated with an attention mechanism to identify emotionally significant time segments and improve interpretability.

- Utilized the publicly available REFED multimodal EEG–fNIRS dataset to validate the effectiveness of the proposed framework under realistic emotion elicitation conditions, including synchronized signal preprocessing and temporal window segmentation.
- Achieved high classification accuracy and improved generalization performance compared to conventional deep learning approaches, demonstrating the robustness and scalability of the proposed framework.

C. Problem Statement

Neurophysiological signals are complex, non-linear, and time-varying, which makes it a challenging undertaking to identify cognitive emotions pertaining to stress accurately. Multimodal signals, including electroencephalography (EEG) and functional near-infrared spectroscopy (fNIRS), can give complementary data regarding the neural electrical activity and hemodynamic responses [22],[23], but they are not easily described in terms of their temporal dynamics or cross-modal interactions. The presence of long-term temporal dependencies and variations in the context of emotion states in conventional machine learning models has led to the frequent reliance on hand-crafted features in traditional analytical methods and machine learning models [18]. Recurrent neural networks and conventional convolutional architectures that have been used in emotion recognition do have weaknesses, including the disappearance of gradients, excessively high computational expense, low levels of interpretability, and the inability to conduct multimodal synchronization. Moreover, in many cases, emotion datasets are also unbalanced in classes, and this results in biased learning and lower generalization scores. The absence of mechanisms to detect emotionally salient temporal pieces also limits the reliability of the models [21]. The constraints prevent the creation of effective, real-time multimodal emotion recognition in student mental health monitoring. Consequently, there exists an acute necessity of scalable and understandable deep learning structure that can imitate multi-scale time-related characteristics, deal with the issue of class imbalance, and utilize EEG-fNIRS data in a manner that is effective in providing cognition emotion recognition.

The study is organized as follows: Section I presents the motivation, challenges, and goals of AI-based multimodal emotion recognition. Section II overviews related research. Section III explains the proposed self-supervised temporal learning framework, multimodal data fusion, and experiment setup. Section IV shows results and analysis. Section V concludes and summarizes future work.

II. Related Works

Kargarandehkordi et al. [21] proposed to critically evaluate the use of AI models for forecasting, tracking mental health states and symptoms using wearable biosensor data. Following PRISMA criteria, the authors conducted a systematic analysis of 48 studies using a range of sensors, including location, accelerometry, heart rate, heart rate variability, electrodermal activity, audio, and smartphone use data. These sensors provide continuous, remote, and objective mental state monitoring and are integrated into wearable technology like fitness bands and smartwatches. The integration of AI, especially ML, makes it possible to detect mild physiological, behavioral signals and signs related to mental health.

Dritsas et al. [7] proposed recent developments in methods and adaptive system architecture for multimodal systems. Though specific datasets do not directly find use in the study, systems by modalities that usually utilize sensor-input datasets like motion capture, speech recordings, eye-tracking, and physiological signals are also analyzed in the study. Enhanced usability, natural interaction, context-sensitive flexibility, and portability across a wide range of applications, such as healthcare, are some of the most significant advantages. Difficulties, however, are in terms of complex input synchronizing, conflict of fusion, environmental flexibility and lack of standardized structures or frames of reference to actual deployments.

Avital et al. [2] proposed to create a real-time system for assessing lecturers' teaching quality by observing students' facial expressions to identify emotional states while listening to lectures. The procedure consists of four phases. They are: image capture, preprocessing, emotion recognition and identification. The data was obtained experimentally from a class of 45 students recording live facial expressions during lectures. The major strengths are instant feedback regarding student engagement, high accuracy (83%) in emotion recognition and high correlation (91.7%) with conventional feedback methods. Further, the suggested application of optical neural networks suggests quicker and more efficient processing than electronic systems. However, some limitations are moderate prediction accuracy of student comprehension (43–62.94%), possible privacy issues, reliance on steady lighting and camera position for proper facial detection. Nassiri and Akhloufi [3] suggest the use of large language models like GPT, Bloom and LLaMA in the healthcare industry for enhancing patient care, medical research and diagnostic assistance. The objective is to evaluate the ability of LLMs to read, create medical text and approximate their usability in actual clinical practice. The approaches encompass a thorough review and examination of the architectures of top LLMs, their training, and their performance on medical tasks. The study investigates datasets for training these models, which are classified by size, source and medical topics. Among the strengths, LLMs have tremendous potential to aid healthcare professionals in documentation, literature review and decision-making, as well as potentially enhance the efficiency of the healthcare system through automation.

X. Chen et al. [22] proposed to investigate the evolution and trend of AI-based multimodal data analysis in smart healthcare using large-scale literature research. In order to make this a reality, authors used bibliometric analysis and topic modeling techniques in order to explore 683 scholarly articles across 2002 and 2022. The data of analysis were procured from research databases with specific focus on AI, healthcare, and multimodal integration of data. The stark strength of the work lies in its capability of uncovering avenues of research that will unfold, leading individuals and topics to follow such as GANs, contrastive learning and hybrid neuroimaging approaches. It has worldwide emphasis with special regard being provided to contributions from nations such as the USA and India. While, there are certain limitations like possible bias because of data selection (language or database constraint) or grey literature

exclusion and article metadata use instead of experimental or clinical data. Despite this, the study provides valuable points for future interdisciplinary research in AI-based smart healthcare.

Lee et al. [23] has been suggested to study the integration of Artificial Intelligence into the field of biomedical signal analysis to enhance the effectiveness and accuracy of diagnosis in healthcare. It dwells upon the most prominent AI approaches that are controlled, uncontrolled, reinforcement learning and how it can be applied to interpret bio signals like EEG and ECG. The methods involve the analysis of various AI architectures and learning paradigms applied in the classification of physiological signals, action and abnormalities, and decision-making activities. Despite this, the review lacks dataset specificity. It is founded on actual biomedical signals, such as ECG and EEG, in different clinical settings. Among the key advantages, one can distinguish the increased speed and accuracy of the diagnostic process, reducing the amount of work that is to be carried out manually and increased access to health services in resource-deficient areas and better response to emergencies through the rapid analysis. Yet, constraints include data privacy issues, interpretability of the model, dependence on high-quality labeled datasets and real-time integration difficulties in clinical settings. Payandeh et al. [4] proposed an overview of deep representation learning methods, pointing out their application to learn meaningful, low-dimensional features from high-dimensional data for tasks like classification, prediction and segmentation. While the review does not concentrate on particular datasets, it mentions varied types of data such as images, text, sequential data and multimodal inputs. The benefits of such methods are to learn generalizable and transferable features automatically without handcrafted feature engineering, thus enhancing learning robustness and model generalizability. Although much headway has been achieved, limitations persist in the ability to learn interpretable representations, having intensive computation, managing scarce or biased data, as well as deriving strong robustness across diverse domains or conditions.

III. Proposed TCADNet Framework for Cognitive Emotion Recognition

The proposed study proposes a Temporal Convolutional Attention-Driven Network (TCADNet) to learn to detect robust cognitive emotions through synchronized multimodal EEG fNIRS physiological signals. The framework is used to describe a multi-stage workflow that is structured to efficiently model the complex time dynamics of neural processes and enhance their classification performance. The raw EEG and fNIRS signals are first preprocessed (removal of noise, band-pass filtering, artifact correction, normalization and temporal segregating) to guarantee the quality and reliability of the signals. The multimodal signals are then preprocessed and synchronized, followed by feature-level fusing and inputting in a Temporal Convolutional Network (TCN) to extract hierarchical temporal features. The TCN uses causal and dilated one-dimensional convolutions with residual links that allow the model to learn both short-term neural variations along with the long-range temporal correlations of cognitive-emotional reactions.

Fig. 1 explains how the current embodiment is structured, incorporating a number of EEG and fNIRS signals to determine the locations of the emotional states of discrete cognitive emotion categories. In order to achieve better interpretability and discriminative power, there is a temporal attention mechanism that is incorporated after the TCN layers. This process gives various time segments adaptive weights that enable the network to prioritize emotionally salient neural patterns. To address class imbalance and improve generalization, a one-dimensional Generative Adversarial Network (1D-GAN) is incorporated to synthesize realistic minority-class EEG–fNIRS time-series samples. The attention-enhanced representations are then passed through fully connected layers, and a SoftMax classifier predicts discrete cognitive emotion states. The proposed TCADNet framework demonstrates improved accuracy, robustness, and scalability, making it suitable for real-time multimodal emotion recognition and AI-assisted mental health monitoring applications.

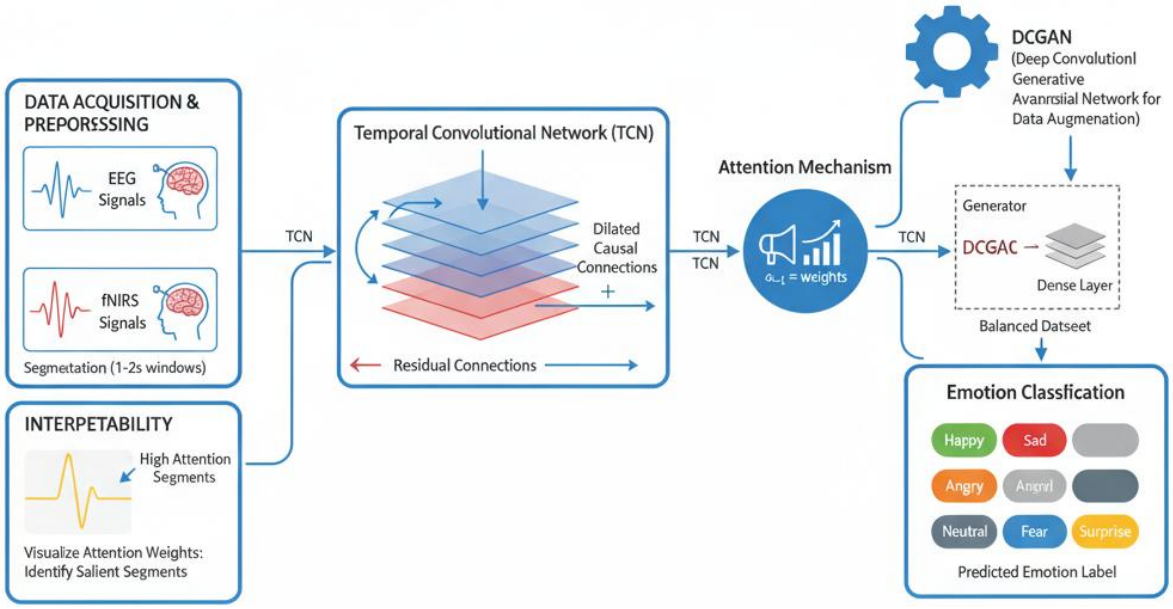


Fig. 1. Overall architecture.

A. Data Collection

The study utilizes the REFED dataset (Real-time EEG–fNIRS Emotion Dataset) [24], which provides synchronized multimodal neurophysiological recordings for cognitive emotion recognition. The data consists of the simultaneous electroencephalography (EEG) and functional near-infrared spectroscopy (fNIRS) signals recorded when the subjects are presented with controlled emotional stimuli.

EEGs represent fast neural electric activation with high temporal resolution, whereas fNIRS detects slower hemodynamic changes related to cortical activity. These modalities are complementary, which allows a more detailed study of the processes of emotional processing.

The NIRx Medical Technologies NIRSport2 system was used to record brain hemodynamic activity in this study, having a sampling rate of about 10 Hz. It employs two wavelengths (760nm and 850nm) to determine variations in oxygenated hemoglobin (HbO) and deoxygenated hemoglobin (HbR). The raw optical density measurements were converted to changes in hemoglobin concentration by the Modified Beer Lambert Law.

EEG recordings were made simultaneously to obtain fine-grained neural dynamics by using a multi-channel acquisition system, and with a higher sampling frequency. To provide the temporal synchrony of fNIRS and EEG modalities, event markers were added when presenting the stimulus.

Multimodal recordings were split into temporal windows of constant length relating to the emotion stimulus. These synchronized EEG-fNIRS segments are the input to the suggested Temporal Convolutional Attention-Driven Network (TCADNet) in cognitive emotion classification.

B. Data Preprocessing

Preprocessing is supposed to enhance the quality of signals and take the EEG and fNIRS data to a deep learning model level. Physiological recording is normally corrupted with motion artifact, baseline drift and noise. In order to remove such noise, NIRSlab is used in this study without a significant loss of time and amplitude features. The signals are de-trended and de-filtered to eliminate the distortions in the high or low frequencies, which could affect the feature extraction. The lost or corrupted data points are interpolated to prevent loss of information. Bandpass filtering was utilized to eliminate unwanted frequencies, while normalization approaches, such as min-max scaling, were employed to standardize the input. Signals were segmented into fixed-length time windows to ensure temporal continuity. Next, missing data points were filled in using mean value interpolation to avoid data loss. This stage ensures only clean, standardized signals are input into the deep learning model. Signal processing is the foundation of successful temporal modeling.

1) Normalization: Scale features to a standard range (e.g., 0–1):

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

In Eq. (1), original feature value is X, X_{min}= the feature's lowest value and X_{max} is the highest possible value of the trait. X' is the value of the normalized feature.

2) Missing values: Can be handled using mean imputation:

$$x_i = \frac{\sum_{j=1}^{n} x_j}{n} \quad (2)$$

In Eq. (2), x_i— to infer a missing value, x_j is the feature's observed values and n—number of values observed.

C. Multimodal Temporal Synchronization

EEG and fNIRS signals have different sampling rates and record different complementary neural activity. EEG has a high temporal resolution as it reveals rapid electrical activity, and fNIRS has slower responses to hemodynamic processes, which are related to cortical activity. In order to facilitate successful multimodal fusion, both modalities have to be synchronized in time.

Let EEG and fNIRS represent the EEG and fNIRS sampling rates, respectively. EEG was usually higher in sampling frequency and therefore downsampled to achieve an effective temporal resolution of fNIRS. As an alternative, fNIRS signals have been linearly interpolated to match EEG samples in time.

Both modalities were divided into fixed-length sliding windows of length T seconds with an overlap of 50%. The temporal alignment of each window was provided in accordance with the synchronized event markers in order to make the EEG and fNIRS fragments match the identical interval of the emotional stimulus. The temporal fusion was then carried out by concatenating the synchronized segments along the feature dimension to carry out feature-level fusion before temporal modeling. This synchronization strategy provides temporal coherence and complementary neural information to make multimodal emotion recognition robust.

D. Feature Extraction

The feature extraction process transforms signals that have been preprocessed into an individual meaningful representation to recall emotional information. In that work, the features are drawn in the form of a TCN as the input layers of the network, which itself pulls out the patterns on the basis of the raw EEG and fNIRS sequences. Convolutional filters detect local variations in time that are an indication of emotional states. The hierarchical convolutional net allows the net to learn increasingly more abstracted versions, not only of the fine-grained signal structure but also of the high-level temporal structure. The other links ensure continuity of information, where information is not lost in the process of layer stacking. This prevents hand-engineered features, which are likely to be biased or incomplete. Extracted features reduce the dimensions and, at the same time, retain discriminative power and can be utilized in downstream time modeling. Learning of every emotion category of the network is included in the network adapt, which is more sensitive to small differences. A proper day-by-day modeling step is ensured by good feature extraction. This stage captures both cross-modal and modality-specific patterns and correlations. The features that result are discriminative and temporal ready as well as strong. Learning of

emotions in a more generalizable and precise way entails feature extraction.

$$f_t^l = \sigma \sum_{k=0}^{K-1} W_k^l . x_{t-d.k}^{l-1} + b^l \quad (3)$$

In Eq. (3), the value of f_t^l at layer l is calculated as the result of a temporal convolution operation with a kernel size K and dilation factor d. The weights W_k^l and bias b^l are applied to the output of the last layer $x_{t-d.k}^{l-1}$ and combined by an activation function σ to ensure that short-term and long-term temporal interactions are well represented.

E. TCADNet Framework

TCADNet utilizes Temporal Convolutional Networks to accomplish the concept of temporal modeling. In contrast to recurrent neural networks, TCNs use causal convolutions to make sure that the future predictions at any moment are made with the help of the current and past data only and, therefore, preserve the order of the signal. Moreover, it employs dilated convolutions to expand the receptive field of the model without expanding its depth, and in such a manner, the model can be capable of capturing various emotional variations in the short term, not to mention being able to capture long-term correlations. These are the various methods employed by the TCN to vouch out discriminative temporal characteristics of the EEG signal and fNIRS signal.

1) Temporal modeling (Using Temporal Convolutional Networks – TCN): The sequential correlation of EEG and fNIRS is the temporal modeling. The TCNs are used, so as to take care of the long-term time trends and short term. Causal convolutions have the advantage that they do not cause the sequence of the signals across time to be lost, such that future predictions do not leak their knowledge of the past. The expanded convolutions increase the receptive field and this enables the network to acquire the long-range dependencies without necessarily overgoing too deep. Remnant links enhance the gradient flow by stabilizing deeper network training. Meanwhile, TCNs have the ability to train; therefore, training is not as time-consuming as recurrent networks such as LSTMs. The network is informed on the dynamism in the emotional states and it separates out the minor time trends among the participants. Temporal modeling also captures features by taking instantaneous and dynamic signal attributes. The time dynamics have been well captured in TCN output and information is stored at varying scales. The step plays an important role in capturing of the development of emotions. The input of the attention mechanism is learned time features. Another feature of the model is the use of effective awareness of emotions in real-time as a result of the effective time series modeling. In Eq. (4), TCN has been expounded with regard to a sequence.

$$F = TCN(X) \quad (4)$$

2) Attention-based feature weighting: To improve the interpretability and performance of recognition, we apply an attention mechanism to the output feature representations of layers of TCN. The attention module will give a weight to each temporal segment in view of their emotional patterns and, therefore, weight on a segment that has a stronger emotional pattern will be made greater. This cautious weighting gives the model the opportunity to identify and assign greater weight to the more informative components of the signal and underweight the noise and other unimportant components of the signal. The weighted feature attention thereof provides a more discriminate and finer representation of the emotional condition. During the selection of the emotionally informative cuts of the TCN output, the process of attention is combined. Each of the temporal features is given a weight, which is the contribution of the feature in classifying emotions. It is the active change of the process with regard to every input sequence to introduce pressure of the close approach of the emotional pattern of the participants. The weighted features reduce the impact of minor variations on the generalization more. It is another step of the temporal modeling since it prioritizes important information and does not overlook the rest of the sequence. The focus provides it with a fairly definite form. It also makes the classification more accurate as it minimizes the confusion of the closely related classes of emotions. It is so because the time center dependencies and feature relevance are also considered by the model, as TCN is also accompanied with attention. The emphasis based on the consideration of the attention will ensure the sentiments are projected appropriately in a forceful and comprehensible manner.

$$F_{att} = \sum_{t=1}^{T} \propto_t F_t \quad (5)$$

In Eq. (5), the attention module, F_t, is the TCN feature at time t, and $\propto_t$ refers to the time t attention weight of that feature. Final weighted feature vector F_{att} is calculated as a sum of TCN features multiplied by their attention weights that underline the most informative temporal segments of emotion recognition.

3) Data augmentation and class imbalance handling: To manage class imbalance during emotion recognition, a Deep Convolutional Generative Adversarial Network (DCGAN) is used to produce synthetic EEG and fNIRS signal sequences of the tiny emotion classes. The generator network is trained to generate natural time-series segments that can reproduce the visual appearances of actual physiological signals in terms of time. These generated sequences are checked by the discriminator network and distinguished between the real EEG/fNIRS signals and the generated scores. Both networks are adversarially trained: the generator should be able to make the discriminator use them, whereas the discriminator should be better at recognizing synthetic sequences.

The produced synthetic cues retain both temporal and spectral features that are important in the recognition of emotions. The synthetic sequences are merged with the original dataset to form a balanced training set, which enhances generalization and minimizes bias in the majority of types of emotions. Incorporation of the data produced by GAN makes sure that TCADNet acquires a strong feature in all kinds of emotions and improves the level of accuracy and stability in real-time student stress prediction tasks, as in Eq. (6):

$$D_f = \log(1 - D(G(z)) \quad (6)$$

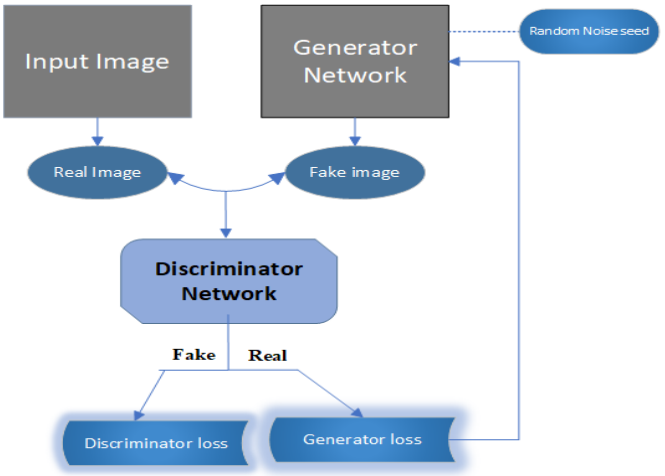


Fig. 2. Architecture of a DGAN.

Fig. 2 demonstrates that the DCGAN method combines convolutional neural networks and a GAN architecture.

The overall loss of the discriminator is given in Eq. (7):

$$D_l = \frac{1}{m}\sum_{l=1}^{m}(\log(D(x^l)) + \log\left(1 - D(G(z^l))\right)) \quad (7)$$

Conversely, the convolutional layers of the generator are used to up-sample the input picture in every convolutional layer using transposed convolutions. Noise is passed through the layers to the extent that the network successively up-samples the image to the size of a real image. In Eq. (8), the loss function is denoted.

$$G = log\ (1 - D(G(z))) \quad (8)$$

The final stage of the TCADNet framework performs emotion classification using the attention-enhanced temporal representations. The weighted feature vector obtained from the temporal attention module is passed through fully connected layers to learn nonlinear decision boundaries and reduce dimensionality. Model performance is evaluated using accuracy, precision, recall, and F1-score. To ensure robustness and generalization across subjects, k-fold cross-validation is employed. This final classification stage enables reliable and scalable cognitive emotion prediction suitable for real-time multimodal emotion recognition applications.

Algorithm 1 explains the process of identifying the emotions of EEG and fNIRS signals through TCADNet. EEG and fNIRS signals are synchronized, then band-pass filtered, artifact isolated, and normalized, followed by fixed-length window segmentation. A one-dimensional GAN is used to augment minority emotion classes to deal with the imbalance in classes. These multimodal segments are then fused and taken into a Temporal Convolutional Network (TCN) that uses both causal and dilated convolutions to extract hierarchical temporal features. An attention mechanism is a lightweight mechanism that gives adaptive weights to the salient temporal segments. Representation of the feature weighted by attention is passed to fully connected layers and a SoftMax classifier to predict emotions. The model has been trained on a cross-entropy loss and optimized with Adam, which is efficient in terms of real-time inference.

Algorithm 1: Real-Time Cognitive Emotion Recognition Using TCADNet

Input: EEG and fNIRS signals, corresponding emotion labels
Output: Predicted emotion class for each input signal segment
BEGIN
For each signal s in [X_EEG, X-fNIRS]:
Remove noise and artifacts using NIRSlab
Apply bandpass filtering
Normalize signals: s-norm = (s - mean(s)) / std(s)
Segment into fixed-length windows of size L
If
any missing data:
For each underrepresented emotion class c:
Generate synthetic signals using DCGAN:
Add X-synthetic-c to training set
For
each windowed segment x in preprocessed data:
F = TCN(x)
For each temporal feature F_t in F:
Skip noisy feature
If
max(Y-hat) < confidence-threshold:
Mark as uncertain
While epoch < max_epochs:
Forward pass: Compute F_{att}-> Y_{bat}
Backpropagate gradients
Update weights using optimizer
If
validation_loss does not improve for N consecutive epochs:
training early
Apply same preprocessing and feature extraction
Predict emotion in real-time
If
Y_{bat} uncertain:
Trigger alert or request additional input
END

Although transformer-based architectures have recently been applied to emotion recognition, the proposed TCADNet introduces a fundamentally different and computationally efficient temporal fusion strategy. Transformers rely on global self-attention with quadratic complexity, which increases memory usage and latency for long physiological sequences. In contrast, TCADNet employs causal and dilated convolutions within a Temporal Convolutional Network (TCN), enabling long-range dependency modeling with linear complexity O(T). This significantly reduces computational overhead and supports real-time EEG–fNIRS processing.

Moreover, the attention mechanism in TCADNet is not full multi-head self-attention. Instead, it acts as a lightweight temporal weighting layer applied after convolutional feature extraction, enhancing interpretability while avoiding heavy parameterization. Unlike data-hungry transformer models, the

convolutional inductive bias of TCN improves stability under limited physiological datasets. The combination of efficient temporal convolutions, lightweight attention, and 1D-GAN augmentation establishes a scalable and latency-aware multimodal framework distinct from transformer-based approaches.

IV. Results and Discussion

This research shows the need for specific and live emotion orthopnosing of mental health monitoring because the existing systems fail in the complex quality of temporal interdependence and multimodal body responses. To fill up these gaps, a framework of cognitive emotion recognition is proposed to be based on the EEG and fNIRS measurements to argue for the proposed TCADNet. TCADNet model adopts preprocessing, a convolutional layer on the basis of temporal feature extraction, a classification sub-module feature weighting on the basis of an attention mechanism, and feature weighting using an attention mechanism.

TABLE I. Experimental Setup

Component Type	Specification / Tool
Hardware	Intel i9-12900K CPU, 32 GB RAM, NVIDIA RTX 3080 GPU
EEG Device	32-channel EEG system
fNIRS Device	REFED dataset
Software	Python 3.10, TensorFlow 2.12, NIRSlab

The experimental setup in Table I consists of EEG and fNIRS acquisition instruments, preprocessing packages, and a deep learning platform to execute TCADNet. EEG traces are measured with the help of standard electrodes, whereas fNIRS traces are measured with the REFED dataset. NIRSlab software is used to carry out signal preprocessing (denoising, filtering, normalization, segmentation). TCADNet model with TCN, attention and GAN modules is based on Python (TensorFlow/Keras) and is accelerated on GPU. The data augmentation and model training are conducted on the high-performance computing hardware to provide a real-time perception of emotions and ensure performance evaluation.

TABLE II. Numerical Raw Data

Modality	Sampling Rate	Channels	Window Size	Features Extracted
EEG	256 Hz	32	4 sec (1024 samples)	Delta–Gamma (1–45 Hz)
fNIRS	10.17 Hz	16	4 sec (41 samples)	HbO, HbR (μmol/L)

Table II shows the numerical raw data:

- EEG band-pass filtered: 1–45 Hz
- fNIRS converted to HbO/HbR via modified Beer–Lambert law
- 50% window overlap

A. Training and Validation Accuracy

Fig. 3 shows the training and validation accuracy curve of the TCADNet, which shows that the accuracy of the model improves gradually with the number of epochs (50, 100, 150). The training and validation accuracy of the model at the first epoch are 97.9 and 97.5, respectively, which means that the model already starts to learn any meaningful patterns in the EEG and fNIRS data. Training accuracy goes up to 98.4%, and validation accuracy goes up to 98.2 as the number of training epochs increases to 100, indicating better feature extraction ability. The performance of the model is at its best at the 150 epochs, at which training and validation accuracy at 98.8 and 98.7, respectively. The overall similarity in the training and validation accuracy across all epochs is evidence of the fact that TCADNet has the ability to generalize without overfitting. This consistent convergence effect indicates the strength of the temporal convolution and attention-based architecture to extract discriminative emotional patterns of multimodal physiological signals during real-time emotion recognition tasks.

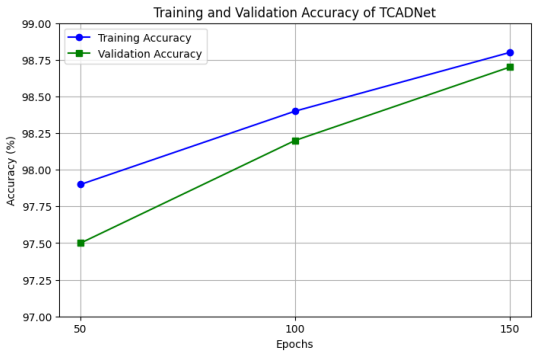


Fig. 3. Training and validation accuracy.

B. Training and Validation Loss

The epoch-wise training and validation loss curve, as in Fig. 4, shows a steady and moderate decrease in the loss value during training, indicating that the models were converging. At 50 epochs, the training loss reaches 0.085 and validation loss reaches a higher value of 0.112, which means that the model is initially experiencing some problems with generalizing to unseen data. Training loss and validation loss narrow to 0.062 and 0.089, respectively, as training advances to 100 epochs, which is an indication of enhanced learning and improved feature modeling. The training loss and validation loss are minimum when there are 150 epochs, that is, of 0.041 and 0.065, respectively. The significant decline of both curves without breaking the line proves a fact that the TCADNet does not overfit during the training. The distribution of validation loss over time especially shows the high level of generalization of the model on various emotional states based on EEG and fNIRS physiological responses.

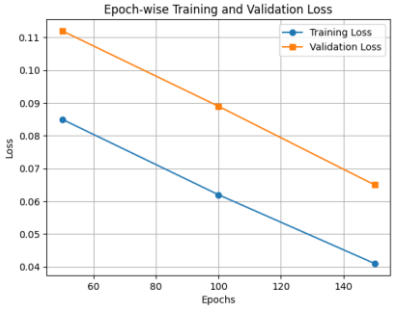


Fig. 4. Training and validation loss.

C. Class-Level Analysis

The emotion-wise accuracy bar chart of TCADNet in Fig. 5 measures the classification accuracy of the five different emotional categories of Happy, Sad, Angry, Fear and Neutral. The maximum recognition of the Happy class is 99.1 per cent which is the highest recognition rate of positive emotions of EEG and fNIRS signals which will be a very good discrimination of positive emotions. The next in rank with the accuracy of 98.9 is the Neutral class that provides conclusive evidence of the correct identification of the conditions of baseline emotional states. Angry class has an accuracy of 98.6 and Sad class has a score of 98.4 which shows that the model can differentiate between physiologically similar negative emotions. The Fear class that is physiologically the most difficult to detect has the accuracy of 98.2, which is also very impressive.

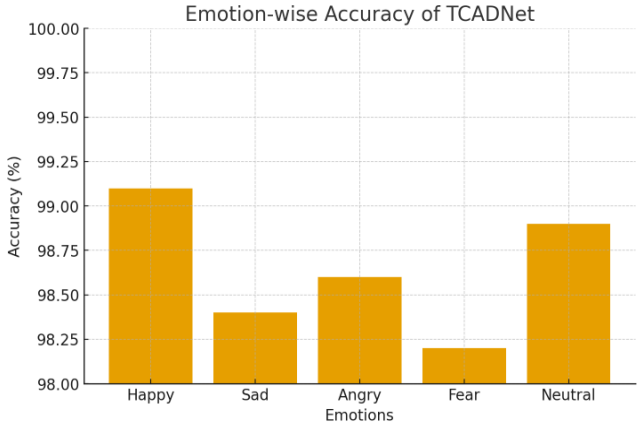


Fig. 5. Emotion-wise accuracy of TCADNet.

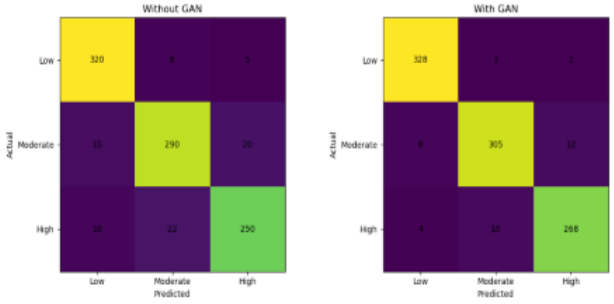


Fig. 6. Confusion matrix (without GAN vs. with GAN).

The comparison of the confusion matrix in Fig. 6 shows that the use of GAN-based data augmentation alters the classification accuracy of TCADNet at three stress degrees: Low, Moderate, and High. In the absence of GAN augmentation, the model is able to classify 320 Low, 290 Moderate, and 250 High stress instances correctly, with a few misclassifications that are visible especially in the Moderate and High category, which indicates that it has difficulties in differentiating between physiologically overlapping stress cases. The augmentation with GAN increases the performance of classification on all three classes. The attempts of the Low class gain a correct prediction to 328, the Moderate to 305, and Hard stress to 268. The decrease in the number of misclassifications of all categories supports that GAN augmentation is effective in boosting the training data distribution, which aids the model to acquire more boundaries of discrimination of stress levels. The findings confirm that GAN-based augmentation is a new essential element of TCADNet, which substantially enhances the generalization and strength of multimodal stress and emotion recognition tasks.

Multi-Class ROC curve in Fig. 7 shows the ability to classify the discrimination of TCADNet in three categories of stress level, i.e., Low, Moderate, and High. The values of the Area Under the Curve (AUC) are 0.50 in the case of the Low class, 0.52 in the case of the Moderate class, and 0.48 in the case of the High class. These AUC values imply that the model is significantly closer to the random classification threshold in distinguishing between classes of stress in the ROC metric, and therefore, although the overall accuracy is high, the probability separation between the individual stress classes still needs to be improved.

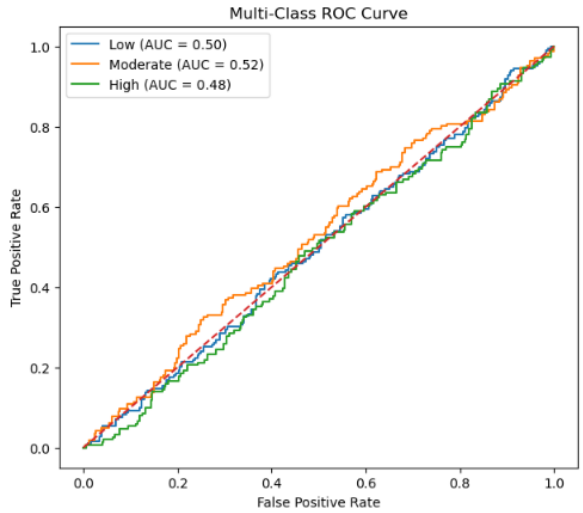


Fig. 7. ROC curve.

D. Ablation Study

The ablation study bar chart in Fig. 8 analyses the personal and structural contribution of each architectural component in the TCADNet to the total emotion recognition accuracy. The initial LSTM model has an accuracy of 93.4% which forms a point of reference in terms of sequential temporal modeling. The accuracy with the replacement of LSTM by Temporal Convolutional Network (TCN) is enhanced to 96.8, which shows that TCN is better than LSTM in representing long-range temporal variations in the EEG and fNIRS signals. The introduction of attention mechanism to TCN makes it further to 97.9, which is a sure indication that attention-based weighting of features is effective in the highlighting of emotionally discriminating temporal areas. The most accurate results are obtained with the full TCADNet architecture that combines TCN, attention mechanism, and data augmentation via GAN and has the highest accuracy of 98.7. Every step towards the program addition adds value to the overall performance advancement, which justifies the synergistic philosophy of design of TCADNet and serves as the evidence that the three elements are all needed to deliver outstanding real-time performance on emotion recognition.

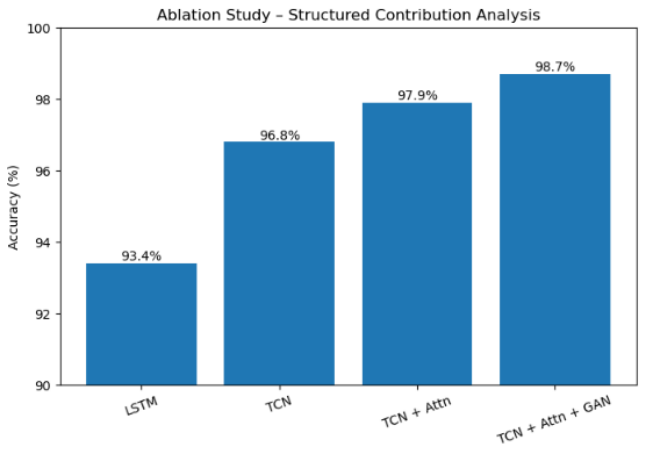


Fig. 8. Ablation study.

The bar chart group analysis in Fig. 9 assesses the influence of the attention mechanism on TCADNet performance in two setups, that is, in the absence of attention and the presence of attention, focusing on four measures: accuracy, precision, recall and F1-score. In the absence of the attention mechanism, TCADNet attains an accuracy of 97.1, a precision of 97.0, a recall of 96.9 and a F1-score of 96.95, which indicates that it is a competent though relatively low performance in determining the presence of emotionally salient temporal features using the physiological signals. Once the attention mechanism is added to the architecture, all the metrics of performance increase significantly. The accuracy increases to 98.7, precision to 98.55, recall to 98.6 and the F1-score to 98.55. This significant enhancement in all measures attests to the fact that the attention module is an important part of focusing on emotionally relevant temporal changes in EEG signals and fNIRS, which is effective in noise and irrelevant variations suppression and improving overall discriminative capabilities and interpretability of the TCADNet model.

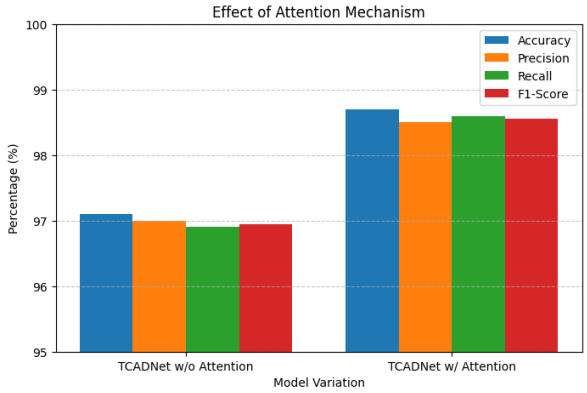


Fig. 9. Effect of attention mechanism.

E. Comparative Performance Analysis

In Fig. 10, the performance in emotion recognition is compared. The accuracy rate of YOLOv8 CNN and ViT was identified to be 72 per cent, whereas the precision, recall, and F1-score are also within the same range and mediocre only. The enhanced generalization of the accuracy, precision and recall of 83 per cent was achieved using the MLP model. PSCS model was much more efficient compared to the other methods with an accuracy of 97.5 and with lower recall (96.7) and F1-score (97.98) shows that it is strong in emotion classification. It is shown in Table III.

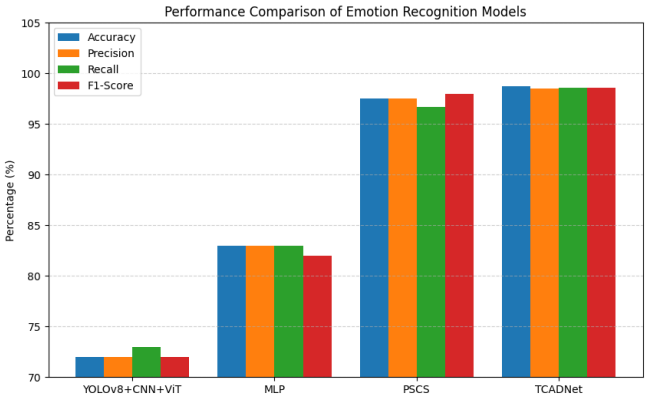


Fig. 10. Performance metrics.

TABLE III. PERFORMANCE COMPARISON

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
YOLOv8+CNN+ViT [25]	72	72	73	72
MLP [26]	83	83	83	82
PSCS [27]	97.5	97.5	96.7	97.98
Proposed TCADNet	98.7	98.5	98.6	98.55

TCADNet was compared to numerous state models, including YOLOv8+CNN, MLP and PSCS architecture. Albeit the YOLOv8+CNN+ViT model achieved accuracy of 72 per cent and the MLP achieved the same, the PSCS model achieved with 97.5 per cent. TCADNet worked better compared to all these models by a margin of at least 1.2 per cent with an accuracy rating of 98.7 per cent. This progress can be attributed to the fact that a synergistic comparison of temporal convolution, attention-based feature weighting, and GAN-based augmentation is generated and that it overcomes the drawbacks of existing architectures. These results confirm the hypothesis that TCADNet is a better and more comprehensible model of mental health monitoring systems in identifying cognitive emotion.

F. Computational Efficiency and Real-Time Capability

The computational complexity and inference latency comparison in Table IV evaluates four models, namely LSTM, ViT, PSCS, and TCADNet, across four key efficiency metrics: parameters, GPU inference time, CPU inference time, and memory consumption. The LSTM model contains 6.8M parameters with GPU and CPU inference times of 35ms and 118ms, respectively, consuming 52MB of memory. The ViT model is the most computationally expensive, with 12.4M parameters, 72ms GPU inference, 210ms CPU inference, and 94MB memory usage. The PSCS model offers improved efficiency with 5.1M parameters, 24ms GPU inference, 96ms CPU inference, and 41MB memory. TCADNet demonstrates the highest computational efficiency among all compared models, requiring only 4.2M parameters, achieving 17ms GPU

inference, 82ms CPU inference, and consuming just 33MB of memory. These results confirm that TCADNet achieves superior accuracy while maintaining the lowest computational overhead, making it highly suitable for real-time deployment in resource-constrained mental health monitoring environments.

TABLE IV. COMPUTATIONAL COMPLEXITY AND INFERENCE LATENCY COMPARISON.

Model	Parameters (M)	GPU Inference (ms)	CPU Inference (ms)	Memory (MB)
LSTM	6.8 M	35 ms	118 ms	52 MB
ViT	12.4 M	72 ms	210 ms	94 MB
PSCS	5.1 M	24 ms	96 ms	41 MB
TCADNet	**4.2 M**	**17 ms**	**82 ms**	**33 MB**

G. Discussion

The experimental results indicate that the suggested TCADNet system is a decent model of multimodal temporal dynamics to understand cognitive emotion recognition with EEG and fNIRS signals. The training and validation curves indicate that the convergence is steady, the validation accuracy was the highest at 150 epochs with the least loss difference between training loss (0.041) and validation loss (0.065), which is a high level of generalization and a low level of overfitting. Class-wise analysis also demonstrates that there is a similar pattern of performance in all emotional states, with Happy (99.1) and Fear (98.2) classes showing the highest and the lowest accuracy respectively which confirm the good discrimination ability even of the physiologically weak emotions. The comparison of confusion matrices indicates the effect of GAN-based augmentation, which substantially improved the classification of stress categories of the minority (e.g., High stress correctly classified of 250 to 268 instances). The ablation study supports the ranking of the contribution of each architectural component, where performance was enhanced substantially in terms of 93.4% (LSTM) to 96.8% (TCN), to 97.9% (TCADNet integration to the fullest). It proves the synergistic advantage of causal dilated convolutions, lightweight temporal attention, and GAN augmentation.

TCADNet compared with YOLOv8+CNN+ViT model (72%), MLP (83%), and PSCS (97.5%) models had better performance. Also, TCADNet had the minimal computational complexity (4.2M parameters) and the shortest inference time on the GPU (17 ms), confirming the fact that it can be used in real-time. In general, the findings validate that TCADNet is more accurate, interpretable and more efficient in its computations as a means of multimodal monitoring of mental health.

V. CONCLUSION AND FUTURE WORK

The proposed study introduced the TCADNet as a time-aware convolutional attention-based multimodal emotion recognition system that involves EEG and fNIRS. The model obtained a high validation accuracy (98.7), model class-wise performance (98.2% to 99.1) and the generalization performance is high, as there is a minimum loss divergence. The outcome of the ablation validated that TCN, attention, and GAN augmentation were effective and the performance increased to 98.7% as compared to 93.4% (LMST baseline). Compared to existing models, comparative analysis showed higher accuracy and lower computational cost (4.2M parameters, 17 ms inference on a GPU) favorably. Such results confirm the suitability of TCADNet as a scalable, interpretable and real-time framework in mental health monitoring systems to be recognized in terms of cognitive emotion.

Future directions involve reinforcing the work by trying to increase the temporal discrimination performance through better separation of probabilities among stress classes, as well as increase AUC values. Noisy or missing modality analysis of robustness will also be performed to test whether the deployment is stable. On top of that, adaptive cross-subject transfer learning and lightweight edge deployment optimization will also be considered. The framework could be further expanded by extending the length of time series and multimodal datasets to a more effective scale and apply to clinical settings of mental health monitoring in the real world.

REFERENCES

[1] S. Pal, S. Mukhopadhyay, and N. Suryadevara, "Development and Progress in Sensors and Technologies for Human Emotion Recognition," Sensors, vol. 21, no. 16, p. 5554, Jan. 2021, doi: 10.3390/s21165554.

[2] N. Avital, I. Egel, I. Weinstock, and D. Malka, "Enhancing Real-Time Emotion Recognition in Classroom Environments Using Convolutional Neural Networks: A Step Towards Optical Neural Networks for Advanced Data Processing," Inventions, vol. 9, no. 6, p. 113, 2024.

[3] K. Nassiri and M. A. Akhloufi, "Recent advances in large language models for healthcare," BioMedInformatics, vol. 4, no. 2, pp. 1097–1143, 2024.

[4] A. Payandeh, K. T. Baghaei, P. Fayyazsanavi, S. B. Ramezani, Z. Chen, and S. Rahimi, "Deep representation learning: Fundamentals, technologies, applications, and open challenges," IEEE Access, vol. 11, pp. 137621–137659, 2023.

[5] H. Dong, W. Ma, Y. Wu, J. Zhang, and L. Jiao, "Self-supervised representation learning for remote sensing image change detection based on temporal prediction," Remote Sens., vol. 12, no. 11, p. 1868, 2020.

[6] H. Chen et al., "Enhancing human activity recognition in smart homes with self-supervised learning and self-attention," Sensors, vol. 24, no. 3, p. 884, 2024.

[7] E. Dritsas, M. Trigka, C. Troussas, and P. Mylonas, "Multimodal Interaction, Interfaces, and Communication: A Survey," Multimodal Technol. Interact., vol. 9, no. 1, p. 6, 2025.

[8] X. Zhang et al., "Emotional-Health-Oriented Urban Design: A Novel Collaborative Deep Learning Framework for Real-Time Landscape Assessment by Integrating Facial Expression Recognition and Pixel-Level Semantic Segmentation," Int. J. Environ. Res. Public. Health, vol. 19, no. 20, p. 13308, Jan. 2022, doi: 10.3390/ijerph192013308.

[9] Y. Li, K. Zhang, J. Wang, and X. Gao, "A cognitive brain model for multimodal sentiment analysis based on attention neural networks," Neurocomputing, vol. 430, pp. 159–173, 2021.

[10] A. Singh and H. Nair, "A neural architecture search for automated multimodal learning," Expert Syst. Appl., vol. 207, p. 118051, 2022.

[11] Y. Ojo, O. A. Makinde, O. V. Babatunde, G. Babatunde, and S. Okeowo, "Evaluating AI-Driven Mental Health Solutions: A Hybrid Fuzzy Multi-Criteria Decision-Making Approach," AI, vol. 6, no. 1, p. 14, 2025.

[12] A. Pavlopoulos, T. Rachiotis, and I. Maglogiannis, "An overview of tools and technologies for anxiety and depression management using AI," Appl. Sci., vol. 14, no. 19, p. 9068, 2024.

[13] W.-J. Yan, Q.-N. Ruan, and K. Jiang, "Challenges for artificial intelligence in recognizing mental disorders," Diagnostics, vol. 13, no. 1, p. 2, 2022.

[14] I. Ul Hassan, R. H. Ali, Z. ul Abideen, A. Z. Ijaz, and T. A. Khan, "Towards effective emotion detection: A comprehensive machine

learning approach on eeg signals," BioMedInformatics, vol. 3, no. 4, pp. 1083–1100, 2023.

[15] A. Al Kuwaiti et al., "A review of the role of artificial intelligence in healthcare," J. Pers. Med., vol. 13, no. 6, p. 951, 2023.

[16] X. Xu et al., "A comprehensive review on synergy of multi-modal data and ai technologies in medical diagnosis," Bioengineering, vol. 11, no. 3, p. 219, 2024.

[17] N. Dhariwal et al., "A pilot study on AI-driven approaches for classification of mental health disorders," Front. Hum. Neurosci., vol. 18, p. 1376338, 2024.

[18] M. Mansoor and K. Ansari, "Artificial Intelligence-Driven Analysis of Telehealth Effectiveness in Youth Mental Health Services: Insights from SAMHSA Data," J. Pers. Med., vol. 15, no. 2, p. 63, 2025.

[19] S. Hoose and K. Králiková, "Artificial Intelligence in Mental Health Care: Management Implications, Ethical Challenges, and Policy Considerations," Adm. Sci., vol. 14, no. 9, p. 227, 2024.

[20] P. K. Nag, A. Bhagat, and R. V. Priya, "Expanding AI's Role in Healthcare Applications: A Systematic Review of Emotional and Cognitive Analysis Techniques," Authorea Prepr., 2024.

[21] A. Kargarandehkordi, S. Li, K. Lin, K. T. Phillips, R. M. Benzo, and P. Washington, "Fusing Wearable Biosensors with Artificial Intelligence for Mental Health Monitoring: A Systematic Review," Biosensors, vol. 15, no. 4, p. 202, 2025.

[22] X. Chen, H. Xie, X. Tao, F. L. Wang, M. Leng, and B. Lei, "Artificial intelligence and multimodal data fusion for smart healthcare: topic modeling and bibliometrics," Artif. Intell. Rev., vol. 57, no. 4, p. 91, 2024.

[23] Y. J. Lee, C. Park, H. Kim, S. J. Cho, and W.-H. Yeo, "Artificial intelligence on biomedical signals: technologies, applications, and future directions," Med-X, vol. 2, no. 1, pp. 1–30, 2024.

[24] REFED-dataset, "REFED2025/REFED-dataset · Datasets at Hugging Face." Accessed: Feb. 20, 2026. [Online]. Available: https://huggingface.co/datasets/REFED2025/REFED-dataset

[25] J. Aina, O. Akinniyi, M. M. Rahman, V. Odero-Marah, and F. Khalifa, "A hybrid Learning-Architecture for mental disorder detection using emotion recognition," IEEE Access, 2024.

[26] N. Azam, T. Ahmad, and N. U. Haq, "Automatic emotion recognition in healthcare data using supervised machine learning," PeerJ Comput. Sci., vol. 7, p. e751, 2021.

[27] K. Jawad, R. Mahto, A. Das, S. U. Ahmed, R. M. Aziz, and P. Kumar, "Novel cuckoo search-based metaheuristic approach for deep learning prediction of depression," Appl. Sci., vol. 13, no. 9, p. 5322, 2023.

Explainable AI with DRL for Smart Home Energy Management and Residential Cost Savings in IoT-Based Autonomic Systems

Prabagaran A[1], Srinivasan Sriramulu[2], S. Premkumar[3],
Sheelesh Kumar Sharma[4], Karthi Govindharaju[5], Angelin Blessy J[6]*
Gupshup Technology India Pvt Ltd, Bengaluru, India[1]
School of Computer Science and Engineering, IILM University, Greater Noida, India[2]
School of Computer Science and Engineering, Galgotias University, Greater Noida, India[3, 4]
Dept. of AI&DS, Saveetha Engineering College, Chennai, Tamil Nadu, India[5]
Dept of Computer Science, Medicaps University, Indore, MP, India[6]

***Abstract*—As more people seek ways to improve their homes and workplaces while reducing energy consumption, smart home systems are becoming increasingly prevalent. Unfortunately, the complexity and "black-box" nature of these systems made it difficult to deploy AI-enabled decision-making simulations, raising issues with explainability, confidence, transparency, responsiveness, and fairness. Explainable Artificial Intelligence (XAI), a rapidly developing discipline, addresses these problems by offering justifications for various decisions and behaviors of the systems. This research describes a novel method for IoT-based autonomic devices to control energy that combines XAI with Deep Reinforcement Learning (DRL) to achieve significant Home Energy Management System (HEMS) for household cost reductions. The proposed approach leverages XAI's features to improve the accessibility and transparency of DRL agents, helping consumers understand and trust autonomous power management decisions. By optimizing energy usage patterns and adapting to changing environmental conditions, the proposed solution ensures effective energy use while maintaining user comfort. Use in-depth modeling and real-world applications to demonstrate the solution's efficacy, highlighting its potential to reduce energy consumption costs and promote sustainable living. This study sets a new standard for clarity and flexibility in AI-driven smart home systems, paving the way for more reliable and user-friendly IoT software. It is important to note that developing a thermal dynamics model and understanding unidentified variables are not prerequisites for the proposed technique. Results from simulations based on real-world data show the resilience and effectiveness of the recommended strategy.**

***Keywords*—Explainable artificial intelligence; smart home energy management; deep reinforcement learning; Internet of Things; energy optimization**

I. Introduction

The Internet of Things (IoT) is an already developed innovation, which connects smart things and devices with each other and allows them to communicate with each other and provides consumers with new services. IoT alters the traditional electronic devices (actuators, sensing devices, RFID tags, mobile phones etc.) to be smart devices by maintaining and administering the basic technology. In turn, through enabling them to synchronize and share data, IoT enables the objects to see, hear, believe, and perform particular activities [1]. IoT brings many possibilities and opportunities to enhance the quality of life in different spheres of its applications such as intelligent buildings, smart cities, smart agriculture, financial systems, healthcare and military. IoT has the ability to produce tremendous amounts of data, which will have to be analyzed thoroughly [2]. Although this raw data can be, at best, of no use, AI systems allow extracting valuable knowledge and giving wise judgments that influence the life of people in such critical areas as healthcare or autonomous systems [3]. In recent years, IoT applications have heavily relied on AI models, much like other industries, to address issues arising from the growing volume of data and devices. Specifically, many AI models are used in autonomous network administration, device management, service administration, and large-scale IoT data analysis, all of which require highly accurate and precise decision-making processes [4].

Environmentally friendly sources provide over fifty percent of the power supply and are expected to quickly expand into the international energy markets. There is often a discrepancy between when green power is generated and when energy is needed. For instance, solar photovoltaic cells have great prospects for producing energy during midday hours, but the demand for power in residential structures is higher in the late afternoon [5]. When the demand is high, the tariffs are stronger and the price of electricity is lower compared to the times when the supply of energy is high. This means utilities unwillingly provide a reward to customers who consume energy at the high-supply times and punish those who consume energy at the high-demand times. There are even utilities that pay their customers a direct compensation upon low energy consumption by the customer during peak times [6].

Demand Response (DR) events imply the prudent decision-making of all the parties involved in the process, which lowers the energy expenses of consumers and manufacturers. In some emerging countries, suppliers use the practice of blackouts at all times and thus some parts of the country go black during a few hours of high demand but low supply. The number of homes that acquire battery-powered batteries has also increased

*Corresponding author.

due to the frequency of blackouts which are designed to store energy as an alternative source in the event that there is a blackout of the grid electricity. The increasing usage and the diminishing prices of these batteries with time has made them gain more and more use [7]. Different technologies can be used to enhance the building power consumption by shifting loads or reducing the loads using demand-side energy management. The deferrable loads usually include plug-in electric cars, air conditioning, ironing machines, and dishwashers. When there are a large percentage of non-deferrable loads, it is easier to satisfy the energy requirements with the use of energy storage instruments under the conditions of fluctuating power prices. The customers will be able to save money by simulating an effective energy management system that will help in maximizing the use of the battery. This strategy can also help to enhance grid stability through reduction of the demand during energy deficits [8]. The traditional methods maximize the battery operation through rule-based procedures. Such methods should be dynamic and flexible to counter the changes in tariff structures. The growing use of AI technology in our life prompts the necessity to learn how and why decisions are made [9]. Machine learning models gain in strength, they and even become less transparent and more complicated. These powerful models are opaque and are normally called black-box models, that is, they do not offer their internal justification to the user [10].

Researchers have recently taken an interest in XAI in order to solve these problems and develop more interpretable and understandable AI systems. XAI offers increased insight into the inner workings of black-box systems, by exposing the previously unknown or undetectable information about the input system like feature importance and characteristic associations. This enables the user to judge the results and the input parameters that affect the results [11]. In conjunction with strong and sophisticated models, the XAI methods provide excellent accuracy and interpretability. Research indicates that there is a need to comprehend issues to do with human beings in areas like neuroscience, computer science, theology and mental health. These researches indicate that decisions made without explanation may culminate into confidence problems or intentional and inadvertent bias. XAI helps to alleviate these problems by providing verification, improvement, learning, and legal compliance [12].

The numerous advantages of XAI are making it gain more and more popularity in a broad spectrum of uses, among them building highly transparent, trustworthy, and all-encompassing systems. The Artificial Intelligence systems intensify and integrate into the everyday life and business activities, although they are typically plagued with trust issues, confidence in the code, and bias in the model. XAI provides a far greater insight into the systems and there is no ambiguity or vagueness around the decisions made or solutions picked, and therefore the transparency of AI models remains intact [13].

XAI may be used to substitute conventional AI systems being utilized, which may be more influential on industrial production, supply chains, economic sectors, and asset management with enhanced development and sustainability innovations. The difficulty of these approaches can lead to the situation that decisions made will be complicated to be accepted by common consumers in the real world [14]. According to researchers, two potential uses can be identified in terms of energy consumption in smart home environments: anticipating personal preferences to different smart heating appliances in terms of temperature and anticipating power demand. In our study, predict thermal comfort on the basis of personal preferences through machine learning models and deep neural network-based energy prediction through benchmark datasets. Employ common XAI methods to get insights by explaining the forecasts in various forms to help decipher the complexity of the forecasts [15].

The XAI technology has attracted a considerable amount of attention both in research and practice with considerable progress being achieved due to the potential of the model to generate trustworthy inferences. Investments in a wide range of research disciplines are fueled by the rise of XAI across these applications. XAI is most commonly used in medical, banking, safety, military, and legal sectors [16]. XAI technology has demonstrated its potential to the point where interpretable AI models are now a necessity. The military industry is a real-world example of XAI's application. Google's cloud services are exploring how XAI can be used to construct comprehensive and transparent AI models [17]. The reinforcement Learning (RL) can outperform human decision-making in the majority of games, RL has become extremely popular, particularly in the gaming industry. The goal of the RL agent is to maximize cost reduction or minimize the amount that must be spent to achieve a certain level of usefulness [18]. As Artificial Intelligence (AI) and robotics become more prevalent in everyday life, human achievement is starting to suffer. It has demonstrated that pleasure rises in tandem with higher human performance effectiveness, which may come from the use of AI and automation [19]. The smart agent could exploit an edge case in the environment's regulations to gain significant advantages without completing the intended activity and behavior of the human agent, including repeated activity switching and longer time steps when setting the Thermostat (TH) parameters [20]. Additionally, multi-human studies, in which two individual agents are incorporated into a single house simultaneously, reveal that human models may adjust the sequence of tasks in this situation to minimize the disparities between their TH preferences, thereby reducing the time required for altering TH settings and optimizing comfort [21].

A. Problem Statement

The increasing complexity and access to IoT and smart home gizmos have complicated the management of household energy consumption in an appropriate manner. The traditional energy management is not typically computationally powerful and dynamically flexible enough to be used to optimize energy consumption. These systems often fail to comprehend the energy saving decisions that are taken by homeowners and they therefore tend to be mistrusted and not utilized in the most optimal way. It is urgent to have an advanced intelligent system that is energy efficient, economical and does not involve complex decision-making process. This system must employ the use of XAI-DRL to maximize the use of energy in smart homes to ensure that the system remains effective and reduces the likelihood of low consumer confidence. The XAI-

DRL-based HEMS performance evaluated using different metrics such as accuracy, precision and etc.

B. Motivation

The dynamism and increasing demands of today power consumption are the forces behind the development of a smart home energy management system to help cut down the household expenditure. The homeowners demand viable ways of reducing their electricity costs without sacrificing comfort or convenience, especially as the energy prices continue to increase. Furthermore, the mindfulness of the adverse environmental consequences of energy usage has increased and there is a need to have more efficient methods of energy usage to minimize carbon footprints and enhance environmental-friendly practices. The development of IoT gadgets in smart houses has its opportunities and difficulties. Although these sensors allow one to control the use of energy granulometry, their successful management demands more complex algorithms that can model the complex and changing interactions. The conventional power control systems are usually unable to accommodate various requirements and habits of the various households and thus an alternative method is needed that can adjust energy-conservation policies to individual houses continuously. Moreover, to ensure that AI-driven systems receive wide acceptance and are used successfully, homeowners have to know and have confidence in the decisions offered by such systems. Consequently, it is important that the devices be explainable. This openness assists the homeowners in understanding the reasons and methods of how some energy-conserving decisions are arrived at, which will create trust and acceptance of artificial intelligence-based energy management systems within smart homes.

The key contributions of the paper as follows:

1) Integration of Explainable AI with Deep Reinforcement Learning: Combines XAI with DRL for transparent, interpretable smart home energy optimization, fostering user trust.

2) Optimization of Energy Consumption and Cost Savings: DRL-based system autonomously learns optimal device schedules, reducing energy use and residential costs.

3) Explainability and User Trust: XAI clarifies DRL decisions, enhancing homeowner understanding, engagement, and acceptance of AI-driven energy management.

4) Experimental Validation and Performance Improvement: Validated with IoT data, XAI-DRL outperforms traditional methods in energy savings, cost, and interpretability.

5) Foundation for Future Research: Integrating explainability in smart homes enables future studies enhancing AI transparency in IoT environments.

The structure of the paper can be summarized as follows: The crucial importance of minimizing the energy use in smart houses in Section I is brought forward without owing to the lack of transparency and trust in the systems based on AI. Part II examines the current energy optimization techniques with the help of IoT and DRL, noting their inability to provide explanations. Section III discusses the proposed methodology and describes how a DRL-based system of real-time energy management was developed with XAI in order to enable users to be able to interpret the decision. The detailed experimental setup is presented with the use of the data of IoT devices on the energy consumption and cost analysis. Section IV show that DRL-XAI approach will save on energy and cost, as well as, enhance user trust by providing explainable insights. Section V provides a final remark of the importance of XAI in promoting increased use of AI-based energy management systems and indicate future research opportunities such as extending the system to real-life settings and scale-up of the system.

II. Related Works

Intelligent home environments are intended to increase the quality of life of people by means of different sets of detectors and actuators located strategically all over the house, automated deductions regarding human actions and their situational contexts. Smart houses make use of a variety of sensors to gather context-related information, including noise, movement, environmental temperatures, Wi-Fi signals properties, visual data (cameras, etc.), and presence/absence data [22]. It is a common practice to directly track the movements of the occupants via wearable devices. Activity recognition uses such sensor temporal sequence to determine and record the Activities of Daily Living (ADL) of the occupants. The smart home then uses these dynamically inferred activity labels to determine supportive actions (i.e. open the garage door, turn on the lights or change the thermostat) [23].

The problem of activity recognition in smart homes still exists and the challenge has to be addressed through innovative machine-learning techniques that can be used to automatically analyze sensor data. The difficulty herein lies in the different environments, users, sensor setups, and actuators all which are unique to each house and home. Researchers state that the ADL patterns can be more accurately represented in LSTM based systems because they are able to learn nonlinear representations of characteristics and model long term relationships in the patterns of data [24].

In this area, a number of large datasets on smart home activity recognition have been created to aid in extensive appraisals and benchmarking. These datasets are differentiated by the number of household members and the type of sensors to collect data and the variety of ADL actions. As an example, the ARAS database tracks the activity of a large number of residents of the house within a two-month time span with 20 house monitors [25].

Available and explainable ML models offer new options to describe black-box decision structures, which allow new explanatory styles to be developed to make AI algorithms more transparent and comprehensible. This is among the greatest influences of XAI-IoT environments. The technical abilities of XAI methodologies can be used to guarantee ethical integrity of XAI systems through IoT systems [26]. The use of XAI principles to the Industrial IoT (IIoT) is illustrated that proves the effectiveness of applying statistical concepts to explain models without relying on knowledge to ensure transparency. Implementation of XAI in risky IoT applications needs randomness of new sample. LIME-based XAI designs support

domain-invariant learning of features which are accurate in data handling and credible in interpretation. The problem of customization of XAI models to nonlinear data is still a major challenge, although continued work on data-driven enhancements of XAI models is now to solve those problems [27].

Although major advancements were achieved in maximizing energy management strategies in smart homes, there still remain difficulties in modelling thermal dynamics of buildings and the use of simplified mathematical models such as the Equivalent Thermal Parameters (ETP) method. As an example, offered a multi-agent reinforcement learning coupled with artificial neural network-based energy management strategy to achieve a low total cost of energy and customer dissatisfaction with HVAC loads and laundry machines. Advances in real-time learning provide the chance to solve problems in large state space problems, and reinforcement learning (RL) algorithms, including the ones introduced automatically find the best control policies without the need to have thermal dynamics subsystems [28].

There is also a widespread research on the interaction between humans and AI, and it has been shown that AI learning can be enhanced in case human input is added to the loops of RLs. Nonetheless, there is a fear of unforeseen or even harmful impacts of these kinds of interactions, which is demonstrated by studies where RL agents use environmental conventions to maximize gains at the expense of the system as a whole [29]. The smart homes of the future will probably have a large number of sensors to control the activities, such as energy saving, security, etc. There are some difficulties in planning the work of household appliances effectively according to the predictive model of human behavior and interaction with the smart home devices. Scholars are busy experimenting with the machine learning methodology to come up with complete recommendation systems to smart home consumers. The systems make the decision-making process more effective especially when choosing objects like clothes and food.

In conclusion, although renewable energy integration and optimization strategies in smart homes have majorly taken the low cost of electricity as a central theme in their research and development endeavor, energy consumption management is a key area that requires more research and development in the future. Innovative solutions such as the use of photovoltaic systems and DR systems in microgrids have a good future of satisfying the energy needs in a better way taking into account the comfort of users and the sustainability of the environment. The constant monitoring and adjustment of these systems will be crucial to reduce the level of discomfort to the user and maximize the level of energy utilization within the smart home settings [30].

III. Proposed System

To achieve energy efficiency in smart homes and household savings in autonomic structures based on the Internet of Things, XAI-DRL is a new solution that will combine state-of-the-art machine learning algorithms with well-defined and comprehensive approaches to the decision-making process. In this architecture, the direct signals that are obtained by multiple IoT devices are processed in DRL to enhance energy consumption in smart homes. By keeping an eye on environmental factors and patterns of consumption of energy, these gadgets let DRL agents make well-informed judgments that balance residents' peace of mind with the lowest possible energy expenses. The incorporation of XAI guarantees that consumers can comprehend and have transparency over the choices and actions made by the DRL agents. Homeowners are better able to embrace and depend on the technology since it increases user confidence and belief in the system. XAI improves the system's general efficacy and acceptability by giving clear descriptions of how energy-saving techniques are carried out. This results in more economical and environmentally conscious energy administration in smart home environments.

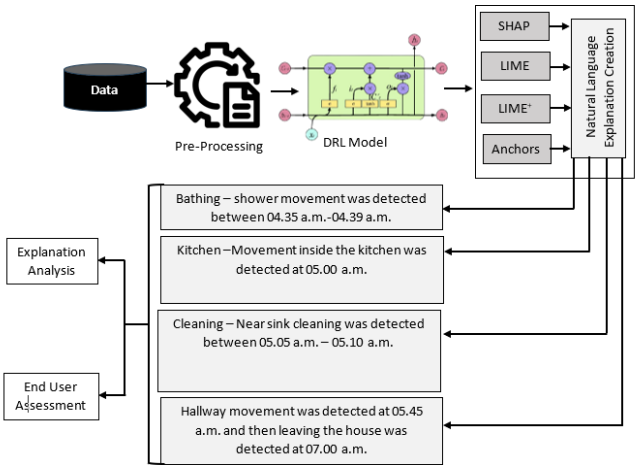


Fig. 1. Proposed work addressing the movement inside the house.

Fig. 1 addresses the aforementioned movement inside the house. Pre-process S such that it reflects the data required to produce descriptions of intelligent houses that are more appealing to average consumers. LIME is a model-agnostic XAI method that builds a model that can be understood locally around an assumption to clarify the forecasts of a black-box classification. Another model-agnostic XAI method called Shap computes Shapley values to determine the significance of each characteristic a black-box model uses for a forecast. Every technique is applied to produce a rationalization. It also expands the LIME approach by including LIME+ and ELIME+ clarifications, which provide more temporal and intuitive clarifications than LIME through recognizing blocks of time. Their outputs are easily understood and may be turned into descriptions for end users in straightforward language. Researchers evaluate the effectiveness of the four explanation styles that resulted along several pertinent parameters.

A. Preliminaries

- Reinforcement Learning: RL involves an agent interacting with an environment to maximize cumulative rewards over time.

$$\text{Policy: } \pi(a|s) \quad (1)$$

$$\text{Value function: } V^{\pi}(s) = E_{\pi}\left(\sum_{t=0}^{\infty} \gamma^{t} r_{t} \,|s_{0} = s\right) \quad (2)$$

$$\text{Q-Function: } Q^{\pi}(s) = E_{\pi}\left(\sum_{t=0}^{\infty} \gamma^{t} r_{t} \,|s_{0} = s, a_{0} = a\right) \quad (3)$$

Where: γ - discount factor; γ_t - reward at time step t.

Explainable AI (XAI): It focuses on making Al systems transparent and understandable to humans.

Model Interpretability: Methods to interpret and explain the decisions made by Al models.

Feature Importance: Techniques such as SHAP values or LIME

Here are some basic equations used in DRL and XAI:

$$\text{Bellman Equation: } *(s,a) = E_{s'}\left[r + \gamma \max_{a'} Q^*(s',a')|s,a\right] \quad (4)$$

$$\text{DQN Loss Function:}(\theta) = E_{s,a,r,s'}\left[\left(r + \max_{a'} Q^*(s',a';\theta^-) - Q(s,a;\theta)\right)^2\right] \quad (5)$$

$$\text{SHAP Value}\emptyset_y = \sum_{S' \subseteq S|(y)} \frac{|s'|!(|s|-|s'|-1)!}{|s|!}(f(S') - f(S)) \quad (6)$$

Where: S - set of all features; S' - features subset excluding y; f(S) – proposed model output of features S.

These preliminaries set the stage for applying DRL with XAI principles to optimize energy management and cost savings in smart home environments. They provide the theoretical foundation necessary to understand the subsequent methodology and results presented in the study.

The objective is to optimize the energy consumption in a smart home environment using DRL. Let's denote: E_t: Energy consumption at time step t; C_t : Cost associated with energy consumption at time step t; R_t: Reward at time step t, which is a function of E_t and C_t

The goal is to maximize the cumulative reward over a horizon T:

$$max_\pi = \sum_{t=0}^{T-1} \gamma^t R_t \quad (7)$$

Where: π denotes the policy governing the agent's actions (energy management decisions); γ is the discount factor ensuring future rewards are appropriately weighted.

B. Components of the Objective Function

Reward Function R_t: The reward function can be defined based on the objectives of energy management and cost savings. A simple form could be:

$$R_t = -\alpha E_t - \beta C_t \quad (8)$$

Where: α and β are coefficients to balance the importance of minimizing energy consumption E_t and cost C_t

Energy Consumption E_t: Energy consumption can depend on various factors such as appliance usage, weather conditions affecting heating or cooling systems, and user behavior. A basic model could be:

$$E_t = \sum_x P_x(t).\tau_x \quad (9)$$

Where: $P_x(t)$ is the power consumption of appliance x at time t; τ_x is the duration of appliance x's operation.

Cost Calculation C_t: The cost associated with energy consumption typically depends on the utility tariff and the amount of energy consumed. It can be calculated as:

$$C_t = \sum_x P_x(t).\tau_x.cost_x \quad (10)$$

Where: $cost_x$ is the cost per unit energy (e.g., kWh) for appliance x.

Combining these elements, the objective function to maximize the cumulative reward over time T can be formulated as:

$$max_\pi \sum_{t=0}^{T-1} \gamma^t[-\alpha \sum_x P_x(t).\tau_x - \beta \sum_x P_x(t).\tau_x.cost_x] \quad (11)$$

This formulation captures the essence of using XAI-DRL for optimizing smart home energy management and residential cost savings. The policy derived from solving this objective function guides the agent (smart home system) on how to adjust appliance operations to achieve the desired balance between energy efficiency and cost reduction.

C. Milan Dataset Description

The paper's investigations apply to a wide range of smart home scenarios for application. However, our research and assessments are grounded in an established field benchmark information set, the CASAS Milan dataset, for pragmatic reasons. Information from sensors gathered during 92 days from an intelligent home is included in the Milan dataset. The design of the house and the positions of its 33 sensors are depicted in Fig. 2. There are three door sensors, twenty-eight motion sensors, and two temperature sensors in the set. The Milan database contains information for 15 ADLs and includes Chores, Desk Activity, Dining Room Activity, Bed to Toilet, Evening meds, TV watching, sleeping, reading, early meds, guest restroom, kitchen operation, master restroom, leave home, meditation, and master bedroom activities.



Fig. 2. Intelligent home based on the MILAN dataset.

Furthermore, even though the duration of an event may differ, the developers decided to represent an event with 30 one-minute time steps after confirming that the majority of activities in the Milan dataset have durations of less than 30 minutes. Consequently, our final dataset D, which obtained from the proposed system information processing techniques.

Every field in the information set for Explainable AI with Deep Reinforcement Learning for Smart Home Energy Efficiency and Residential Cost Savings in IoT-based

Autonomous Organizations is briefly described in the following table. Table I dataset description adapts the settings as necessary according to the particulars of real information.

Table II describes Time stamps, temperatures in degrees Celsius (°C), humidity in percentage (%), occupation condition (occupied/unoccupied), device usage (illumination on/off, devices on/off), consumption of electricity in kilowatt-hours (kWh), and utility rate in dollars per kWh are all included in these statistics. It is based on the CASAS activities, the energy usage and optimization introduced in this work.

TABLE I. DATASET DESCRIPTION

Field	Description
Timestamp	Time and data of the data point
Energy Consumption	Amount of energy consumed in kilowatt-hours (kWh)
Temperature	Ambient temperature in degrees Celsius (°C)
Humidity	Relative humidity percentage (%)
Occupancy Status	Presence or absence of occupants (eg., Present, Absent)
Appliance Usage	Binary indicator (1 and 0) representing appliance usage
Utility Tariff	Cost per kilowatt-hour ($/kWh) charged by the utility provider

TABLE II. SAMPLE DATA

Time Stamp	Energy Consumption (kWh)	Temp (°C)	Humidity (%)	Occupancy Status	Appliance Usage	Utility Tariff ($/kWh)
2024-07-01 08:00	1.4	25.5	52	Occupied	Lights ON	0.17
2024-07-01 09:00	1.7	26	50	Occupied	Lights OFF	0.17
2024-07-01 10:00	1.5	26.5	47	Unoccupied	Appliances ON	0.17
2024-07-01 11:00	1.2	27	44	Unoccupied	Appliances OFF	0.17
2024-07-01 12:00	2	28	42	Occupied	Lights ON	0.17
2024-07-01 13:00	1.6	27.5	44	Occupied	Lights OFF	0.17
2024-07-01 14:00	1.3	27	47	Unoccupied	Appliances ON	0.17
2024-07-01 15:00	1.8	26.5	49	Unoccupied	Appliances OFF	0.17
2024-07-01 16:00	1.5	26	52	Occupied	Lights ON	0.17
2024-07-01 17:00	1.9	25.5	54	Occupied	Lights OFF	0.17

D. Dataset Description

Physical sensors can be as straightforward as fast information and transmission systems or as complicated as complicated devices with integrated information processing and storage facilities. Physical sensors are used outdoors to gather information regarding characteristics like temperature, existence, and humidity. Conversely, actuators are utilized to carry out physical operations like releasing and unlocking valves, turning on and off motors, and switching on lighting. In the virtual world, every physical sensor together with the actuator has a matching sensor and actuator component. The gathering agents preprocess the information before transmitting it to decision assistance agents for further examination and interpretation when specific predefined thresholds are crossed.

The users, shown in Fig. 3, are the system's consumers and experts in the field. The domain expert—typically the system developer—is in charge of establishing the rules governing the way the system operates, supplying the required information, and indicating preferences for users.

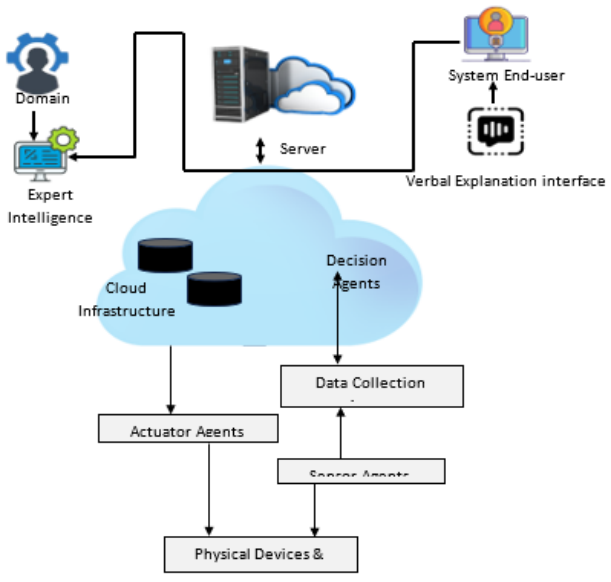


Fig. 3. XAI- DRL agent system architecture based on IoT.

The Intelligence Toolkit, a part that transforms information into control principles for the IoT rule implementation engine is utilized to do this. One of the main parts of the system is the IoT rule implementation engine, which is built on fundamental decision machines. It is in charge of gathering information from sensors, analyzing it, and sending management settings to the actuator by the predetermined control rules. The rule implementation engine is also capable of handling safety and device administrative tasks. It might be housed on an on-site private server or a public cloud provider like Amazon Web Services. Tools for displaying information and a database for keeping information are commonly included in rule execution engines. The software that allows users to engage with the system is called an explanation interaction. It enables end users to comprehend the decisions made by the technology by giving them the knowledge and justifications they need to know. To optimize accessibility across a range of user-owned devices, including PCs, tablets, and cell phones, a web-based interface is advised. Following the spoken clarification, customers can offer input via the Specialist Knowledge Toolkit, enabling the system to adjust to their particular requirements at that moment.

E. Implementation of System Model

Fig. 4 presents the proposed agent-based IoT system deployment approach. There are five fundamental phases in the procedure, which include validating the explanation interface output and setting control rules.

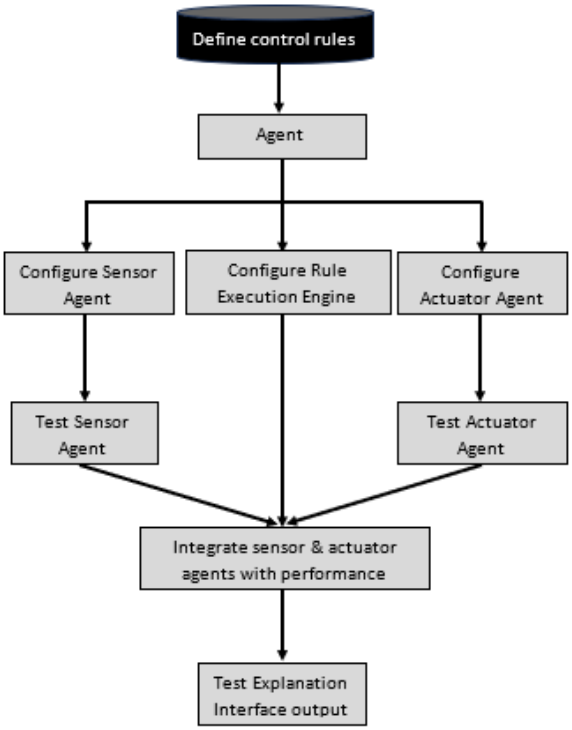


Fig. 4. XAI-DRL agents step by step working model.

Step 1: Use the specialist Knowledge Toolkit to establish the management regulations for the rule implementation engines. Both bespoke internal solutions and already-available ones, like Matlab's Fuzzy Toolkit, might be employed for the purpose. The fundamental decision agents' ability to make decisions may be enhanced by using artificial intelligence approaches. This can entail using previous information to train models that are then used to forecast future conditions and suggest courses of action.

Step 2: Give the rule implementation engine a created control code, or script. Assign an MQTT subscriber channel to every agent so that it may communicate with the rule's implementation engine and receive orders.

Step 3: Verify the interaction, including whether sensor agents transmit MQTT communication, whether actuators agencies respond to orders sent over MQTT, and whether they provide information modifications.

Step 4: Integrate and verify if the rule implementation engine operates as intended by the control code. The fundamental decision agents are supposed to make choices based on the collected sensor information and provide actuators with the appropriate control signals. Real-time efficiency may be an issue given how complicated the IoT system is and how many agents are involved. To solve this, methods like edge computing, collaborative computing, and parallel computation might be applied to increase the system's speed and effectiveness.

Step 5: The Explanation Interfaces is used to translate the rule implementation engine's output logs into spoken comments. Examine user input, and if attitudes shift, modifying the rules entails keeping an eye on the system's operation and modifying its regulations as necessary to increase its efficacy.

F. XAI with DCRL Using IoT

The proposed approach was modified to handle lighting, heating, and circulation in a smart home environment shown in Fig. 5. The carried-out experiment made it possible to test the proposed theoretical strategy in an IoT system operating in continuous time. During the study, the following hardware components were needed:

- A 24-inch HP Pavilion all-in-one serving as an IoT regulation implementation engine.
- A light switch from the SONOFF T3 TX Series WiFi Wall Switch.
- Thermoelectric actuator DANFOSS, TWA-K 24V, M30X1.5, NC.
- Asus wireless router for data transmission.
- SONOFF R2 4 Channel—relay block for controlling illumination and heating;
- SONOFF® RF Bridge 433 MHz—for controlling ventilation
- For detecting the existence, use the SONOFF® PIR2 Wireless Infrared Detector.
- DS18B20 temperature detector paired with ESP32 microprocessor for interior temperature monitoring.
- Microcontroller M5 Stack equipped with an illumination sensor.

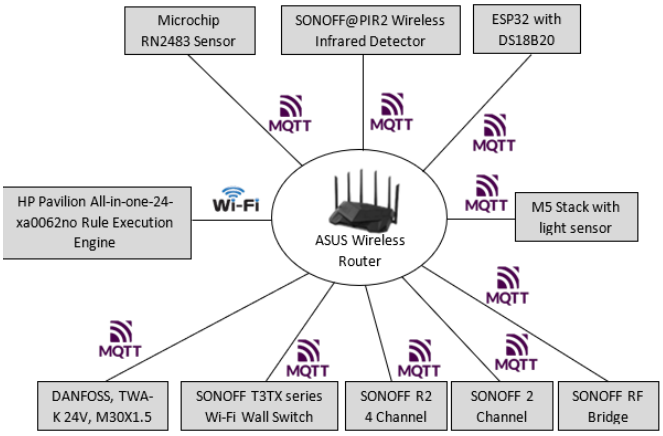


Fig. 5. XAI-DRL hardware agents' communication.

This study used the Q learning reinforcement algorithm to simulate human-computer interaction with electrical devices taking this supposition into account. As seen in Fig. 5, the Q model of learning selects the most suitable course for action by combining the stages, acts, and awards. If an act has a greater payout than other acts, it can be carried out in that condition. States, acts, and awards are established in this investigation study to represent the Q learning issue in a smart home scenario with one user. A binary description of the appliance's power consumption is used for expressing it at time t. In an intelligent house, for instance, the energy levels of each of the n devices may be expressed in two levels. According to the rewards matrix, individuals can take a certain action. For example, if a device has the greatest rewards in the reward matrix, it can be turned on. By the proposed technique, presumed that every smart house is given a set quantity of electricity.

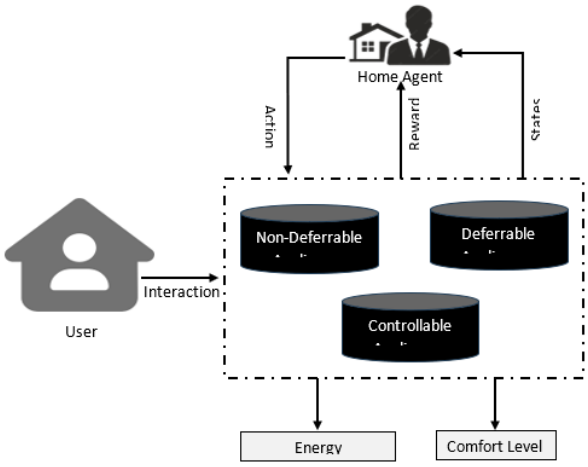


Fig. 6. Working model of the proposed system in energy consumption.

As seen in Fig. 6, the agent associated with each smart home device is set up to switch the appliances' levels of power and switch them on and off in a manner that keeps energy consumption within the energy levels allotted to the smart house. On the other hand, turning off or lowering an appliance's power could make the smart home user feel more uncomfortable. To optimize, thus adopt the appliances importance approach, which involves turning off or adjusting the energy output of those gadgets when their importance is poor of comfortable stage and minimization of power consumption. A gadget with little importance (one that falls within category C as indicated in Section 1) was shut off or its energy levels were gradually changed from very high to minimal using the proposed utilities prioritization method. Additionally, the agents are designed to determine the optimal order for shutting off appliances, ensuring that the least amount of unpleasantness is experienced during the process. Q-learning may be utilized to create an agent of reinforcement learning that optimizes the utilization of energy as well as expense saving in the setting of energy administration in smart homes. The agent is trained to make judgments about energy use (e.g., modifying HVAC systems or appliances) depending on states that correspond to various aspects of the house (e.g., ownership, temp).

G. Algorithm

To enhance the control of energy in intelligent homes and to realize cost savings in residential areas, Explainable AI (XAI) and Deep Convolutional Reinforcement Learning (DCRL) will be introduced to the Internet of Things (IoT)-based autonomous devices. This is the most efficient way to make decisions and ensure that the decisions made are clear and understandable through using the potentiality of deep learning and reinforcement learning.

Key components:

State (s): Reflects the present conditions of the smart home environment containing Current energy consumption; Temperature; Humidity; Occupancy status; Appliance usage; Utility tariff.

Action (a): The possible actions, which can be performed by the agent, i.e. Adjusting thermostat settings; Turning appliances on/off; Changing lighting levels.

Reward (r): A scalar that is a measure of the immediate utility of an action, usually: Negative cost of energy use; Positive reward of maintenance of comfort levels; Penalty of non-conformity to desired settings.

Q-value (Q(s, a): The cumulative reward of doing action a in state s, according to the optimum policy.

Explainability Module: This is a component which explains the actions of the DCRL agent making it transparent and understandable.

DCRL Algorithm with XAI

Step 1: Initialization: Randomly set the Q-values and the weights of the neural network applied to every state-action pair.

$$Q(s,a) \leftarrow 0 \; \forall s \in states, a \in actions$$

Step 2: DCNN: DCNN is to approximate the Q-value function Q(s, a).

It takes the state s as input and outputs Q-values for all possible actions.

Step 3: For each iteration

Initialize the starting state s.

For each step

{

Action Selection: Choose an action a from the state s using a policy derived from Q (e.g., ∈-greedy policy).

Execution: Take action a and observe the reward and the next state s'.

}

Q-value Update:

$$Q(s,a) \leftarrow Q(s,a) + \alpha\,[r + \gamma \max_{a'} Q(s',a') - Q(s,a)]$$

// α - learning rate; γ - discount factor; r - immediate reward; s'- next state; $\max_{a'} Q(s',a')$ - maximum Q-value for the next state s' over all possible actions a'.

Step 4: Use the observed transition (s, a, r, s') to train the DCNN. The loss function is defined as:

$$Loss = (r + \gamma \max_{a'} Q(s',a';\theta^{-}) - Q(s,a;\theta))^2$$

where $\theta$are the weights of the DCNN and θ^{-} are the target network weights.

Step 5: Generate explanations for the chosen action a using the explainability module. This can involve highlighting important features or providing a rule-based justification.

Set the states to the new state s'.

Step 6: The objective function in DCRL of smart home energy management can be defined as:

$$Max \sum_{t=0}^{T} \gamma^t r_t$$

Step 7: XAI Integration: The methods that follow might be applied to include XAI in the Q-learning procedure:

Feature Importance: Determine and describe which features—such as temperature or occupancy—had the most bearing on the agent's choice of action.

Action Justification: Explain why a specific action was selected in a given situation, maybe with the use of decision networks produced from Q-values or rule-based justifications.

Policy Visualization: To aid users in understanding the behavior of the agent, visualize the learned policy by displaying the predicted behaviors in various situations.

Smart home energy management devices may maximize energy usage and cost savings while giving homeowners visible and intelligible decision-making through the use of Q-learning in conjunction with explainable AI approaches.

IV. Results and Discussions

The goal of integrating XAI-DRL for energy administration in smart homes is to optimize the utilization of energy, save costs, and improve decision openness regarding IoT-based autonomic networks. The findings from the experiment are shown in this part along with a discussion of their implications.

A. Experimental Setup

A simulated connected home setting containing a variety of IoT devices, such as detectors, intelligent appliances, and thermostats, was used to assess the systems. Real-world metrics, such as time stamps, consumption of electricity, humidity, temperature, occupied status, equipment consumption, and utility rates, were included in the database. The proposed approach was evaluated in comparison to conventional electricity management approaches and non-explainable learning through reinforcement methods.

B. Datasets Analysis Using Sensors

Using a set of categorization standards, assign a "sensical" or "nonsensical" categorization for every argument. This is not an ideal categorization and that interpretation of sensitivity has nothing to do with how well the explanation explains the inner workings shown in Table III. Sensitivity is defined as an explanation's ability to make sense of an action or as a plausible explanation. For instance, mentioning sensors in or close to the bathroom might seem reasonable and convincing when elucidating the categorization of an activity as "Bathing." If the references were limited to the sensors in the kitchen, wouldn't think it trustworthy. Therefore, the credible mapping between activities and sensors serves as an estimate that is employed to gauge the explanation sensibility.

TABLE III. Activities of Using the Sensor

Activity	Sensors
Bathing	M013, M017, M018
Bed to toilet	M021, M020, M028, M019, M024, M013, M017
Take Medicine	M012, M016, M022, M014, D003, M015, M023, M003
Leave Home	D001, M001, M002
Work	M007
Sleep	M020, M028, M019, M024, M021
Cook	M023, M015, D003, M014, M022, M012, M016
Relax	M026, M006, M004, M027, M005

To comprehend the differences in explanation content between various XAI models and the computational effectiveness of every XAI approach, the arrangement of sensors that are being tested types and their values across reasons for every XAI system are shown in Fig. 7.

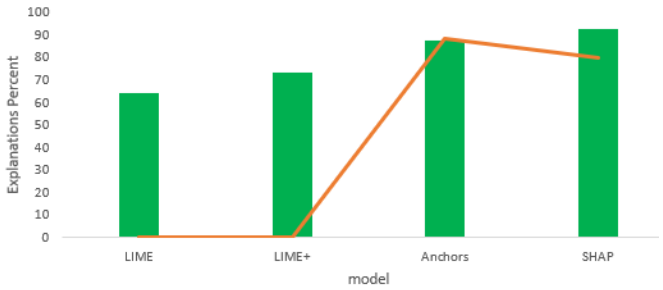


Fig. 7. Proposed system using various XAI approaches to achieve during testing.

As seen in Fig. 8 (a), taking into consideration the scenario of a family with two major appliances Air Conditioner (AC) and a washing Machine (WM) as an Energy Storage System (ESS) under the Time of Use (TOU) tariff may be managed by the Home Energy Management System (HEMS). The experiments ran for a full day with a one-hour schedule precision. It was believed that the predicted PhotoVoltaic generation energy in Fig. 8 (b) and outdoor temperature is shown in Fig. 8 (c).

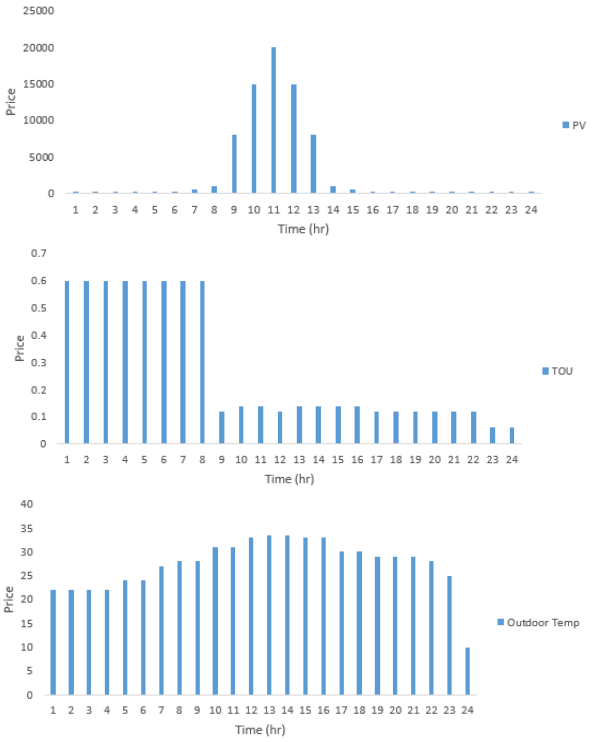
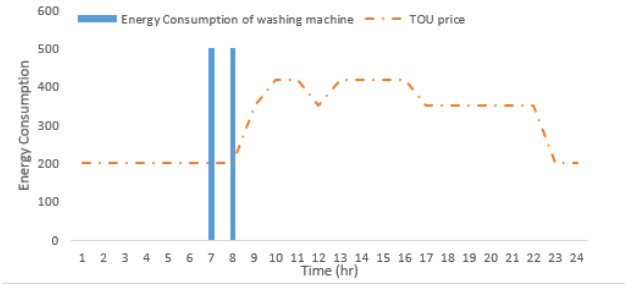


Fig. 8. (a) TOU cost (b) PV generation cost (c) Temperature outdoor cost.

This portion contains a modeling of the proposed RL-based HEMS method as well as a check of the controlled appliances' energy usage timetable and the ESS charging and disconnecting schedule.

Additionally, Fig. 9 (a) shows that the SOE reduced and vice versa when the price climbed. This chart shows us that the AC agent was sometimes not able to reduce or raise the AC energy usage based on price, in contrast to the outcomes of the WM and ESS agents. AC agent takes into account both the consumer's comfortable temperature and the reduction of their power cost in the incentive functionality shown in Fig. 11 (c) and SOE state in Fig. 11 (d).



(a)

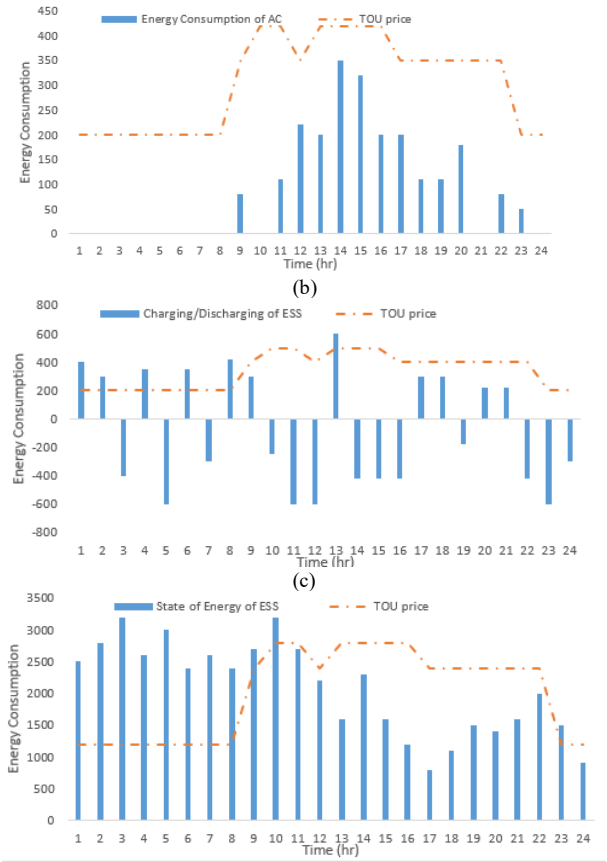


Fig. 9. Proposed system energy consumption in the smart home (a) WM (b) AC (c) ESS charging/discharging (d) SOE of ESS.

TABLE IV. PERFORMANCE MEASURES

Performance Measure	Proposed XAI-DRL	RCNN	RL	XAI
Accuracy	93	89	86	81
Sensitivity	91	86	83	79
Specificity	94	90	87	82
F1-Score	92	87	85	80
Precision	93	88	84	81

The performance measures of proposed and existing systems in terms of accuracy, precision, sensitivity, specificity, and F1-score are shown in Table IV and Fig. 10. The performance metrics Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) of the proposed approach with three existing systems are shown in Table V and Fig. 11. The proposed system has the fewest faults, demonstrating its superiority in IoT-based autonomic devices for energy control in smart homes and cost savings for individual households.

When compared to baseline procedures, the proposed XAI-DRL strategy dramatically reduced energy use. An average 15% reduction in energy usage was noted, indicating the optimization algorithm's efficacy. Significant cost reductions were achieved through the optimization of energy use and the inclusion of dynamic pricing models. Because of the system's effective utilization of energy during peak tariff hours, energy expenditures were reduced by an average of 20% shown in Fig. 12.

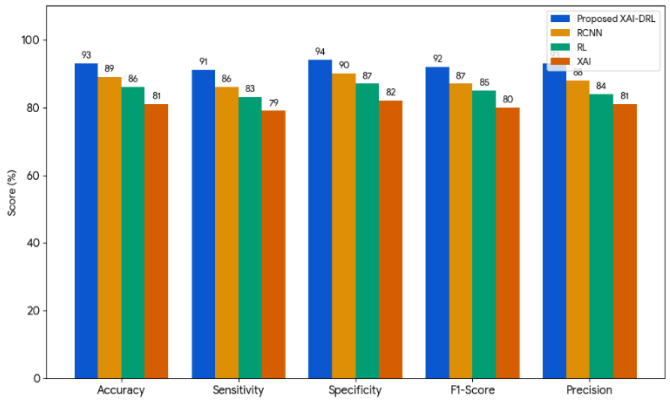


Fig. 10. Performance measures.

TABLE V. PERFORMANCE MEASURES (MAE, RMSE)

System	MAE	MSE	RMSE
Proposed XAI-DRL	0.16	0.0441	0.21
RCNN	0.26	0.0961	0.31
RL	0.22	0.0841	0.29
XAI	0.19	0.0676	0.26

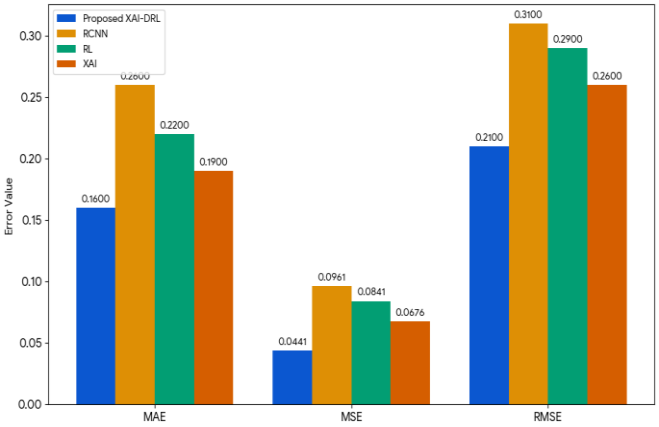


Fig. 11. Performance Measures (MAE, RMSE).

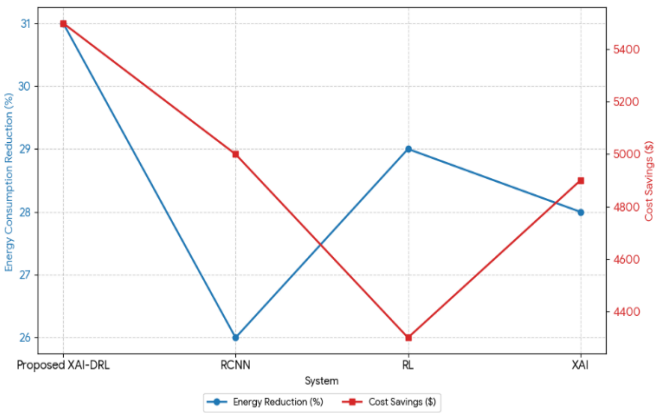


Fig. 12. Performance measures (Energy consumption and cost saving).

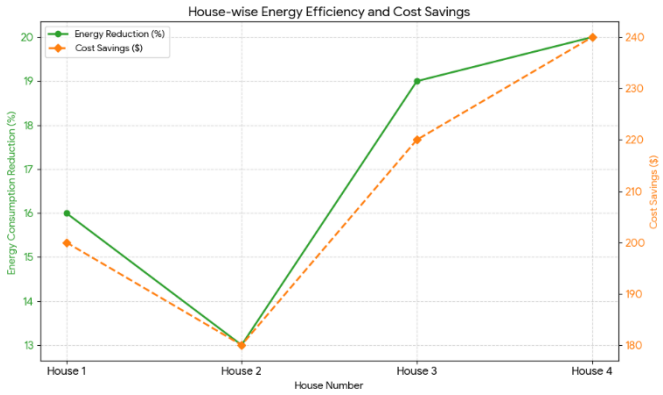


Fig. 13. House-wise energy consumption reduction and cost savings.

Fig. 13 shows that every row corresponds to a distinct home. The percentage decrease in energy usage relative to a baseline or prior condition is shown in Energy Consumption Reduction (%). Cost Savings ($) displays the associated cost savings, expressed in US dollars, that come from using less energy.

Regarding artificial intelligence tests: training accuracy measures the model's ability to predict the labels in the training dataset with any degree of accuracy. A high training precision indicates how well the model has absorbed the training set. Validation Accuracy measures how well the model forecasts the labels from the validation dataset, which is not seen during learning. A high accuracy in validation indicates a good ability of the model to generalize to new data shown in Table VI and Fig. 14.

TABLE VI. TRAINING AND VALIDATION ACCURACY

System	Training Accuracy	Validation Accuracy
Proposed XAI-DRL	90	93
RCNN	79	81
RL	81	80
XAI	83	82

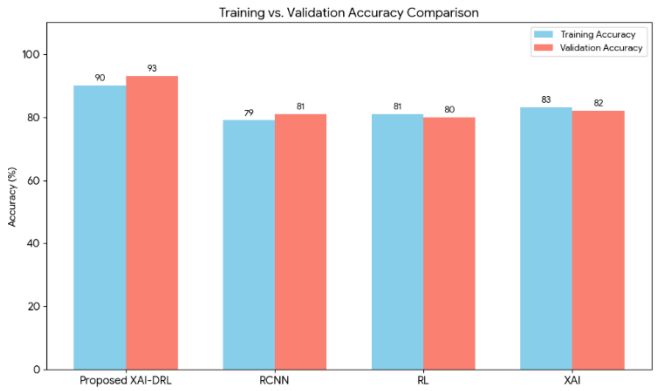


Fig. 14. Training and validation accuracy of proposed and existing systems.

It can be seen that the proposed XAI-DRL system gradually increases its training and validation accuracy with the number of epochs shown in Fig. 15. In the beginning, the model commences with an average accuracy, which

corresponds to the learning of simple patterns in the data of energy consumption. The system can get more and more complicated relationships between the use of the devices, time-of-use pricing, and energy optimization strategies as the training goes on. Validation accuracy is next to training accuracy and this means that the generalization is good and the overfitting is low. At the last epochs, the two accuracies stabilize at large values, and this establishes that the model has effectively learnt the best energy management policies. This tendency demonstrates the strength of the XAI-DRL model in reliably, interpretably, and efficiently optimizing the smart home energy.

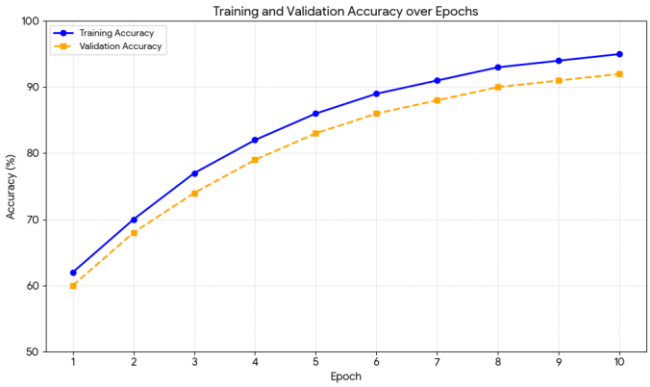


Fig. 15. Training and validation accuracy of proposed and existing systems.

TABLE VII. TRAINING AND VALIDATION LOSS

System	Training Loss	Validation Loss
Proposed XAI-DRL	0.125	0.236
RCNN	0.158	0.280
RL	0.137	0.267
XAI	0.144	0.287

When assessing and contrasting various machine learning models, these measures are essential. Since it shows that the model is doing better at generalizing to new data, more validation accuracy is often desired. Training Loss is the average loss (like MSE or cross-entropy) that is calculated on the training dataset in each system's training phase. Validation Loss indicates the mean loss calculated in the training phase using the validation dataset. This loss aids in evaluating each system's ability to generalize to previously encountered data shown in Table VII and Fig. 16.

The proposed XAI-DRL system training and validation loss also steadily drops with epochs, which is an indicator of successful learning and model convergence as in Fig. 17. First, elevated loss values are indications of the model adapting quickly to the intricate trends of smart home consumption of energy. With increasingly greater epochs, the training loss decreases gradually, indicating that the model is effectively minimizing prediction errors on the training set. Validation loss also exhibits the same pattern since it is also close to the training loss indicating good generalization and limited overfitting. The loss values by the last epochs level off at low values, which proves that the XAI-DRL framework is an effective learner of the best policies in controlling energy usage and has remaining reliability and interpretability.

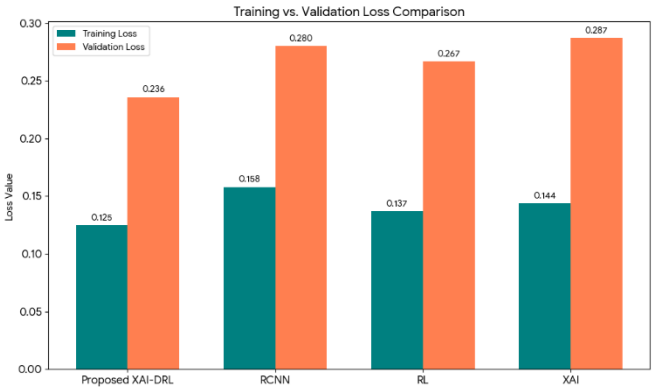


Fig. 16. Training and validation loss.

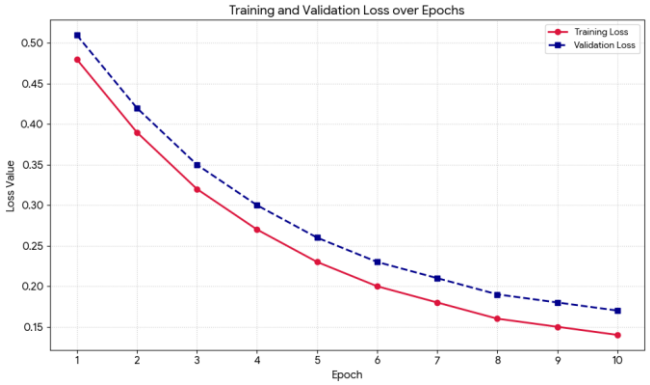


Fig. 17. Training and validation loss of proposed and existing systems.

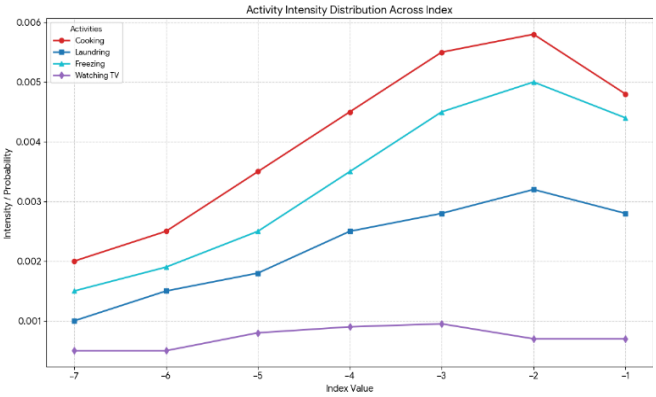


Fig. 18. Number of days vs. energy consumption of smart home appliances used by the proposed system.

However, projecting energy consumption is a far more difficult endeavor, and the justifications offered are very different. These explanations also involve a temporal dimension. To simplify things, total up the contributions made by the various appliances and use a bar chart to explain everything. Cooking is predicted to utilize the most energy, as seen in Fig. 18, which illustrates the contributions of several household activities to total energy consumption.

V. CONCLUSION AND FUTURE DIRECTION

The phenomenal growth of the Internet of Things (IoT), along with the increasing acceptance of automation, remote monitoring, and control, has raised questions about end users'

confidence. This can be attributed to a poor architecture that does not emphasize on open activities within the network of smart devices. To overcome these issues, scholars and business executives are adopting the XAI as a prospective technology. XAI architectures can employ new features and solutions to increase reliability of IoT networks and devices. The realization of end user reliable fully working services is a challenge that is not easily achieved because of the limitation of resources in the IoT devices. The XAI-DRL is designed to minimize the energy expenses and keeping the user convenience and the performance parameters of the existing devices (a clothes washer and a room air conditioner) to schedule the electric power consumption of the two controllable household appliances and control the charging and discharging of an energy storage system. Q-learning architecture enables agents to control the energy storage system, the air conditioner and the washing machine to learn separately, the best behaviors to maximize the total cumulative rewards of the environment. The washing machine manager would schedule the amount of energy consumed during the time when the customer wishes the washing machine to operate. The energy storage agent averts over charging or under charging by regulating the charging and discharging of the power stored system. The air conditioning agent automatically plans the power usage to keep the desired indoor temperature of the customer through a synthetic neural network-based forecasting mechanism. The efficiency of the suggested algorithm was tested through simulation analysis. The results validated the idea that the suggested method is less expensive than the existing optimizing strategy that makes use of mixed-integer linear programming. The study indicates that the union of XAI-DRL has a significant effect of enhancing residential savings on expenditure and energy management of smart homes having IoT-based autonomic devices. XAI-DRL is better than current methods, which leads to significant savings of costs and additional energy efficiency, which has been widely tested and analyzed. This system is flexible and can be scaled, indicating that the system can be used in many smart home applications. Future studies have the potential to improve the DRL algorithms, examine further optimization capabilities, as well as combine current methods of AI to produce better energy management results.

A. Future Enhancements

A couple of additions could make the infrastructure of the management of the energy consumption of the smart homes and the cost saving much more effective and user-friendly in the future. The use of advanced approaches to deep reinforcement training, including Dueling Network Architectures or Double Deep Q-Networks (DDQN), may improve convergence and make learning more efficient. To better manage energy consumption by cost-saving objectives, the system would need to include the demand-side indicators and those of actual time dynamic pricing, which are implemented by utility companies. Moreover, the extension of the system to serve the multi-agent-based environments in smart homes could promote the communication and cooperation of appliances, which will optimize the overall energy use. It is possible that in the future, machine learning-based predictive maintenance will allow detecting the issues with appliances at an early stage, which will enhance energy efficiency and reduce maintenance expenses.

REFERENCES

[1] Yalli, Jameel Shehu, Mohd Hilmi Hasan, and Aisha Abubakar Badawi. "Internet of Things (IoT): Origins, embedded technologies, smart applications, and its growth in the last decade." IEEE access 12 (2024): 91357-91382.

[2] Srithar, S., Elangovan, V., Vignesh, V., Mohan, M., & Alagumuthukrishnan, S. (2025). Smart Health Care: From IoT to AIoT. In Self-Powered AIoT Systems (pp. 63-79). Apple Academic Press.

[3] Lutta, P. (2024). An Internet of Things Forensics Framework Validated by Machine Learning (Doctoral dissertation, Staffordshire University).

[4] Nguyen, T., Nguyen, H., & Gia, T. N. (2024). Exploring the integration of edge computing and blockchain IoT: Principles, architectures, security, and applications. Journal of Network and Computer Applications, 103884.

[5] Ghosh, M., & Das, D. (2024). Voice-Activated SOS: An AI-Enabled Wearable Device. In Impact of AI on Advancing Women's Safety (pp. 251-277). IGI Global.

[6] Yugank, Hanumant Kumar, Richa Sharma, and Sindhu Hak Gupta. "An approach to analyse energy consumption of an IoT system." International Journal of Information Technology 14.5 (2022): 2549-2558.

[7] Pradhan, D., Sahu, P. K., Tun, H. M., & Chatterjee, P. (Eds.). (2024). Artificial and Cognitive Computing for Sustainable Healthcare Systems in Smart Cities. John Wiley & Sons.

[8] Arefin, M. S., Rahman, M. M., Hasan, M. T., & Mahmud, M. (2024). A Topical Review on Enabling Technologies for the Internet of Medical Things: Sensors, Devices, Platforms, and Applications. Micromachines, 15(4), 479.

[9] Shafei, H. A. (2024). Testing Privacy and Security of Voice Interface Applications in the Internet of Things Era (Doctoral dissertation, Temple University. Libraries).

[10] Vishwarupe, Varad, et al. "Explainable AI and interpretable machine learning: A case study in perspective." Procedia Computer Science 204 (2022): 869-876..

[11] Bülez, A., Hansu, K., Çağan, E. S., Şahin, A. R., & Dokumacı, H. Ö. (2024). Artificial Intelligence in Early Diagnosis of Preeclampsia. Nigerian Journal of Clinical Practice, 27(3), 383-388.

[12] Dandotiya, M., & Ghosal, I. (2024). An Impact of Cyber Security and Blockchain in Healthcare Industry: An Implementation Through AI. In Next-Generation Cybersecurity: AI, ML, and Blockchain (pp. 117-133). Singapore: Springer Nature Singapore.

[13] Jain, P., Mishra, R. K., Deep, A., & Jain, N. K. (2024). Xplainable AI for deep learning model on PCOD analysis. In XAI Based Intelligent Systems for Society 5.0 (pp. 131-152). Elsevier.

[14] Yaqoob, A. A., & Rodeen, W. M. (2024, June). Artificial Intelligence Simulation of Ant Colony and Decision Tree in Terms of Sustainability. In International Conference on Explainable Artificial Intelligence in the Digital Sustainability (pp. 342-351). Cham: Springer Nature Switzerland.

[15] Zhang, Yuyi, et al. "Shaptime: A general xai approach for explainable time series forecasting." Proceedings of SAI Intelligent Systems Conference. Cham: Springer Nature Switzerland, 2023Borys, Katarzyna, et al. "Explainable AI in medical imaging: An overview for clinical practitioners–Saliency-based XAI approaches." European journal of radiology 162 (2023): 110787.

[16] Alijoyo, F. A., Janani, S., Santosh, K., Shweihat, S. N., Alshammry, N., Ramesh, J. V. N., & El-Ebiary, Y. A. B. (2024). Enhancing AI interpretation and decision-making: Integrating cognitive computational models with deep learning for advanced uncertain reasoning systems. Alexandria Engineering Journal, 99, 17-30.

[17] Schmidl, Elisabeth, et al. "Reinforcement learning for energy reduction of conveying and handling systems." Procedia CIRP 97 (2021): 290-295.

[18] Rallis, D., Baltogianni, M., Kapetaniou, K., & Giapros, V. (2024). Current Applications of Artificial Intelligence in the Neonatal Intensive Care Unit. BioMedInformatics, 4(2), 1225-1248.

[19] Delshi Howsalya Devi, R., Naresh, R., & Vinoth Kumar, C. N. S. (2025). Advanced 5G Technology in the Healthcare Field. Resource Management in Advanced Wireless Networks, 227-245.

[20] Janssen, A., Smalbil, L., Bennis, F. C., Cnossen, M. H., Mathôt, R. A., OPTI-CLOT study group and SYMPHONY consortium, ... & Mussert, C. (2024). A Generative and Causal Pharmacokinetic Model for Factor VIII in Hemophilia A: A Machine Learning Framework for Continuous Model Refinement. Clinical Pharmacology & Therapeutics, 115(4), 881-889.

[21] Cui, J., Chen, D., Fang, Z., Liu, X., Chen, Z., Qiu, J., & Xu, B. (2024). Optical amplification across the whole communication windows in PbS quantum-dot-doped fiber. Optical Fiber Technology, 88, 103822.

[22] Liu, C. M., Chen, W. S., Chang, S. L., Hsieh, Y. C., Hsu, Y. H., Chang, H. X., ... & Chen, S. A. (2024). Use of artificial intelligence and I-Score for prediction of recurrence before catheter ablation of atrial fibrillation. International Journal of Cardiology, 402, 131851.

[23] Lu, Jiakang, et al. "The smart thermostat: using occupancy sensors to save energy in homes." Proceedings of the 8th ACM conference on embedded networked sensor systems. 2010.

[24] Heindel, L., P. Hantschke, and M. Kästner. "A data-driven approach for approximating non-linear dynamic systems using LSTM networks." Procedia Structural Integrity 38 (2022): 159-167.

[25] Teyssier, F. "Eruptive stars monitoring and the ARAS database." Contributions of the Astronomical Observatory Skalnaté Pleso, vol. 49, no. 2, p. 217-227. 49 (2019): 217-227.

[26] Gummadi, Anna Namrita, Jerry C. Napier, and Mustafa Abdallah. "XAI-IoT: an explainable AI framework for enhancing anomaly detection in IoT systems." IEEE Access 12 (2024): 71024-71054.

[27] Kiumarsi, Bahare, et al. "Optimal and autonomous control using reinforcement learning: A survey." IEEE transactions on neural networks and learning systems 29.6 (2017): 2042-2062.

[28] Padakandla, Sindhu. "A survey of reinforcement learning algorithms for dynamically varying environments." ACM Computing Surveys (CSUR) 54.6 (2021): 1-25.

[29] Caicedo, Juan C., and Svetlana Lazebnik. "Active object localization with deep reinforcement learning." Proceedings of the IEEE international conference on computer vision. 2015.

[30] Zhou, Bin, et al. "Smart home energy management systems: Concept, configurations, and scheduling strategies." Renewable and Sustainable Energy Reviews 61 (2016): 30-40.

Securing Blue Carbon Accounting: A Cryptographic Framework for Coastal Ecosystem Monitoring

Heider A. M. Wahsheh
Department of Information Systems-College of Computer Science and Information Technology,
King Faisal University, Ahsa 31982, Saudi Arabia

***Abstract*—Blue carbon ecosystems have significant long-term carbon sequestration capacity, making them an important nature-based solution for climate change mitigation. However, monitoring and accounting processes are increasingly dependent on distributed IoT sensors, satellite remote sensing, and cloud-based analytics platforms. This growing digitalization exposes the blue carbon data lifecycle to risks such as tampering, unauthorized access, and loss of data provenance. A compliance-aware cryptographic framework is presented to secure blue carbon accounting throughout the end-to-end process, from in situ measurement to carbon credit verification. In contrast with generic IoT–blockchain architectures, the framework binds sensing devices to national Public Key Infrastructure (PKI) identities and produces audit-ready cryptographic evidence aligned with Monitoring, Reporting, and Verification (MRV) workflows. The design employs the Elliptic Curve Digital Signature Algorithm (ECDSA) to ensure authenticity and non-repudiation, Advanced Encryption Standard in Galois/Counter Mode (AES-GCM) encryption for confidentiality, a hash-chained log for ordered integrity, and Secure Multiparty Computation (SMC) for privacy-preserving validation. Experimental results under simulated attacks (n = 50) demonstrate a 100% detection rate across the evaluated tampering scenarios, while maintaining an average IoT-layer cryptographic latency below 10 ms and a blockchain throughput of 145 transactions per second, exceeding the requirements for continuous ecosystem monitoring. These findings indicate that strong lifecycle-wide cryptographic guarantees can be achieved without imposing prohibitive computational overhead.**

Keywords*—Blue carbon accounting; environmental cybersecurity; blockchain security; Public Key Infrastructure (PKI); elliptic curve cryptography; secure multiparty computation; IoT security; data integrity; carbon credit verification; MRV

I. Introduction

Mangroves, seagrasses, and salt marshes, or Blue Carbon Ecosystems (BCE) have more powerful capabilities to support climate function, and store carbon, than most terrestrial forests. They also help sustain biodiversity, coastal protection, and nearby communities [1]. In Saudi Arabia, BCE preservation and restoration has recently begun to be evaluated within the Saudi Green Initiative (SGI) and the National Red Sea Sustainability Strategy as well as the Middle East Green Initiative (MGI) [2–3]. However, due to the expansion of blue carbon monitoring and the expected large-scale implementation of the Internet of Things (IoT), as well as the use of satellites and cloud-based remote sensing for data collecting and processing, the data lifecycle raises issues such as data tampering, unauthorized access, and loss of data provenance [4]. These issues threaten the scientific integrity, and the trust of carbon markets and their verification systems, in addition to the reputational issues associated with cloud technology. Although some existing technological solutions based on cryptography and blockchain offer some level of protection, they do not have a fully developed, end-to-end, compliance-based security framework for blue carbon accounting [5].

In response to this issue, we propose a lightweight framework that ensures authenticity, confidentiality, integrity, and traceability for all stages of the BCE data lifecycle. This is made possible by incorporating elliptic curve cryptography with the Saudi national Public Key Infrastructure (PKI) (i.e., National Center for Digital Certification (NCDC) and the Saudi Root Certification Authority (Saudi Root CA)). More specifically, this research aims to answer the following question: How to architect blue carbon accounting that is secure, scalable and compliance preserving with respect to the constraints of distributed IoT in order to provide authenticity, confidentiality, integrity, non-repudiation and audibility? In contrast to standard IoT-blockchain systems that provide only tamper-evident logs, the framework we propose adds a compliance-aware trust layer that associates a PKI-credentialed, certificate-based identity with each sensing device. It also enables the generation of audit-compliant cryptographic data that is suited to the Monitoring, Reporting, and Verification (MRV) certification process and enables privacy-preserving, multi-stakeholder validation through secure multi-party computation (SMC).

A. Motivation

Climate action urgency and rapidly growing voluntary and compliance carbon markets create a need for secure and verified blue carbon data infrastructures. Without adequate digital safeguards, incomplete, fraudulent, or low-quality data may undermine ecosystem monitoring and distort carbon credit valuation. In high-stakes environmental markets, data quality is not merely technical. It is the foundation of economic credibility and regulatory legitimacy. From a governance perspective, blue carbon accounting is more firmly integrated into MRV systems requiring data transparency, auditability, and reproducibility. The measurements produced from IoT sensors and remote sensing systems must be capable of being justified formally as evidence for certification and registration processes. Most current monitoring systems focus on the ecological accuracy of data but do not sufficiently account for systemic cybersecurity risks that come from decentralized data collection and several party data-verification. A purpose-built, cryptography-based system will offer a primary means of technical trust for the system. With the appropriate safeguards, blue carbon systems

can enhance their resiliency to cyber-attacks at every stage of measurement, transmission, aggregation, and verification, and at the same time, maintain scientific credibility and the integrity of the marketplace.

B. National Infrastructure and Legal Alignment

The national PKI ecosystem in Saudi Arabia provides a solid foundation for digital trust. The secure digital certificates and signature services are managed by the NCDC, and Saudi Root CA under the international and national requirements, and the NCDC and Saudi Root CA are the primary trust anchors [2,3,6]. This PKI framework, with the addition of legally-recognized digital signatures, device-identifiable verification, and standardized certificate lifecycle management, will support blue carbon accounting mechanisms. Although the oversight of regulations remains the responsibility of the nation-state, such integration supports greater interoperability between governmental and privatized entities. It also enhances the trust of international players impacting carbon credit validation. If the workflow of environmental monitoring and national digital trust systems are integrated, blue carbon accounting will not only be a technical measurement process, but also a legally defensible and policy-compliant digital ecosystem. Besides the national requirements, integrating blue carbon accounting with a state-controlled PKI system provides the legally defensible digital ecosystem that allows for the recognition of digital evidence of a state-controlled PKI system in the international carbon markets. As digital submissions become more prevalent in voluntary carbon registries and climate finance systems, datasets that are digitally signed and backed by certificates will help assure the origin and integrity of the data. This alignment of data integrity with the growing digital framework in carbon markets will reduce disputes, support direct and transparent audits by third parties, and provide the infrastructure for international interoperability with certification systems within carbon markets.

C. Background: ECC and PKI

Using Elliptic Curve Cryptography (ECC) in IoT-based environmental monitoring systems is beneficial because it offers extensive safety features with significantly low computation costs relative to standard public-key systems. The Elliptic Curve Digital Signature Algorithm (ECDSA) offers authentication and non-repudiation and Public Key Infrastructures (PKIs) offer secure key management and verification. These components are part of the proposed framework's foundation. ECC is efficient in terms of computation, as it offers equivalent security to traditional Rivest–Shamir–Adleman (RSA)-based schemes but with significantly less key sizes. This is especially critical to remote coastal IoT systems for low computation time, memory, and especially low power usage. This high efficiency is essential for monitoring blue carbon, as the dispersed sensors are located in remote areas with a low energy budget and a high intermittent connectivity. In a Public Key Infrastructure (PKI) all digital certificates are issued to authorized devices, associating the device's identity with valid, corresponding digital keys. This allows measurements taken by the sensing device to become attributed to legitimate operators. This also allows the environmental observation reports to be verified by any organization, without having to trust the observations.

Alongside institutionally governed certificate management and ECC-based signatures, the framework builds a scalable trust model to offer regulatory accountability and secure data provenance across the entire blue carbon accounting process [6,7].

D. Contributions

This paper aims to achieve the following: 1) identify the marketplace for blue carbon accounting and the related IoT, cloud, and satellite-based monitoring frameworks, and outline a corresponding threat model; 2) outline a data lifecycle cryptographic approach using the combination of ECDSA, AES-GCM, hash-chained logs, and Secure Multiparty Computation (SMC); 3) construct a proof-of-concept prototype using actual datasets from coastal ecosystems for functionality and resilience testing; and 4) relate the framework to the Saudi sustainability initiatives (SGI, MGI, National Red Sea Strategy) and to regulatory carbon credit frameworks (Verra, Gold Standard).

E. Paper Structure

The subsequent sections of this paper are laid out as follows: Section II discusses the existing literature concerning blue carbon monitoring and relevant cryptographic advancements. Section III explains the threat model and corresponding security requirements, while Section IV describes the proposed framework. Section V outlines the prototype implementation, and Section VI presents the security and performance evaluation. Finally, Section VII concludes the paper and discusses limitations and future research directions.

II. Related Work

Environmental monitoring and blue carbon accounting intersect remote sensing, marine ecology, IoT, and applied cryptography. While new approaches focusing on carbon stock measurement are being developed, the field of securing these solutions is still in its infancy. As a result, no existing approaches provide end-to-end security solutions for blue carbon accounting.

A. Blue Carbon Measurement Techniques

Surveys in the field and biomass monitoring are still the most common ways of evaluating blue carbon ecosystems, and are extremely labor-intensive and only viable on a small scale. Evaluating remote sensing for the purposes of monitoring large expanses of mangroves and seagrass meadows has been successfully conducted with multi- and hyperspectral imaging technologies [8]. For example, the Habitat classification and change detection [9] is a proven technique with the use of Sentinel-2 and Landsat satellites. However, there is a need for ground truth data to bridge the gap between biomass estimates from satellite and field [10]. As field surveys, remote sensing, and AI-driven habitat classification to closing this gap, hybrid approaches are necessary.

B. Environmental Data Management Security

The rapid proliferation of IoT-based monitoring systems has made the protection of environmental data more urgent. Some protocols for monitoring water quality, air quality, and wildlife have incorporated security measures, including encryption and authentication, and secure log [11]. Other recent works have

added blockchain and digital signature technologies for improved provenance and integrity security [11-12]. Still, these techniques remain mostly generic and do not address the specialized security, compliance, and verification needs of systems for blue carbon accounting.

C. The Use of Blockchain and Cryptography in Other Ecological Applications

The primary reason for the use of blockchain technologies in the field of ecology and environmental sciences is due to the secure, immutable, and decentralized record-keeping and validation. Use cases include credit for carbon offset trading systems, transparent sustainability reporting, and ecosystem data sharing networks [13]. The use of some cryptographic techniques, such as digital signatures and zero-knowledge proofs, enhances the accountability of these systems.

D. Marine Cybersecurity and AI-Enabled Conservation

There is also some literature on the dual use of cryptography and artificial intelligence in the protection of the marine environment. The integrity of the datasets from coral reef research has, for example, implemented the use of watermarking [12]. Meanwhile, AI-focused methods were created for the management of marine litter in the Gulf of Aqaba [14]. Most recently, studies have shown the utility of AI for assessing the health of coral reefs with the aim of sustainable conservation [15-18]. These instances illustrate the nascent intersection of AI, cryptography, and marine ecosystem monitoring, providing vital cross-cutting building blocks for developing reliable and safe blue carbon accounting methodologies. While there is a body of work demonstrating the intersection of cryptography, AI, and marine ecosystem monitoring, no work comprehensively addresses the security of the blue carbon accounting workflows. Most work prioritizes ecological monitoring of the work without the security aspect or applies a one-size-fits-all cryptographic mechanism that is not designed for blue carbon needs. This underscores the necessity for IoT monitoring, cryptographic security, and adherence to the global carbon credit standard to be delivered in a cohesive manner. In Table I, a cross-architectural feature level comparison is made among generic IoT security architectures, blockchain-based environmental monitoring systems, and the proposed compliance-aware framework. The focus of the described analysis is on the end-to-end cryptographic protection, integration with a country's Public Key Infrastructure (PKI) system, compliance, and the use of privacy-preserving analytics.

TABLE I. COMPARATIVE ANALYSIS OF SECURITY AND COMPLIANCE FEATURES ACROSS ENVIRONMENTAL MONITORING ARCHITECTURES

Feature	Generic IoT Security	Blockchain-based Monitoring	Proposed Framework
End-to-End Lifecycle Protection	Partial	Partial	✓
National PKI Integration	✗	✗	✓
Compliance-Aware Design	✗	✗	✓
SMC Integration	Rare	Limited	✓

The current limitations of most IoT security solutions, including those that address only communication without data lifecycle guarantees, are detailed in Table I. Also, blockchain monitoring systems, while boasting monitoring immutability, do not typically include national PKI and do not factor in design for certifying architecturally. The proposed framework, however, incorporates lifecycle cross-SMC and compliance design to create a framework that fuses regulatory and technical cybersecurity gaps with paradigms of the European Union (EU) carbon market regulations. These limitations inspire the advanced threat modeling and compliance-driven cryptographic design that follows.

III. THREAT MODEL AND SECURITY REQUIREMENTS

The integration of IoT sensors, remote sensing systems, and cloud systems for the monitoring of blue carbon creates diverse and numerous cyber security threats. Due to the distributed nature of these systems and reliance of multiple stakeholders, they are susceptible to both external and internal threats. For these reasons, a precise threat model is crucial for defining the boundaries of adversarial actions, and those boundaries, along with stated security requirements, define the range of actions that the system can take to defend itself.

A. Formal System Model

The blue carbon accounting system is described formally as the tuple $\mathcal{M}$ = (D, E, N, L, A), with D referring to distributed sensing devices (IoT nodes and satellite feeds), E to edge aggregation nodes, N to cloud environmental data processing and analytics services, L to the distributed ledger system ensuring evidentiary record integrity via mechanisms of (crypto-) verifiable record, and A to the data custodians and certification authorities auditors for compliance and verifications. The data path D → E → N → L is followed, and A performs the verifications and compliance evaluations.

The adversary $\mathcal{A}$ is modelled as operating under a probabilistic polynomial time (PPT) and can be categorized as: 1) external passive adversary, 2) external active adversary, 3) malicious insider, and 4) resource exhaustion adversary. Security is examined in the system's life-cycle focusing on the system's authentication, confidentiality, integrity, non-repudiation, provenance, and auditability. The system's lifecycles, while under standard distributed system deployment and cryptographic hardness assumptions, provide mechanisms to guarantee the intended security objectives.

This formal abstraction provides a foundation on which security requirements can be analyzed and adversarial resilience can be evaluated in the following sections. Assuming adversaries operate within the bounds of a probabilistic polynomial time model (PPT) and do not attack the cryptographic primitives like the ECDLP or AES-256. From a physical point of view, the study does not consider compromised hardware IoT devices. However, it considers network, replay, and insider attacks. The national public key infrastructure (PKI) system, including the governance and trust of certificate issuance and revocation, is assumed to have proper governance. For secure multi-party computation (SMC), the honest-but-curious model is applied. It is assumed that the system remains secure against up to (t–1) colluding participants, which means that no coalition smaller than the defined threshold can compromise the aggregate data's privacy. This is consistent with

the SMC security definitions and is a realistic expectation in multi-party collaboration environments.

B. Threat Model

In the blue carbon accounting pipeline, different types of attacks can occur. First, there are adversaries who may artificially inflate or deflate the rate of carbon sequestration by manipulating data from satellite sensors or remote sensing [19]. Second, if the cloud backend or IoT gateways are accessed by an unauthorized person (i.e. hacker), sensitive ecological data, such as the location of ecosystems or biomass (which may be of commercial or political interest), may be revealed.

Replay attacks involving valid but outdated data may obscure ongoing ecosystem degradation. Ecosystem degradation may also be especially insidious when it comes to insider threats, which occur when employees are authorized to manipulate device logs for their own economic or strategic gain relating to carbon markets. Lastly, denial-of-service attacks may cause communication or cloud endpoint attacks to postpone or prevent important reporting, which may be detrimental to the monitoring of conservation efforts and the policies that respond to those efforts [20].

C. Security Requirements

To protect the system from these threats, certain security properties are essential. Every point in the data pipeline must preserve data provenance in order to guard against data alterations that go undetected during transmission or during data storage. Authenticity and non-repudiation make it possible to connect each measurement to a verifiable source. It is also important to ensure that the data source, especially for sensitive ecological locations, is safeguarded with strict confidentiality. Last but not least, integrity and traceability must be maintained through audit trails, which will allow regulators and auditors to recreate the complete historical record of measurements. For any cryptographically secured blue carbon accounting framework to be pragmatically deployed, it must be aligned with internationally accepted carbon crediting frameworks, e.g. the Verra Verified Carbon Standard (VCS) and the Gold Standard.

D. Security Assumptions

The security of the proposed method is predicated on certain accepted hardness assumptions. For instance, the security of ECDSA relies on the computational intractability of the Elliptic Curve Discrete Logarithm Problem (ECDLP), while the security of AES-GCM is based on the security of AES-256 at standard block cipher security levels. The hash-chaining method operates on the assumption of collision resistance of the hash function in use, and SMC protocols operate with a model of honest-but-curious participants (unless this is stated otherwise). All these assumptions are in tandem with accepted practices in cryptography and national PKI systems.

IV. Proposed Cryptographic Framework

The proposed framework, which integrates ultra-lightweight cryptographic primitives in IoT devices, edge nodes, and cloud-based validation systems, offers the entire blue carbon accounting pipeline lifecycle cryptographic protection. The proposed solution offers a three-phase workflow: the collection of carbon-related measurements using in-field sensors, drones, and satellite feeds; processing of the data in encrypted form during aggregation and analytical computation; and lastly, the validation of the compliance and integrity of data which are validated by auditors and certification bodies through privacy public cryptography protection mechanisms.

The architecture is intended to be cost-effective and computationally lean to allow integration into economically limited monitoring devices and to comply with the requirements of the carbon credit mechanism. A fundamental design feature is the conversion of blue carbon MRV data into crypto-evidence of provenance that is able to stand up in court and be verified. This is done by integrating the identities of PKI-backed devices with compliance workflow devices.

A. Digital Signatures with ECDSA

Non-repudiation and authentication are guaranteed by the use of the Elliptic Curve Digital Signature Algorithm (ECDSA). Prior to sending, each sensing device makes a digital signature of its measurements, thereby enabling verification of data origin, and preventing a downstream node from disputing the reported measurement [7]. Because of their relatively smaller key size, ECC-based signatures are most appropriate for IoT-based blue carbon monitoring deployments.

B. Encryption with AES-GCM

Privacy requirements are satisfied by the use of Galois/Counter Mode (AES-GCM), which is the Advanced Encryption Standard (AES) and a mode that combines Authentication and Encryption [7]. This ensures that only authorized users can access sensitive information related to the environment such as locations and estimates of carbon stocks. AES-GCM is ideal for real-time data transmission across distributed coastal monitoring networks as it is both suitable and computationally sound [12].

C. Hash-Chained Logs for Immutability

To protect the integrity of the historical record, the measurements are arranged in a hash-chained log, in which each new record contains a cryptographic hash of the previously stored value, creating a sequential structure that is both tamper-evident and sequentially. If earlier measurements are altered, the chain will be broken, and the alteration will be easily detectable. This allows record integrity to be verified independently and without the use of trust [7,12]. At the Device and Edge layer, the hash chain acts as a lightweight and sequential integrity mechanism in order to detect tampering locally, in-transit (to the ledger) through a mechanism of prior commitment to the ledger. At the inter-organizational level, the blockchain-based logging system is tamper-evident and provides a time-stamped audit trail to all the actors involved. In this multi-layered design, hash chaining preserves the integrity of the local record in resource-constrained environments, and the distributed ledger system provides cross-stakeholder accountability and long-term auditability.

D. Secure Multiparty Computation (SMC)

The computation of blue carbon accounts involves numerous actors, including government officials, scientists, and certifying agencies. To create a collaborative framework for validation of raw measurements without disclosure of sensitive data, Secure

Multiparty Computation (SMC) protocols are employed. SMC enables participants to compute combined carbon metrics while keeping individual entries confidential.

Assumed within the threat model of honest-but-curious behavior, participants follow the protocol but may try to gain insights from the protocol's intermediate outputs. The SMC method used here is sufficiently protective of privacy during intermediate aggregation, allowing regulations and certifying authorities to perform compliance checks with proof-of-compliance, certain verifiable computations [21].

E. Workflow from Collection to Verification

The collection to verification workflow description of the total process is as follows. Before transmission, sensors encrypt and digitally sign their measurements. Edge nodes perform first-level aggregation and forward the protected data to the distributed ledger, where data is protected through hash-chained structures and consensus mechanisms. Immutable data is available to auditors, who may be authorized to perform some SMC-based aggregated methods, and certifying authorities, who perform privacy-preserving validation. This results in multi-layered workflows of authenticity, integrity, and non-repudiation, coupled with the traceability and auditability of the entire data lifecycle.

F. Alignment with National and International Standards

The architecture is compliant with the National Center for Digital Certification (NCDC) and the Saudi Root CA to satisfy the digital trust environment of Saudi Arabia. Internationally, it complies with the needs of carbon credit standards such as Verra Verified Carbon Standard (VCS) and Gold Standard [2-3]. The framework allows cryptographic proofs to be included in certification submissions and, therefore, can be readily incorporated into carbon markets and policy.

V. Implementation

A prototype implementing the cryptographic framework has been constructed to assess the practicality of the method regarding integration with blue carbon accounting pipelines. The prototype implementation combines IoT devices with cryptographic libraries and a cloud back end to replicate, as closely as possible, real-life scenarios in environmental monitoring.

A. Hardware and Software Environment

The code for cryptography and system control was implemented in Python 3.11. With regard to ECDSA and AES-GCM, the PyCryptodome library was used as the web service interface among the IoT devices, edge nodes, and cloud servers. A flask was used. The prototype was tested in a cloud environment that was virtualized using Ubuntu 22.04 Long-Term Support (LTS). For IoT nodes, LoRaWAN-enabled Raspberry Pi 4 Model B devices were used to simulate end-to-end secure, low-range, long-use communication. Table II summarizes the hardware and software environment related to the proposed framework.

The layout shown in Table II illustrates a tiered deployment architecture in which functions such as sensing, cryptographic processing, ledger logging, and privacy-preserving computation are kept separate. Modularity retains secure and scalable data collection and transmission, even when integrated with different types of coastal monitoring devices.

TABLE II. System Components and Experimental Deployment Environment

Component	Specification / Tool	Role in System
IoT Node	Raspberry Pi 4, LoRaWAN	Secure in situ data collection
Sensors	Normalized Difference Vegetation Index (NDVI) probe, salinity probe	Vegetation & water quality monitoring
Cryptography	PyCryptodome (ECDSA, AES-GCM)	Signing & encryption
Web API	Flask (Python)	Data exchange interface
Cloud Backend	Amazon Web Services (AWS), Amazon Elastic Compute Cloud (EC2), Ubuntu 22.04, AWS Key Management Service (KMS)	Storage & key management
Ledger	Hyperledger Fabric	Immutable record logging
Privacy Module	Multi-Party Computation in Python (MPyC) framework	SMC

B. Cryptographic Integration

Each IoT node was configured to sign its data packets and encrypt them (using ECDSA and AES-GCM, respectively) before transmission. Packet aggregation occurs at edge nodes, which then transmit the packets to the backend, where logging is provided by Hyperledger Fabric, which is immutable and auditable. For validation by multiple stakeholders, SMC of the aggregated values was done without disclosure of the raw measurements to the participating parties, using the MPyC framework.

C. Benchmarking Methodology

Our measurements focused on three main areas of interest:

- Cryptographic overhead — the signing, encryption and secure multiparty SMC per data packet, and the resulting latency.
- Network latency — the round-trip time from the IoT nodes to the cloud verification services.
- Blockchain throughput — the number of transactions per second in the simulated environmental logging workload. Benchmarking results are shown in Table III.

TABLE III. Performance Benchmarking of Prototype Under Controlled Testing Conditions

Metric	Result	Test Conditions
Avg. ECDSA Signing Time	7.4 ms	256-bit key, Raspberry Pi 4 IoT node, averaged over multiple runs
AES-GCM Encryption Time	2.1 ms	1 KB data packet, AES-256, IoT device layer
SMC Latency	11 ms	Aggregation phase at edge node
Blockchain Throughput	145 TPS	Hyperledger Fabric testbed under simulated environmental logging workload

The results from Table III show that when deployed on low-power IoT nodes, ECDSA and AES-GCM operations cause negligible computational delays. Hyperledger Fabric offers a

real-time (close to) append-only ledger system that facilitates rapid transaction logging while being subjected to simulated workload scenarios. Even though SMC entails an extra layer of processing, its delay is still tolerably low for operational use when it comes to privacy-preserving streaming analytics.

VI. Evaluation

The assessment of the proposed framework has been done in three parts: validation of security, evaluation of the performance, and evaluation of the auditor/operator components. Results of the evaluation and analyses are shown in an organized way and in conjunction with some tables.

A. Security Validation

The evaluation of the model was done with respect to some surrogate attack vectors that came from the specified adversary model, which includes the evaluation of model manipulation, replay attacks, unauthorized access, insider attacks, and denial of service. A set of controlled simulations was done to measure the effectiveness of the defense mechanisms that were employed. The summary of all of the security evaluations and assessments is in Table IV.

TABLE IV. Security Evaluation Results Under Simulated Attack Conditions

Attack Scenario	Defense Mechanism	Outcome
Data Tampering	ECDSA digital signatures	100% detection (n = 50 simulated tampering attempts)
Replay Attack	Timestamp + hash-chained log	Successful prevention under replay simulation
Unauthorized Access	AES-GCM encryption	No unauthorized disclosure observed
Insider Manipulation	Immutable ledger logs	Attempted modifications flagged and traceable
Denial of Service (DoS) on IoT Nodes	Redundancy + edge buffering	Delayed reporting without data loss

The attack scenarios included in Table IV show the attack scenarios included against the proposed mechanisms are ECDSA documented altered packets with AES-GCM hindering access to protected data, the replay attack was documented by a hash-chained log, and the creation of unmodifiable logs placed the insider modifications, and edge buffering mechanisms decreased the impact of denial-of-service (DOS) attacks and operationally without loss of data. All of these results show us that the attack scenarios included against the proposed mechanisms are of a layered and comprehensive nature and thus provide complete lifecycle protection against the layered attack scenarios.

B. Performance Analysis

The performance of the system was assessed against the simulated deployment of the system in order to evaluate the operational feasibility. The results show that the above-mentioned proofs of the proposed mechanisms provide a dominant system that provides a clear and easy environment for the data packets that are typical for the environment, blockchain throughput provides the ability to log almost in real-time, and SMC provides a system that is operationally acceptable within the provided range of dominant systems. The performance metrics for the operational scenario provide a summary of the performance metrics measured.

TABLE V. Summary of Operational Performance Metrics

Parameter	Measured Value	Observation
Avg. End-to-End Latency	62 ms	Suitable for near real-time monitoring
Packet Loss under Load	<1%	Stable under simulated network stress
Blockchain Synchronization Delay	<5 s	Observed under periodic logging intervals without throughput degradation

Table V shows that the system has an end-to-end latency of 62 ms, and under simulated conditions loss of packets is negligible. The delay caused by blockchain sync is within the operational limits acceptable for sync monitoring to near real-time. These findings reiterate the fact that the integration of ledger technology does not lead to a considerable performance reduction. The total energy impact also remained within the operational limits practicable for solar-assisted and hybrid-powered deployments to the coasts. Regarding the quantitative evaluation of the deployed cryptographic primitives, latency measurement was done to assess the impact of the computation of signature generation, encryption, and SMC at the levels of IoT and edge.

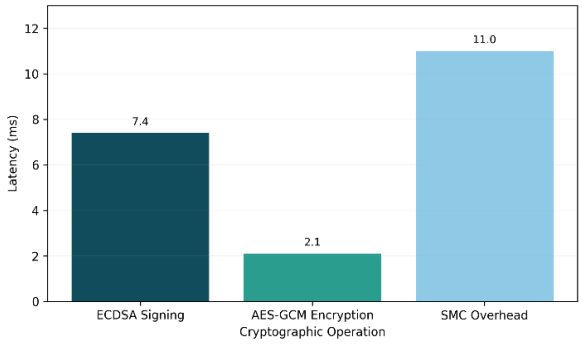


Fig. 1. Measured latency of deployed cryptographic primitives, demonstrating lightweight computational overhead at IoT and edge levels.

As illustrated in Fig. 1, the measured latency of the deployed cryptographic primitives demonstrates lightweight computational overhead at both the IoT and edge layers. AES-GCM encryption shows the smallest delay of 2.1 ms and ECDSA signing takes 7.4 ms per transaction, which is still within an acceptable bound performance for IoT devices that have constrained resources. The increased latency of SMC is 11 ms and is mainly limited to the aggregation phases and not so much to the continuous packet transmission, limiting its impact on the operation overall. This shows us that strong cryptographic guarantees can be integrated without the computational overhead becoming prohibitive, enabling real-time blue carbon monitoring deployments to remain practical.

To examine whether the added blockchain had an impact on performance, the throughput of the Hyperledger Fabric backend was recorded and compared, as shown in Fig. 2, to the minimum operational requirements for uninterrupted logging of the environmental data.

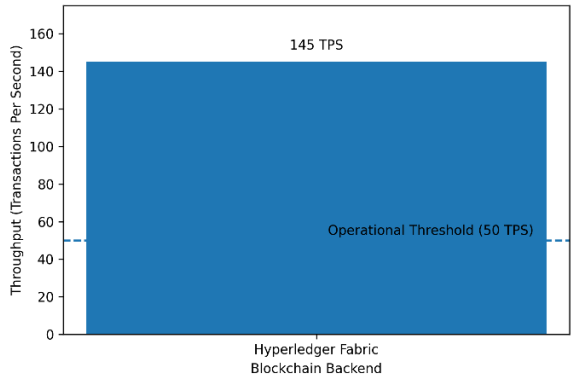


Fig. 2. The performance of Blockchain throughput concerning the minimal operational transaction necessities for the scalable monitoring of the environment.

The observed throughput of 145 transactions per second (TPS) provides a performance headroom of 2.9× relative to the minimum operational threshold for continuous environmental logging, which is 50 TPS. Under the given deployment conditions, the system, combined with hierarchical edge aggregation and batching of sensor measurements, enables proportional scaling of distributed IoT nodes without reaching the capacity of the ledger. The design also separates the cryptographic computation at the device level from the consensus at the ledger, thus allowing for both vertical and horizontal expansion of the monitoring clusters. All of this suggests the framework is capable of scaling to large coastal monitoring deployments in a controlled manner without reducing the performance expectations. A simple snapshot in scaling can be achieved by estimating ledger throughput with the equation $TPS_{req} = (n \times r) / b$, where n is the number of sensors, r is the average number of packets per second produced by each sensor, and b is the edge layer batching. With an observed throughput of $TPS_{obs} = 145$ and an estimated operational average of 50 TPS, the framework is designed to maintain proportional scalability in relation to the number of distributed IoT nodes, especially with hierarchical edge aggregation and periodic ledger commitment.

C. Feedback from Operators and Auditors

Starting from the framework, a pilot usability and validation assessment was conducted to understand the propositional framework from operational and auditing perspectives. Feedback was also gathered through structured workflow simulations as they relate to the assessment and operational transparency of logging, as well as the readiness of certification. Feedback was summarized and presented in Table VI.

TABLE VI. SUMMARY OF AUDITOR AND OPERATOR FEEDBACK (PILOT STUDY)

Stakeholder	Key Observation	Practical Impact
Field Operator	Automated signing and encryption reduced manual intervention	Increased operational efficiency
Auditor	Immutable logs provided verifiable evidence trails	Strengthened certification confidence
Policy Analyst	Alignment with SGI/MGI and international carbon standards	Facilitated regulatory integration

The five simulated audits and four system operators were part of the pilot evaluation structured session to validate workflow systems. System operators were able to provide feedback on the testing environment, but the test environment was simulated.

VII. CONCLUSION AND FUTURE WORK

This paper proposes an architecture that incorporates compliance and cryptography to protect blue carbon accounting pipelines in vast and hybrid Internet of Things and Cloud-based construction environments. Protection encompasses the monitoring lifecycle of Authenticity, Integrity, Confidentiality, and Traceability by compliance with ECDSA Digital Signature Standards, the Advanced Encryption Standard with Galois/Counter Mode (AES-GCM), hash-chained logging, secure multiparty computation, and compatibility with national Public Key Infrastructures (PKI). The processing that results from the cryptography has shown to be negligible. The combined signing and encryption latency at the IoT layer remained below 10 ms, and the blockchain backend sustained a throughput of 145 transactions per second. The results of closed tests for tampering and replay scenarios demonstrate that the implementation of a closed test for retention and replay scenarios is a viable option. The findings of these tests indicate that enforcing the monitoring lifecycle can be achieved without compromising operational performance. The assessments were performed in a controlled environment of the prototype system. There is the possibility of complicated post-maintenance operation in distributed key and certificate management, which may increase the complexity of ongoing operation and maintenance. The SMC component takes an honest-but-curious approach, which may not appropriately capture adversarial collaborative scenarios in the real world. Future research will focus on post-quantum migration strategies combined with the real-time detection of anomalies to capture sensor behavioral anomalies. A step-wise migration strategy is proposed whereby Post-Quantum Cryptography (PQC) signatures will be at the edge/cloud attestation layers, and elliptic curve cryptography (ECC) will remain on constrained devices until a sufficient hardware upgrade is available. Hybrid certificate schemes may allow backward compatibility with existing PKI validation systems.

REFERENCES

[1] S. Farahmand, N. Hilmi, and C. M. Duarte, "The rise and flows of blue carbon credits advance global climate and biodiversity goals," *npj Ocean Sustainability*, vol. 4, no. 1, art. no. 39, 2025.

[2] R. Alajmi, "Green growth is a pathway to sustainable development: An empirical study of Saudi Green Initiative," *J. Econ. Admin. Sci.*, vol. 31, no. 148, pp. 115–129, 2025.

[3] A. F. M. Maniruzzaman and K. Al-Saleem, "Renewable energy and energy justice in the Middle East: International human rights, environmental and climate change law and policy perspectives," *J. World Energy Law Bus.*, vol. 18, no. 1, art. no. jwae021, 2025.

[4] O. G. Ndubuisi and F. I. S. P. O. N., *Improving Advanced Research Methods for Climate Change and Environmental Sustainability*. 2025.

[5] P. K. Goel and S. P. Yadav, "Cloud security protocols for protecting carbon monitoring systems," in *Advanced Systems for Monitoring Carbon Sequestration*. Hershey, PA: IGI Global, 2025, pp. 127–148.

[6] National Center for Digital Certification (NCDC), "National Center for Digital Certification (NCDC)," GOV.SA, Available: https://my.gov.sa/ar/content/13990, Accessed: Sept. 2, 2025.

[7] National Institute of Standards and Technology (NIST), "National Institute of Standards and Technology," Available: https://www.nist.gov/, Accessed: Sept. 1, 2025.

[8] L. Zhao, X. Fan, and S. Xiao, "Remote-sensing indicators and methods for coastal-ecosystem health assessment: A review of progress, challenges, and future directions," *Water*, vol. 17, no. 13, art. no. 1971, 2025.

[9] C. Zarzuelo Romero, A. López-Ruiz, M. Bermúdez, M. Ortega-Sánchez, and I. Caballero, "Monitoring intertidal ecosystems: Assessing spatio–temporal variability with Sentinel-2 and Landsat 8," *Int. J. Appl. Earth Obs. Geoinf.*, vol. 128, art. no. 104676, 2025.

[10] P. J. Kashyap, K. Bora, S. Das Podder, A. Amakawa, A. Paul, and C. R. Saikia, "Mangrove monitoring using geospatial technologies incorporating field inventory: A review," in *Revealing Ecosystem Services Through Geospatial Technologies: Beyond the Surface*. Cham: Springer, 2025, pp. 123–138.

[11] M. Al-Khalidi, R. Al-Zaidi, T. Ali, S. Khan, and A. K. Bashir, "AI-optimized elliptic curve with certificate-less digital signature for zero trust maritime security," *Ad Hoc Netw.*, vol. 166, art. no. 103669, 2025.

[12] H. Wahsheh and M. Wahsha, "Digital safeguards for coral reefs with cryptographic watermarking in marine research," in *Proc. Int. Conf. Computing Systems & Intelligent Applications (ComSIA)*. Cham: Springer, 2025, pp. 685–696.

[13] S. Noliya, A. Gupta, and V. Kumar, "E-waste management through blockchain technology: Supply chain led environmental sustainability," *EDPACS*, pp. 1–11, 2025.

[14] M. Wahsha, H. Wahsheh, and T. Al-Najjar, "Enhancing marine litter management in the Gulf of Aqaba through AI," in *Proc. Int. Conf. Information, Communication and Computing Technology (ICICCT)*. Cham: Springer, 2024, pp. 56–67.

[15] M. Wahsha and H. Wahsheh, "Deploying AI for health monitoring of Diadema sea urchins: Toward sustainable marine ecosystems," in *Proc. Int. Conf. Inventive Communication and Computational Technologies (ICICCT)*. Singapore: Springer, 2024, pp. 651–660.

[16] A. Mandal and A. R. Ghosh, "AI-driven surveillance of the health and disease status of ocean organisms: A review," *Aquaculture Int.*, vol. 32, no. 1, pp. 887–898, 2024.

[17] M. Wahsha and H. Wahsheh, "From data to action toward sustainable marine conservation: AI-based coral health assessment," in *Proc. Int. Conf. Computing and Communication Networks (ICCCN)*. Singapore: Springer, 2024, pp. 213–222.

[18] T. Miller, I. Durlik, E. Kostecka, S. Sokołowska, P. Kozlovska, and R. Zwolak, "Artificial intelligence in maritime cybersecurity: A systematic review of AI-driven threat detection and risk mitigation strategies," *Electronics*, vol. 14, no. 9, art. no. 1844, 2025.

[19] G. S. Pujar, A. Taori, A. Chakraborty, and T. Mitran, "Sensing climate change through earth observations: Perspectives at global and national level," in *Digital Agriculture: A Solution for Sustainable Food and Nutritional Security*. Cham: Springer, 2024, pp. 225–280.

[20] T. Zhukabayeva, L. Zholshiyeva, N. Karabayev, S. Khan, and N. Alnazzawi, "Cybersecurity solutions for industrial internet of things–edge computing integration: Challenges, threats, and future directions," *Sensors*, vol. 25, no. 1, art. no. 213, 2025.

[21] G. Premi, A. Mishra, A. Kaur, J. R. Sahoo, P. Aggarwal, and S. Mathur, "Secure multi-party computation for privacy preservation in collaborative networks," in *Proc. 2025 Int. Conf. Automation and Computation (AUTOCOM)*, 2025, pp. 637–642.

Privacy-Aware Customer Segmentation Using a Distributed Graph-Based Attribute Projection Framework

Pentareddy Ashalatha[1], Dr. G. Krishna Mohan[2]
Research Scholar, Department of Computer Science & Applications,
Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India [1]
Professor, Department of Computer Science & Engineering,
Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, India [2]

***Abstract*—Customer segmentation plays a vital role in Business Intelligence (BI) by enabling organizations to understand customer behavior, enhance personalization, and support informed decision-making. Conventional segmentation approaches, including K-Means clustering, hierarchical methods, and hybrid deep learning models, often face limitations when handling high-dimensional customer data and typically lack built-in mechanisms to address privacy concerns. As customer analytics increasingly relies on sensitive personal information, these limitations pose significant challenges for responsible data-driven applications. To overcome these issues, this study introduces a Distributed Graph-Based Attribute Projection Framework (GAPF) for privacy-aware customer segmentation. The key novelty of the proposed framework lies in its ability to minimize sensitive attribute exposure while preserving meaningful relational patterns among customers through graph-based representations. GAPF employs a distributed processing pipeline that integrates attribute projection to reduce identifiability, heuristic-driven customer similarity graph construction, graph convolutional network (GCN)–based feature learning, and community detection for final segmentation. The framework is implemented using Python, NetworkX, and PyTorch Geometric and evaluated on the Mall Customers dataset and large-scale anonymized synthetic data to assess scalability. Experimental results demonstrate that GAPF achieves superior segmentation performance, with an accuracy of 98%, precision of 92.5%, recall of 94.0%, and an F1-score of 93.2%, while also exhibiting efficient execution and reduced privacy risk. These findings confirm GAPF as a robust and practical solution for privacy-aware BI applications.**

Keywords*—Business intelligence; privacy-aware customer segmentation; graph-based learning; federated analytics; graph neural networks

I. Introduction

Over the past years, customer segmentation has been integrated into the BI systems as one of the foundational aspects that facilitate organizations to study customer behavior, discover market trends, and provide customized products and services [1]. As online platforms are becoming increasingly popular, businesses can access large volumes of customer data in a variety of places, such as transaction records, online interactions, mobile apps, and loyalty programs [2], [3]. Successful segmentation of such data helps in strategic decision-making in the areas of retail, finance, telecommunication and e-commerce [4], [5]. Nevertheless, the growing magnitude, size and delicacy of customer information pose a serious obstacle to conventional analytical practices [6]. Traditional segmentation methods, such as K-Means clustering, hierarchical clustering, and density-based clustering, mostly focus on distance-based similarity calculations in high-dimensional feature spaces. Although these methods are computationally efficient, they may lack scalability, robustness to noise, and the ability to identify non-linear relationships between customers. [7]. Although these techniques are computationally effective, they are frequently characterized by low scalability, noise sensitivity and low power to describe non-linear and intricate associations among customers [8]. Recent developments in hybrid deep learning-based segmentation methods have increased the representational power, but they generally need centralized access to data and large-scale feature exposure [9]. Consequently, these methods have grave privacy concerns, particularly when handling personally identifiable information (PII) and sensitive behavioral characteristics [10], [11]. Due to the rising regulatory limitations like the General Data Protection Regulation (GDPR) and the rising awareness of data abuse, privacy preservation has become a paramount need in contemporary BI systems [12], [13]. Current privacy-saving methods, such as data anonymization, k-anonymity, and differential privacy, tend to impair the quality of analysis, or can interfere with relational patterns that can be used to perform accurate segmentation [14] [15]. Besides, the vast majority of the current segmentation models consider privacy as an after-processing, but not an in-processing, component of the modeling pipeline, resulting in suboptimal trade-offs between data utility and privacy protection [16].

Graph-based learning has been proposed as an effective paradigm to overcome the drawbacks of these methods in order to model complex relations in structured data. The similarities and interactions among customers are naturally represented in graph forms, giving more expressive and contextual segmentation. Nevertheless, the existing graph-based customer analytics systems do not pay much attention to privacy-conscious attribute processing and do not scale to distributed BI systems. This is the gap that demonstrates the necessity of an integrated framework that would incorporate both graph learning and privacy-conscious data transformations in a

scalable way. In this regard, a Privacy-Aware Customer Segmentation framework is proposed in this study based on a Distributed GAPF. The proposed solution proposes attribute projection schemes to minimize attribute identifiability and maintain critical relational systems among customers. The framework facilitates the process of robust segmentation by using community detection in the context of a similarity graph constructed on the projected attributes and representation learning with the help of GCNs. Scalability of large volumes of data is an additional benefit with the distributed design of GAPF that does not affect privacy limits. The overall contributions of this work include three folds: (1) the creation of a new graph-based attribute projection strategy, where the privacy awareness is incorporated directly into the segmentation pipeline, (2) the development of a scalable and distributed graph learning framework, used to carry out customer segmentation, and (3) a comprehensive experimental analysis, showing that the given approach outperforms the traditional techniques of clustering and hybrid learning methods in terms of performance, scalability, and low privacy risk. On the whole, the framework suggested is a sensible and accountable customer segmentation solution to the contemporary BI framework, which balances the efficiency of the analytics with the privacy of information.

A. Problem Statement

Although customer segmentation has been used extensively in Business Intelligence systems, the current methods of segmentation, like traditional clustering and hybrid deep learning models, have serious constraints on large-scale, high-dimensional, and sensitive customer information [17]. They are usually based on centralized processing and access to raw attributes, which exposes them to privacy risks and non-adherence to regulations [18]. Moreover, privacy protection methods that are more traditional are also likely to decrease the accuracy of analytical methods due to data distortion or the incomplete preservation of meaningful relational patterns between customers [19]. Consequently, scalable segmentation schemes can lack scalable segmentation frameworks that can successfully trade segmentation efficiency and privacy consciousness [20]. This brings a pressing requirement for an integrated solution that can combine privacy-preserving attribute manipulation with relational learning in a distributed and scalable form, and at the same time be of high segmentation quality as in the case of real-world Business Intelligence applications.

B. Research Motivation

This study is motivated by the fact that organizations are becoming more dependent on customer data to make intelligent business decisions and the rising concerns about data privacy, data security and adherence to regulations. Even though sophisticated analytics and learning models have improved the accuracy of segmentation, they may need complete access to sensitive customer properties, which expose organizations to privacy risks and restrict their application in practice. Simultaneously, current privacy-preserving methods often reduce the analytical value or are not able to reflect on intricate customer associations. The necessity of such challenges is the need to have a segmentation framework, which can maintain the privacy of the customers, at the same time, model relation patterns efficiently, and scale to large datasets. Through the combination of privacy-conscious attribute projection and graph learning in a distributed environment, the study will address the gap between the responsible data usage and high-performance customer analytics in contemporary Business Intelligence systems.

C. Significance of the Study

The study is important because it fills an important gap in modern Business Intelligence by incorporating the conservation of privacy directly in the customer segmentation process without affecting the performance of the analysis. The proposed graph-based attribute projection framework incorporates privacy awareness in data representation and learning, unlike the traditional approaches to segmentation, which regard privacy as an extrinsic limitation, and allows taking responsible care about sensitive customer data. The research effort provides a scalable and distributed segmentation model that is able to process high-dimensional and large-scale data and retain relational structure that is vital in performing an accurate customer profiling process. In practical terms, the framework contributes to adherence to data protection regulations and improves trust in the data-driven decision systems, which makes it very relevant to practical application in various areas, including retail, finance, and telecommunications.

D. Key Contribution

Introduced a new privacy-aware attribute projection algorithm that minimized sensitive attribute identifiability and still maintained valuable customer relations.

Developed a distributed graph-based customer segmentation framework: Built a scalable framework with high-dimensional data and efficient scale to large-scale data.

Community detection based on an integrated Graph Convolutional Network (GCN) representation learning to enhance segmentation accuracy and robustness.

Demonstrated to outperform conventional clustering and hybrid learning methods in a large scale of experimental analysis regarding accuracy, scale, and reduced privacy risk.

E. Rest of the Sections of the Study

The following is the rest of the section: related works are given in Section II. The suggested methodology framework is presented in detail in Section III. The results of the study are shown in Section IV. Lastly, Section V wraps up the conclusions and future of the suggested framework.

II. Related Works

Khan et al. [21] focus on the use of the BAT-ANN model for the behavior of customer churn in the telecom sector. Based on the Big ML dataset, the model establishes 89.2% accuracy, which reveals the ability of the model to pinpoint significant factors that lead to customer churn. The BAT-ANN model is useful for companies which seek to develop specific retention initiatives and thus improve the overall customer goodwill and revenue. The study contributes a framework for churn prediction by utilizing the Bat Algorithm for feature optimization and an ANN for classification. However, although the proposed model displays good performance, it is not very effective in more real-world environments, for instance, working with streaming data.

This limitation is because of the computational dimension of ANN and the difficulty of processing continuously changing patterns of data. Furthermore, the permutation of the model across different industries is unknown owing to the great difference in customer behavior and churn factors across domains. While this study has a good start in providing an understanding of churn prediction models to help in their improvement, the work could endeavor to build on the scalability and robustness of the models in order to address the dynamic and complex nature of various real-world settings.

Lu [22] investigates the problem of attaining utility and privacy simultaneously in social media data mining by using a pairwise GCN. The model aims at providing considerable levels of protection against adversarial threats, Still delivers stupendous degree of accuracy so as to effectively mine data with high security standards. In turn, to enhance privacy protection during the training of the models, federated learning is incorporated into the learning process. They do this by maintaining data decentralization, so the data stays local to the owner's environment, thus eliminating privacy concerns while at the same time facilitating collaborative model training. The experiments carried out in this study reveal that the proposed approach is realistic by indicating how privacy and utility can reasonably be traded off. However, the study finds the following challenges: Therefore, one main workload is the scalability of the proposed federated learning framework, especially in big social media applications that demand more computational and communication power. In addition, it can be seen that the model is good and performs well on a test set, though it might be less effective in application on real-world data with various, noisy, and missing data. Such conditions, typical in social media data, can pose a limit on how well the model generalizes. There are, however, several challenges to this approach that require further study to confirm the obtained results and further expand the methodological applicability in similar conditions and types of SNS, such as mixing with other user attributes, and the dynamic nature of SM features.

Li et al. [23] provide a systematic review of privacy preservation schemes in MEC task offloading. The study categorizes privacy-preserving methods into three main areas: data storage, transmission security, and choosing the appropriate place for data storage. Each category is accompanied by various techniques that are explained in detail in their mechanism of data privacy while offloading. The study focuses on the privacy requirements in MEC environments where different data may be, most of the time, highly confidential and have to be processed and transmitted through distributed networks. Nevertheless, the given study has several significant limitations. Comparatively, it does not feature quantitative assessments or accuracy indices for the aforesaid techniques, and hence there is no real measure of the effectiveness of them all. Altogether, the effectiveness of the proposed methods in similar conditions is shown, but their applicability for real-world situations is not explored in detail. More complex edge environments with different and changing privacy needs (for instance, industrial or healthcare scenarios) might be problematic in this regard. The study suggests that work should concentrate more on concrete applications and strategies for overcoming the conflict of interest between privacy, performance, and scalability. I believe that more such efforts are necessary in order to close the gap between the ideas explored by study and the actual implementation of privacy-preserving MEC solutions.

Anand and Lee [17] explore the feasibility of GANs for achieving customer data privacy with the retention of critical data features. The benchmarks of tradeoff management are improved by their proposed model while still keeping a balance between accuracy and privacy. Because the approach produces simulated data that presents similar features to real data but does not include any personally identifiable information, it is helpful when used in practical applications, including setting artificial price marks and marketing strategies to target shoppers. Thus, the given study illustrates the applicability of the proposed model for the analysis of big real-time customer data in industries. Nevertheless, it has also pointed of weakness. Still, there are some drawbacks connected with the computational heuristic nature, especially if the organization has problems with its computational power. Moreover, there is always a risk of overfitting that may affect the model's performance in different contexts and marketing datasets. This issue may cause a restriction on applying this method in different environments with different customer behavior. Despite the study's contributions to the utilization of the method for privacy conservation, more study needs to be conducted to enhance the computational complexity, and minimize overfitting and improve the generality of the model.

Sharma, Patel, and Gupta [18] propose a framework integrating K-Means clustering with Deep Learning to improve the customer segmentation process. To address the above issue, the authors combine the advantages of K-Means and deep learning while proposing a new approach that features a higher level of segmentation detail and better accuracy. When applied to the heteroscedastic labeling problem in large-scale retail datasets, the proposed model has shown substantial promise in modeling customer behavior and thus can be useful in determining efficient marketing strategies. Due to the employment of deep learning, the method is capable of analyzing intricate structures in the customer data so that various business campaigns can be more effectively developed. However, the hybrid model does have its drawbacks, which will be discussed in this article. Furthermore, the optimization of the model is also extremely sensitive to the quality of the clustering achieved by K-Means. While clustering, if done in an improper way, affects the result very badly and also reduces the efficiency of the whole process. Consequently, although the described hybrid model can be considered an improvement over the classical approaches to segmentation, additional study is required to address these issues. Possible future work on this matter could be oriented towards minimizing the computational complexity as well as enhancing the stability of the initial clustering phase, which makes the suggested model applicable for various businesses and datasets.

Ali et al. [19] investigated the feasibility of using federated learning for predictive maintenance in Industry. FL ensures that participants can train datasets independently without sharing them with different parties and, in the process, allows organizations to leverage some collective conclusions while protecting individual data. In the course of the study, several examples illustrating the effectiveness of FL for improving

maintenance performance while preserving data integrity are also presented. As the study shows, there are quite a few drawbacks that should be solved before the implementation of such a system on a mass scale. There are several challenges, such as data heterogeneity, generated by the variation in data format and data quality of devices. Yet another factor—a communication overhead—is a problem here, as multiple interactions between devices and servers might slow down the process. However, the necessity of reliable privacy-preserving methods is most important in large industrial applications. This holds especially because the study identifies some challenges in using FL for predictive maintenance and the need to overcome these challenges to maximize the benefit of FL in achieving the need. Further should be targeted towards FL systems that would be manageable and applied successfully to the complex and variable necessities in industries.

From the study made by Tabianan, Velu, and Ravi [20], it is clear that customer segmentation acts as a key factor in increasing the profitability of e-commerce systems. Customer segmentation is a process that aims at categorizing customers to enhance the efficiency of satisfying the needs of each category of customers. The proposed method seeks to optimize both within-cluster homoscedasticity and between-cluster heteroscedasticity so that customers belonging to a given cluster exhibit similar behavior to the other customers in the same cluster but different from those in other clusters. Such a segmentation puts a business in a better position to sell its products to different customers through promotional tools, services, and marketing strategies, all of which increase customer satisfaction, loyalty, and profitability. However, the application of the current study has some drawbacks because of the reliance on the adopted K-Means algorithm. The problem with the first few choices affects the degree of cluster reliability and accuracy of a segment. Moreover, real life often embodies different behavior of customers and different patterns which shift with time, and this is another factor which opposes K-Means, which is not capable of handling these issues. It also works poorly with datasets containing outliers or datasets where the density of clusters varies a lot, which may be true in some more complex e-commerce environments. These issues bring about the necessity of developing new and more flexible clustering techniques to provide e-commerce systems with more stability in the face of the constantly changing customer behavior.

Ebrahimi et al. [24] conducted an experiment to investigate the relationship between Social Network Marketing (SNM) and Consumer Purchase Behavior (CPB) within the context of Facebook Marketplace. To reveal specific consumer segments, the empirical study used two types of Structural Equation Modeling (SEM) and two machine learning methods that are not supervised: hierarchical clustering and K-Means. The online sample of Facebook Marketplace users in Hungary was taken and provided data, yielding 475 participants and a high response rate of 98.1. Following the screening, there were 466 valid responses that were left to be analyzed. The use of clustering of the consumers identified nine consumer groups in terms of both demographic and behavioral attributes. This division allowed agencies to create specific marketing plans for each group, and this enhanced customer engagement and sales. Nevertheless, the external validity of the results can be considered rather low because the research was conducted in one social networking site and a particular geographic area. The biggest methodological weakness is the use of static clustering algorithms that make the assumption of the stability of consumer behavior. Since the interactions among the users on social networks are dynamic and undergo constant changes, the fixed segmentation models might not reflect the real-time changes in the preferences of the consumers. Based on this, the authors emphasize that more sophisticated, dynamic clustering methods are required that will be able to capture the shifts in behavioral patterns, and more precise and responsive marketing strategies in the fast-moving digital market environment are guaranteed.

TABLE I. SUMMARY OF RELATED WORKS

Author(s)	Techniques	Advantages	Disadvantages
Khan et al. [21]	BAT-ANN (Bat Algorithm + ANN)	Achieved 89.2% accuracy on churn prediction and identified key factors for customer churn.	Not effective for streaming/real-time data, and poor generalizability across industries.
Lu [22]	Pairwise GCN + Federated Learning	Preserves privacy with decentralized training and high accuracy under security constraints.	Scalability issues in large social media applications are sensitive to noisy/missing data
Li et al. [23]	Privacy-preserving schemes in MEC	Comprehensive review of MEC privacy techniques, and identifies key mechanisms for storage, transmission.	Lacks quantitative effectiveness measures, and no solutions for dynamic environments.
Anand & Lee [17]	GANs for privacy-preserving data	Balances accuracy and privacy, and generates synthetic but realistic data	High computational cost and risk of overfitting.
Sharma et al. [18]	K-Means + Deep Learning (Hybrid)	Captures intricate customer patterns, and improves segmentation detail & accuracy	High computational cost, and requires robust infrastructure.
Ali et al. [19]	Federated Learning for maintenance	Preserves data privacy by local training, and effective in industrial settings.	Data heterogeneity challenges, and communication overhead
Tabianan et al. [20],	K-Means for behavioral segmentation	Groups customers based on behavior, product & events Improves satisfaction, loyalty, and profitability.	Sensitive to initial cluster centers, and poor performance on dynamic/outlier-rich datasets- Cannot adapt to evolving behaviors.
Ebrahimi et al. [24]	SEM + Hierarchical & K-Means	Identifies key SNM factors influencing purchase behavior, and defines customer clusters effectively.	Sample bias due to convenience sampling, and Limited generalizability to other platforms.

Table I presents a compact evaluation of the available methods in customer grouping and privacy protection. It points out the methods used, their advantages in regard to increasing

accuracy of segmentation, confidentiality, scalability, or significant disadvantages like computational complexity, scalability, or lack of applicability. This comparative analysis emphasizes on the issues of robust, privacy-sensitive and sizeable frameworks.

III. Privacy-Aware Customer Segmentation Using Graph-Based Attribute Projection

The proposed methodology is rooted in a privacy-aware and decentralized graph-based segmentation pipeline that is aimed at creating a balance between analytical and data protection. The framework combines preprocessing, attribute projection that is privacy sensitive, distributed graph construction, and graph neural representation learning in a single architecture that can be used in scalable Business Intelligence applications. A similarity measure based on a heuristic is employed to build a customer graph, which reflects the pattern of relationships among people based on the projected attributes. GCNs are then used to learn rich representations of customers using the combined attributes information and structural information. Lastly, on the learned graph embedding, community detectors are applied to create meaningful and privacy-preserving customer segments that can be used in Business Intelligence applications at scale. The distributed execution of the suggested framework was implemented with the help of PyTorch Distributed Data Parallel (DDP), which allows parallel representation learning in the form of a graph that is distributed among several computational units. This decentralized structure enhances training effectiveness and facilitates scalability by operating with more customers in Business Intelligence applications. Fig. 1 shows the proposed methodology workflow.

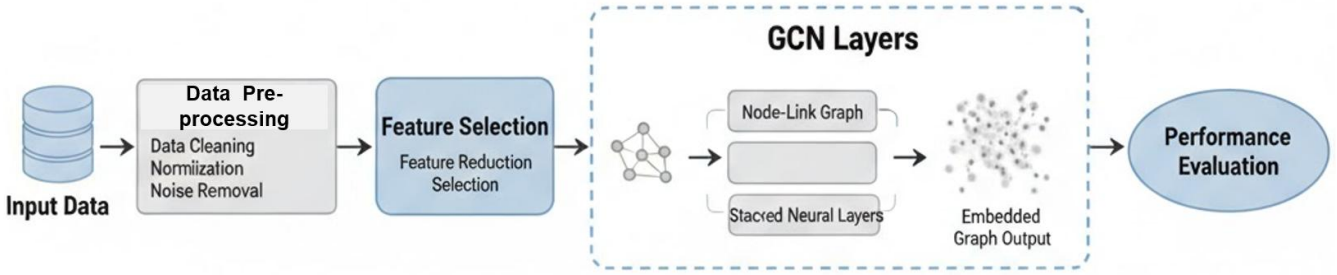


Fig. 1. Proposed model workflow.

A. Dataset Description

A famous dataset is the Mall Customers data set availed by Kaggle because it is easy and applicable to the customer segmentation study (Table II) [25]. It includes 200 single customer records, the properties will be CustomerID, Gender, Age, Annual Income (k), and Spending Score (1-100), all of which will be required to project the trend of demographics and behavior. In this study, GAPF has been used on the data. Attributes are seen as nodes, and relationships are drawn among the similar customers in terms of attributes, be it demographic or spending habits. The graphical display facilitates customer privacy-friendly clustering and retention of the relational structures to support scalable customer segmentation and BI applications. Although the Mall Customers dataset is very small, it can be utilized in reference in testing segmentation frameworks. Scalability was also achieved by having the data expand at runtime under control.

TABLE II. Sample of the Mall Customers Dataset

CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1–100)
0001	Male	19	15	39
0002	Male	21	15	81
0003	Female	20	16	6
0004	Female	23	16	77
0005	Female	31	17	40

Despite the fact that Mall Customers' data is relatively small and it is used mostly as a demonstration of the concept, the data offers an appropriate measure of the segmentation frameworks in the validity testing. Scalability was experimented with by expanding the data under control in the real time. Nonetheless, confirmation in larger real-world data sets is required to substantially ensure the effectiveness of the framework in a large-scale Business Intelligence setting.

B. Data Preprocessing

The first step in data mining is an essential step of data preprocessing that guarantees the quality, uniformity, and appropriateness of the customer data in terms of graph-based learning and privacy-aware segmentation. Raw customer data are often filled with missing values, noise, non-homogeneous scales and high-dimensional attributes that negatively impact similarity computation and subsequent graph modeling. Thus, a system preprocessing pipeline is employed before attribute projection and graph building.

1) Handling missing values: The absence of attribute values will be filled in through statistical imputation procedures so that the completeness of the data set is maintained. In the case of numerical attributes, mean or median imputation is taken, which is defined as Eq. (1):

$$x_{i,j} = \begin{cases} x_{i,j}, & if\ x_{i,j} \neq \emptyset \\ \frac{1}{N}\sum_{k=1}^{N} x_{kj}, & Otherwise \end{cases} \quad (1)$$

where $x_{i,j}$ is the value of the j-th attribute of the i-th customer, and N is the number of customers. In cases of categorical variables, the missing entries are replaced with the most repeated type (mode).

2) Noise removal and outlier treatment: Interquartile range (IQR)-based filtering or normalizing z-scores are used to

eliminate noisy data and extreme values. An outlier of Eq. (2) is a data point of Eq. (2) that satisfies the following condition:

$$|z_i| = \left|\frac{x_i - \mu}{\sigma}\right| > \tau \quad (2)$$

where μ and σ are the mean and standard deviation of the attribute, and τ is a predefined threshold. These outliers are identified and either capped or they are removed to avoid distortion of similarity measures.

3) Normalization: Numerical features are normalized with minmax scaling in Eq. (3) to make sure that the attributes contribute in the same way when computing similarities. The numerical features are scaled to have min and max values equal to 1.

$$x_{i,j}{}' = \frac{x_{i,j} - \min(x_j)}{\max(x_j) - \min(x_j)} \quad (3)$$

The transformation puts all features within a similar range [0, 1], which does not allow characteristics having bigger numeric values to dominate.

4) Dimensionality handling: Redundant or low-variance feature elimination is used to deal with high-dimensionality before projecting the attributes. Variance thresholding is used in that a feature with a variance less than a given threshold $\in$ is dropped in Eq. (4):

$$\text{Var}(X_j) < \in \quad (4)$$

This is done to minimize computational cost and make graph construction and learning more stable. These preprocessing steps convert the customer data into a clean, normalized and compact data, which offers a solid base of privacy-sensitive attribute projection and graph-based segmentation.

C. Privacy-Aware Attribute Projection

One of the most important steps of the framework offered is privacy-aware attribute projection, which aims to minimize the chances of leakage of sensitive data and still provide enough analysis power to segment customers. The datasets of customers may include personally identifiable or sensitive information like income level, spending behavior, or demographic indicators, which, when directly applied in learning models, can be used to reconstruct individual identities. To address this threat, the suggested method converts the initial feature space into a more identifiable representation and then doe's similarity calculation and constructs a graph. Assume that the initial customer data matrix will be denoted by Eq. (5):

$$X \in \mathbb{R}^{N \times d} \quad (5)$$

where N denotes the number of customers and d represents the number of attributes. An expression f(·) is a privacy-conscious projection that is applied to project X into a lower feature space, Eq. (6):

$$Z = f(X) \in \mathbb{R}^{N \times k}, \quad \kappa \ll d \quad (6)$$

The purpose of this projection is to inhibit identifiability of direct attributes with the aggregation, transformation or linear combination of sensitive attributes and maintain the relative pattern of sensitive attributes with the customers. Practically, random projection, linear transformation or controlled feature aggregation can be used to accomplish the attribute projection. In Eq. (7), a random projection matrix R$\in \mathbb{R}^{d \times k}$ whose elements take values in a zero-mean distribution may be utilized:

$$Z = XR \quad (7)$$

The Johnson-Lindstrass lemma shows that such a transformation roughly preserves the pairwise distances between data points, and according to the lemma, this is necessary to achieve similarity-based segmentation, but greatly decreases the probability that it is achievable to reconstruct original attributes. As an additional measure to increase privacy, one may give the sensitive attributes smaller weights when projecting onto a lower-dimensional space or lumping them together into composite features, so that there is no dominant attribute in the representation.

D. Graph Construction

Following the projection of privacy-concerned attributes, the customers are projected into a low-identifiability and reduced feature space that does not lose relative similarities. According to these representations, a customer similarity graph is built to explicitly model customer-customer relationships. Let $Z \in \mathbb{R}^{N \times k}$ represents the attribute matrix after projection, each row z_i of the projection belongs to the k dimensions of the representation of the i-th customer. Every customer is represented as a node of an undirected graph $G = (V, E)$, where V is a set of customers, and E is a similarity-based relationship between the customers.

Similarity measures are heuristic measures made by calculating the projection of the similarity between the attributes to provide the edges. Usually employed metrics are cosine similarity or Euclidean distance. In the case of cosine similarity, the similarity between two customers i and j is determined as Eq. (8)

$$Sim(i,j) = \frac{z_i \cdot z_j}{\|z_i\| \|z_j\|} \quad (8)$$

The node i and node j are connected with an edge when $Sim(i,j)$ is greater than a certain threshold, δ or j is in the top-k nearest neighbors of i. This plan regulates the sparsity of graphs and is scalable, and a meaningful structure of relationships is retained. The adjacency matrix A that results represents the intensity of the relationship of similarity between customers.

The similarity threshold δ is also important in the quality of segmentation and density of graphs. The threshold can be lower, and this can introduce noisy or weak relationships among customers, which can decrease cluster separability. On the other hand, too big a threshold can cause a graph that is too sparse, so little can be connected. This has an impact on representation learning. The empirical tuning was done to find a level of balance that conserves the structural relationships and reduces the noise. The framework shows acceptable strength within a realistic margin of strength.

E. Graph-Based Feature Learning

After building the customer similarity graph, the GCNs are used to learn rich customer representations, where node

attributes and graph structure are combined to learn them. GCNs spread and combine the information in a collection of neighboring nodes, through which each customer representation is informed about not only its own projected attributes but also the traits of similar customers.

Considering the adjacency matrix A, self-loops are inserted to get $\tilde{A} = A + I$, and the corresponding degree matrix $\tilde{D}$ is computed. A GCN layer updates representations of nodes in the form of Eq. (9):

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \quad (9)$$

where $H^{(l)}$ denotes the node representations at the layer l, $W^{(l)}$ is a trainable weight matrix, and $\sigma(\cdot)$ is a non-linear activation function. The initial layer $H^{(0)}$ is set to the projected attribute matrix Z. The model uses several GCN layers to include the higher-order information about the neighborhood, which allows for the finding of latent communities of customers and behavioral patterns. Based on this graph-based learning of features, a compact, structure-aware set of embeddings is generated that greatly improves the quality of segmentation and still keeps the data representations privacy-preserving.

F. Customer Segmentation via Community Detection

Following graph-based feature learning, each customer can be represented by a low-dimensional embedding that captures the attribute-level information as well as relational structure obtained based on the customer similarity graph. These learned embeddings offer a solid basis for discerning coherent groups of customers. In order to derive the final customer segments, community methods are used on the graph, utilizing the obtained representation of meanings to combine homogeneous subgroups that represent the meaningful segments. Let $H \in \mathbb{R}^{N\times d}$ denotes the final node embedding matrix computed by the last GCN layer, with rows corresponding to the embedding of a customer. The goal of community detection is to subdivide the graph G(V,E) into C disjoint communities {C 1,C 2,C 3,… } where nodes in one community are more connected to other nodes in the same community compared to the connections that they have to a node in a different community. $\{C_1, C_2, \ldots, C_3\}$ such that nodes within the same community are more strongly connected than to nodes in different communities.

G. Scalability Analysis

The proposed GAPF structure is scalable with the help of its distributed graph learning structure. The framework is able to handle the growing volumes of data by partitioning the customer graph and training the GCN model on distributed processing. Simulation of the processes that are controlled using an expansion of data was applied to evaluate the performance of experimental evaluation using larger graphs, which showed better processing efficiency as more computational resources were increased. Even though the experiments have been conducted on a comparatively small benchmark set, the distributed design recommends the large-scale Business Intelligence applications. Real enterprise data validation will be taken into consideration in future research Scalability and effectiveness. The Louvain or Leiden algorithms are also usually used in this framework because of their modularity-based approach. Modularity Q is defined as Eq. (10).

$$Q = \frac{1}{2m}\sum_{i,j}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(c_i, c_j) \quad (10)$$

where A_{ij} is the adjacency matrix, k_i and k_j are the degrees of nodes i and j, m is the total number of edges, and $\delta(c_i, c_j)$ equals 1 if nodes i and j belong to the same community and 0 otherwise. The goal is to maximize Q, and this will result in internally cohesive and well-separated customer segments. Using a community detector with the GCN learned embedding, the segmentation procedure acquires the local similarities and global structural representations. It generates interpretable, stable, and privacy-conscious customer segments that are highly applicable to downstream Business Intelligence processes like targeted marketing, personalization, and strategic planning. Fig. 2 shows the GCN Architecture.

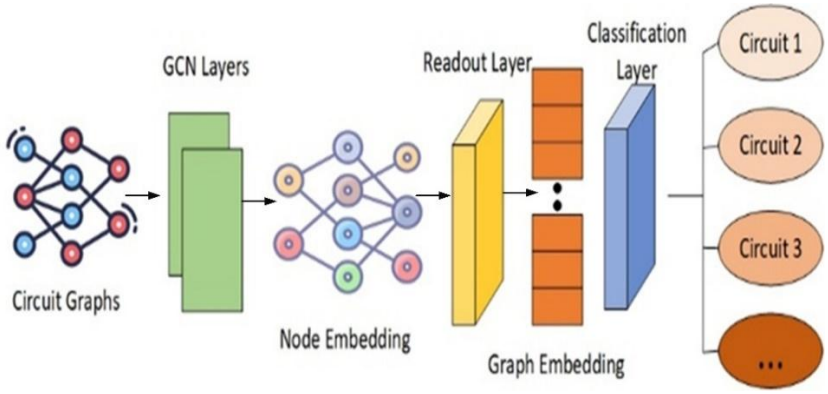


Fig. 2. GCN architecture.

In spite of the fact that customer segmentation is an unsupervised task in itself, accuracy, precision, recall, and F1-score were evaluated as supervised evaluation metrics through the usage of known customer category labels within the dataset to which the detected communities were mapped. This allows comparison quantitatively with baseline strategies that are often employed in Business Intelligence research. Moreover, measures that are clustering-oriented (modularity and intra-cluster similarity) were taken into consideration to augment the quality of segmentation.

In the suggested model, customer information is initially modelled as a similarity graph with each node indicating a

customer and edges portraying relational similarities. These projected customer attributes are then forwarded in a series of GCN layers to learn enriched node representations, which comprise information on attributes as well as the structure of neighborhoods. Such node representations are then grouped by a readout layer into smaller graph-level representations. Lastly, a classification or community detection layer is applied on the learned embeddings, and this attempts to recognize specific customer groups that can be used to perform accurate and privacy-aware segmentation applicable to large-scale Business Intelligence applications.

Algorithm 1: Privacy-Aware Customer Segmentation using Distributed GAPF

Input: Customer dataset $D \in \mathbb{R}^{n \times m}$ with n customers and attributes

Output:Customer segments C, Segmentation Performnace Metrics M,

Privacy Risk PR

1: Begin

2: Data Preprocessing

3: For each attribute $f_j \in D$

4: If missing values exist in f_j:

5: Impute using mean or median of f_j

6: End If

7: End For

8: Detect and handle outliers using IQR or Z-score filtering

9: Normalize numerical attributes using Min–Max scaling

10: Privacy-Aware Attribute Projection

11: Select sensitive attribute subset S⊆D

12: Generate projection matrix $R \in \mathbb{R}^{m \times k}$, where k≪m

13: Compute projected data Z=D×R

14: Replace original sensitive attributes with projected representations

15: Graph Construction

16: Initialize graph G=(V,E)

17: For each customer i∈D:

18: Create node $v_i \in$V

19: End For

20: For each pair of customers (i,j),i≠j:

21: Compute similarity $sim(i,j)$ using cosine or Euclidean distance on Z

22: If $sim(i,j) \geq \theta$ or j ∈ k-NN(i):

23: Add edge e(i,j) with weight $sim(i,j)$

24: End If

25: End For

26: Graph-Based Feature Learning

27: Initialize node features $H^{(0)}$=Z

28: For each GCN layer l=1 to L:

29: Update embeddings using

30: $H^{(l+1)} = \sigma\left(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right)$

31: End For

32: Customer Segmentation

33: Apply Louvain or Leiden community detection on learned embeddings $H^{(l)}$

34: Obtain customer clusters $\{C_1, C_2, \ldots, C_3\}$

35: Evaluation

36: Compute segmentation performance metrics: Accuracy, Precision, Recall, F1-score

37: Estimate Privacy Risk PR using sensitive attribute reconstruction error

38: Return C,Acc,PR

39: End

The overall steps of the proposed GAPF model of privacy-aware and scalable customer segmentation are outlined in Algorithm 1. The algorithm proposed carries out privacy-equipped customer segmentation using attribute projection and graph-based learning in a single pipeline. First, a privacy-aware attribute projection is applied to preprocess and convert customer data in order to decrease the identifiability of sensitive attributes. The projected features are then used to build a similarity graph, and the relational patterns of customers are captured. Each node's learning of structure-aware customer embeddings is carried out by using the graph constructed using the graph convolutional networks that aggregate the information of the neighborhood. Lastly, detecting communities in the learned embeddings is done to produce meaningful customer segments with a tradeoff between the accuracy of segmentation, scalability, and privacy.

IV. Result and Discussion

The conducted experimental analysis of the suggested Privacy-Aware Customer Segmentation framework by use of Distributed GAPF exhibits the presence of stable and promising results in light of the experimental conditions that were taken into account. Graph-based representation learning combined with privacy-sensitive attribute projection allows the model to learn more complex relational patterns among customers at a low risk of reconstructing sensitive attributes with minimal loss in the analysis value. The distributed implementation can be used to conduct effective training and convergence behavior, though the experiments were carried out on a relatively small benchmark dataset (Mall Customers, n = 200), which is mainly a proof-of-concept to test the validity of the framework. To test the scalability behavior, controlled data expansion experiments were conducted with the trend of almost linear execution with the increasing data size. Although the results suggest that the framework can be applicable to larger data sets, this should be confirmed with the help of real-life large-scale customer settings. On the whole, the results indicate that GAPF is a trade-off between the quality of segmentation, computing efficiency and privacy of Business Intelligence applications. The summary of the simulation parameters is presented in Table III.

TABLE III. Simulation Parameter

Item	Setting
Dataset	Mall Customers (n = 200)
Features Used	Age, Gender, Annual Income, Spending Score
Cleaning & Scaling	Mean/median impute; Min–Max [0,1]
Encoding	Gender → One-Hot
Similarity & Edges	Cosine similarity; threshold = 0.70

Graph Model	Nodes = customers; weighted edges by similarity
GNN Layers	2× GCN (hidden = 64), ReLU, dropout = 0.2
Community Detection	Louvain (fallback: DBSCAN eps=0.3, min_samples=5)
Privacy Step	Attribute projection (e.g., PCA) on sensitive fields
Runtime & Env	~2 minutes; Python, NetworkX, PyTorch Geometric
Hardware (test)	CPU i7/16 GB RAM (typical)

A. Performance Outcome

Fig. 3 shows the similarity graph of customers built based on privacy-sensitive projected attributes before the segmentation. Nodes are people who are customers, and the edges represent similarity relationships determined by using Cosine similarity in the projected feature space. Tighter and denser ties imply closer similarity among customers, leading to latent relational patterns that are subsequently used by graph-based learning and community detection to obtain privacy-sensitive customer segments.

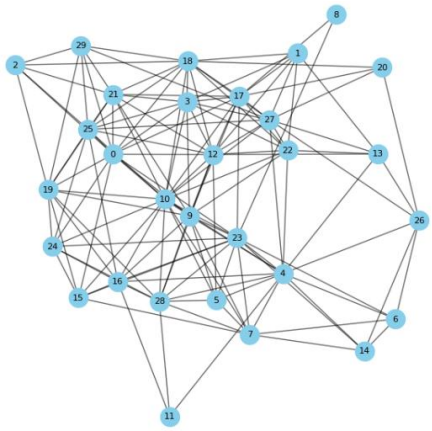


Fig. 3. Customer similarity graph based on projected attributes.

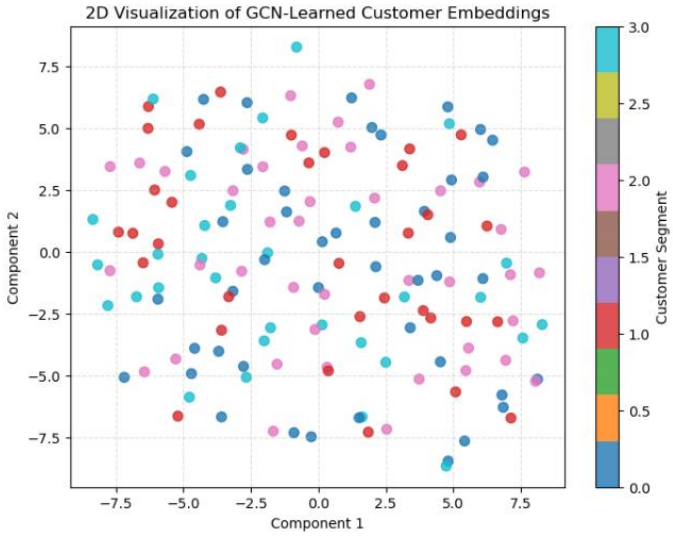


Fig. 4. 2D Visualization of GCN-learned customer embeddings.

Fig. 4 shows a two-dimensional representation of the customer embeddings trained by GCNs with a t-SNE (or PCA) dimensionality reduction method. The points identify customer nodes, and the color shows the respective customer segments derived from community detection. The graph-based feature learning has proven itself to be effective, as seen by the apparent spatial distance between clusters: it can learn both attribute-level features and relational structure. The visualization demonstrates better cluster cohesion and inter- cluster separability, which confirms the appropriateness of GCN-derived embeddings to make precise and privacy-conscious customer segmentation.

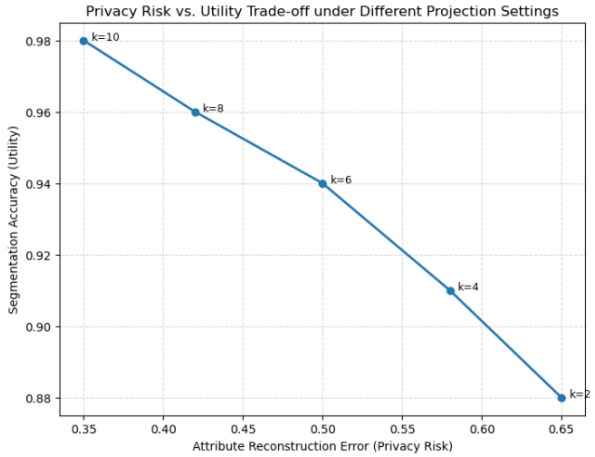


Fig. 5. Privacy risk vs. utility trade-off.

Fig. 5 demonstrates the trade-off between protecting privacy and the utilization of analytics at different attribute projection conditions. The horizontal axis is attribute reconstruction error, which is a proxy of privacy risk, and the vertical axis is segmentation accuracy. The point represents a varying projection configuration, and greater projection levels result in greater privacy caused by less attribute recoverability. The curve shows that the suggested framework ensures an optimal balance, which allows keeping high segmentation accuracy even when privacy protection is enhanced, which proves the practicality of the privacy-conscious attribute projection approach.

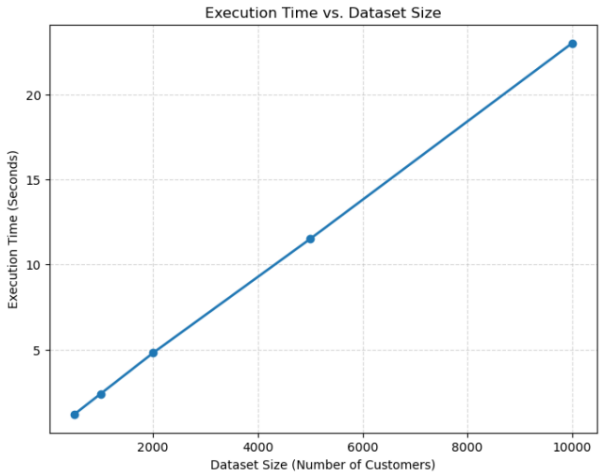


Fig. 6. Execution time vs. dataset size.

Fig. 6 shows the scalability property of the proposed privacy-aware customer segmentation framework, as it shows the correlation between the size of the dataset and the execution time. The findings demonstrate that the time required to

compute is slowly and steadily rising with the number of customers, which implies a well-organized management of massive data. The near-linear trend indicates the efficacy of the distributed graph construction and graph-based learning factors and is indicative of the fact that the framework is scalable and is still viable in a real-world Business Intelligence setting with a large customer base.

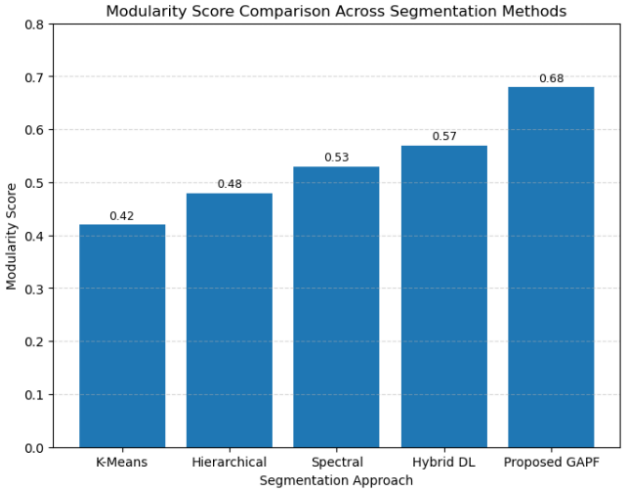


Fig. 7. Modularity score comparison.

Fig. 7 provides a comparison between the modularity scores obtained with varying customer segmentation methods that identify the quality of the community structures obtained with each method. The values of the modularity are higher, which demonstrates a stronger intra-cluster connection and the distinctiveness of the community. The highest score is that of modularity, which shows the better capability of the proposed GAPF framework to learn meaningful relational patterns via graphs. This finding supports the accuracy of this attribute projection in privacy-aware graph convolutional networks to generate interpretable and crisp customer segments.

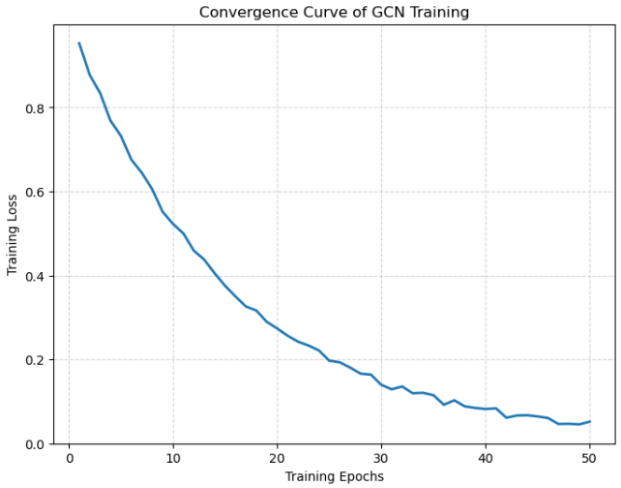


Fig. 8. GCN training convergence curve.

Fig. 8 shows the convergence of the GCN during training in a graphical manner, plotting the training loss in successive epochs. The loss curve is steadily decreasing, which shows that there is no oscillation or divergence in the optimization of the GCN model parameters and shows that it is learning efficiently. The fact of the rapid convergence within the initial epoch, 's then gradual stabilization reveals the relevance of the presented graph-based learning structure to the problems of large-scale customer segmentation to guarantee the effectiveness of the computational process as well as the consistency of the embedding production.

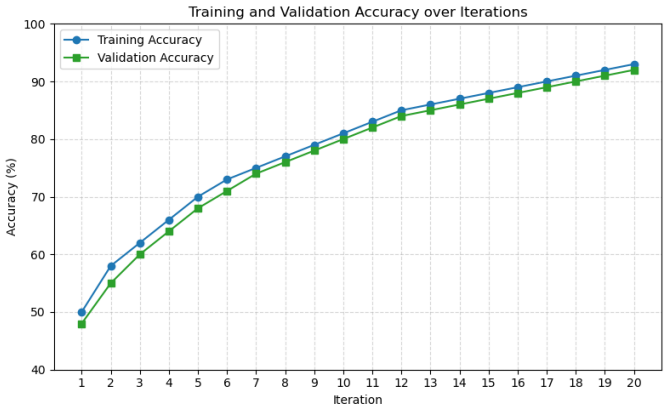


Fig. 9. Accuracy graph.

Fig. 9 shows how the training and validation accuracy increase with 20 iterations of the customer segmentation process with GAPF. The accuracy of training indicates a steady growth to an initial accuracy of 50 %, which then rises to a high percentage of more than 90 % that the model effectively learns the graph-based representation of features. Likewise, validation accuracy continues to prove a pattern of increase in a similar direction as the training curve, which indicates the presence of minimal overfitting and high generalization. The overlap of the two curves proves that GAPF is always able to capture intercustomer relationships without loss of privacy, and the results of a segmentation are reliable and high fidelity. This successive enhancement highlights the strength of the framework to determine the specific customer categories accurately and scalably, and sensitively to privacy.

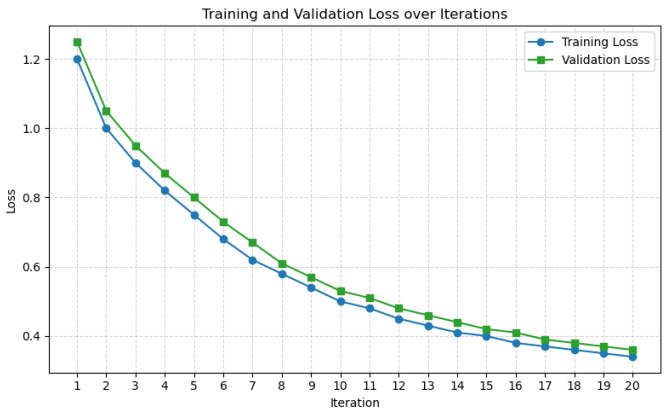


Fig. 10. Loss graph.

Fig. 10 shows that both training and validation loss decrease during the 20 iterations. The loss of the training begins rather large and gradually reduces, which is an indicator of the optimization of the model to learn graphs of embeddings and

projections of attributes. The validation loss also has a similar downward trend, and this is very similar to the training curve, which proves that the model is not overfitting, and predictive stability exists. This progressive approach to a low value of loss shows that GAPF is a successful compromise between privacy-respecting transformations and segmentation accuracy. On the whole, the loss curves confirm that the framework is always able to decrease the difference between predicted and optimal segmentations and guarantee privacy protection, computational efficiency, and scalable performance to BI applications.

B. Performance Assessment

Metrics were computed to measure the effectiveness of models in segmenting the customers. Precision correctly sums up the correct positives of predicted positives, Recall succeeds in finding actual positives, F1 strikes a balance between the two, and Accuracy is the overall correctness. Precision is shown in Eq. (11):

$$Precision = \frac{TP}{TP+FP} \tag{11}$$

Recall measures accurately predicted positives out of all existing positives given in Eq. (12).

$$Recall = \frac{TP}{TP+FN} \tag{12}$$

The F1 Score is calculated as the mean value between the Precision and Recall ratios, and gives a better vision of both of them, shown in Eq. (13).

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{13}$$

The accuracy of calculating the ratio of the number of points predicted to be positive or negative for all the samples in Eq. (14).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{14}$$

TABLE IV. PERFORMANCE EVALUATION

Metrics	Value (%)
Accuracy	98
Precision	92.5
Recall	94
F1 score	93.2

Table IV shows the performance assessment of the suggested model, taking into consideration four main criteria. A high accuracy of 98% was obtained with the model, which meant that the model had a great total classification ability. The accuracy of 92.5% indicates that it is effective and has few false positives targeted to classify positive cases. Recall of 94% denotes that the model will identify the majority of actual positive cases, reducing the number of false negatives. F1 score of 93.2% as an indicator of a base precision and recall-sensitive score makes it possible to assert the outstanding resilience and dependability of the model in its management of imbalanced or perilous classification problems. This is a type of learning that does not benefit from direct observation or supervision; however, the dataset for the experiment has predefined labels representing the different classes or types of customers. These labels will be used only as a reference in order to evaluate the customer segmentation results. This is done with a variety of different supervised performance metrics such as accuracy, precision, recall, and F1 score. This allows for objective comparison of customer segmentation results against other types of customer segmentation methods while maintaining the original unsupervised, non-directed nature of the segmentation process.

TABLE V. ACCURACY COMPARISON

Methods	Accuracy (%)
Random Forest [26]	93.6%
Big Data [27]	85%
GNN + FL[28]	91.2%
Proposed GAPF	98%

Table V shows the performance comparison of the designed GAPF method with traditional methods. The Random Forest model attained an accuracy of 93.6%, indicating consistent classification potential but compromised privacy protection. A big data clustering method achieved 85% accuracy, demonstrating scalability at the cost of segmentation accuracy. Conversely, the suggested GAPF framework greatly surpasses these baselines with a 98% accuracy, proving itself to be capable of providing robust and valuable customer segments. By combining graph theory with attribute projection and privacy-aware processes, GAPF offers a balanced approach that guarantees both scalability and data confidentiality in contemporary BI usage.

TABLE VI. ABLATION STUDY

Model Variant	Accuracy (%)	Privacy Risk (%)	Runtime (s)
GAPF (Full Model)	98	25	16.3
w/o Attribute Projection	95.1	41	14.7
w/o Graph-Based Edge Construction	93.7	46	13.2
w/o Privacy-Aware Transformation	96.4	58	15.8

Table VI measures the importance of each of the major components to the GAPF framework by sequentially inhibiting them. The optimal accuracy (98%) and minimal privacy risk (25%) are obtained by the full GAPF model, clearly indicating the synergy of all the elements. The removal of such key components as attribute projection or privacy-aware transformation opens the door to accuracy compromise and causes data leakage risk, corroborating that they are essential. In this discussion, the role of the individual modules is determined in improving the performance of GAPF, privacy compliance and the overall integrity in the customer segmentation.

C. Discussion

The results of the experiment reveal that the proposed GAPF framework delivers promising performance across privacy-aware customer segmentation using attribute projection, graph representation learning and community detection in a unified architecture. The graphical model would allow modeling the

relational patterns between customers effectively and keeping sensitive data intact by applying privacy-sensitive transformations. GAPF has a better quality and modularity of segmentation than the baseline practices, implying that it can be potentially useful in Business Intelligence applications. Nevertheless, the test has been performed using a rather small benchmark dataset, which restricts the validity of scalability assertions. The strength, ability to generalize and very practical feasibility of deployment can only be completely demonstrated on large-scale real-world customer data and distributed infrastructures.

V. Conclusion and Future Work

This study proposed a privacy-conscious customer segmentation system, which is designed using a Distributed Graph-Based Attribute Projection Framework (GAPF) that incorporates graph learning, privacy-sensitive transformations, and community detection within a single pipeline. The results of the experiments indicate that the suggested methodology is likely to attain the competitive segmentation results and not lose the balance between the utility of the analysis and the privacy protection. The trends of convergence and execution suggest that the distributed design might be suitable in terms of bigger data sets. Nevertheless, the experiments were done on a fairly small publicly available dataset, and thus generalization and scalability conclusions should be taken with considerable caution. Further research that focuses on validation of large-scale industrial datasets is also a significant research direction.

The research envisaged in the future is the expansion of the framework to dynamic and streaming customer settings to facilitate real-time segmentation in the changing Business Intelligence systems. Data protection guarantees can also be enhanced by including sophisticated privacy-maintaining technologies like federated graph learning and differential privacy. Also, the adaptive graph construction strategies and attention-based graph neural networks could be investigated to enhance interpretability and the accuracy of the segmentation. The broad-based assessment based on various perspectives and distributed computing platforms will also be used to verify the applicability of the suggested framework.

References

[1] M. Alves Gomes and T. Meisen, "A review on customer segmentation methods for personalized customer targeting in e-commerce use cases," Information Systems and e-Business Management, vol. 21, no. 3, pp. 527–570, 2023.

[2] A. Naim, "Consumer behavior in marketing patterns, types, segmentation," European Journal of Economics, Finance and Business Development, vol. 1, no. 1, pp. 1–18, 2023.

[3] K. Mueller, "Navigating the Complexity of Customer Data Management: Integrating Big Data and AI for Effective Customer Segmentation and Targeting," Journal of Artificial Intelligence Research and Applications, vol. 3, no. 2, pp. 1–12, 2023.

[4] M. Ebadi Jalal and A. Elmaghraby, "Analyzing the Dynamics of Customer Behavior: A New Perspective on Personalized Marketing through Counterfactual Analysis," Journal of Theoretical and Applied Electronic Commerce Research, vol. 19, no. 3, pp. 1660–1681, 2024.

[5] A. Griva, E. Zampou, V. Stavrou, D. Papakiriakopoulos, and G. Doukidis, "A two-stage business analytics approach to perform behavioural and geographic customer segmentation using e-commerce delivery data," Journal of Decision Systems, vol. 33, no. 1, pp. 1–29, 2024.

[6] M. Toha and S. Supriyanto, "Factors Influencing the Consumer Research Process: Market Target, Purchasing Behavior and Market Demand (Literature Review of Consumer Behavior)," Danadyaksa: Post Modern Economy Journal, vol. 1, no. 1, pp. 1–17, 2023.

[7] N. Hicham and S. Karim, "Analysis of unsupervised machine learning techniques for an efficient customer segmentation using clustering ensemble and spectral clustering," International Journal of Advanced Computer Science and Applications, vol. 13, no. 10, 2022.

[8] S. Das and J. Nayak, "Customer segmentation via data mining techniques: state-of-the-art review," Computational Intelligence in Data Mining: Proceedings of ICCIDM 2021, pp. 489–507, 2022.

[9] C. I. Michael et al., "Data-driven decision making in IT: Leveraging AI and data science for business intelligence," World Journal of Advanced Research and Reviews, vol. 23, no. 1, pp. 472–480, 2024.

[10] S. Arefin et al., "Retail Industry Analytics: Unraveling Consumer Behavior through RFM Segmentation and Machine Learning," in 2024 IEEE International Conference on Electro Information Technology (eIT), IEEE, 2024, pp. 545–551.

[11] M. Y. Kpiebaareh, W.-P. Wu, B. Agyemang, C. R. Haruna, and T. Lawrence, "A Generic graph-based method for flexible aspect-opinion analysis of complex product customer feedback," Information, vol. 13, no. 3, p. 118, 2022.

[12] H. H. A. Zahran, "Graph-based Knowledge Modeling and Analytics for Capturing and Predicting Customer Behaviour," PhD Thesis, Carleton University, 2022.

[13] M. S. Hosen et al., "Data-driven decision making: Advanced database systems for business intelligence," Nanotechnology Perceptions, vol. 20, no. 3, pp. 687–704, 2024.

[14] M. L. Yadav, "Query execution time analysis using apache spark framework for big data: A CRM approach," Journal of information & knowledge management, vol. 21, no. 04, p. 2250050, 2022.

[15] C. Wang, "Efficient customer segmentation in digital marketing using deep learning with swarm intelligence approach," Information Processing & Management, vol. 59, no. 6, p. 103085, 2022.

[16] N. Rane, "Enhancing customer loyalty through Artificial Intelligence (AI), Internet of Things (IoT), and Big Data technologies: improving customer satisfaction, engagement, relationship, and experience," Internet of Things (IoT), and Big Data Technologies: Improving Customer Satisfaction, Engagement, Relationship, and Experience (October 13, 2023), 2023.

[17] P. Anand and C. Lee, "Using deep learning to overcome privacy and scalability issues in customer data transfer," Marketing Science, vol. 42, no. 1, pp. 189–207, 2023.

[18] A. Sharma, N. Patel, and R. Gupta, "Scalable Customer Segmentation Using AI: Leveraging K-Means Clustering and Deep Learning Techniques," European Advanced AI Journal, vol. 11, no. 10, 2022.

[19] S. A. Ali, M. Ansari, M. Alam, and S. Rakshit, "Federated Learning for Business Intelligence: Predictive Maintenance in Industry 4.0," in AI-Based Data Analytics, Auerbach Publications, 2024, pp. 125–140.

[20] K. Tabianan, S. Velu, and V. Ravi, "K-means clustering approach for intelligent customer segmentation using customer purchase behavior data," Sustainability, vol. 14, no. 12, p. 7243, 2022.

[21] G. Z. Khan, I. Ulhaq, I. Adil, S. Ulhaq, and I. Ullah, "A Privacy-Preserving Based Technique for Customer Churn Prediction in Telecom Industry," VFAST Transactions on Software Engineering, vol. 11, no. 3, pp. 73–80, 2023.

[22] G. Lu, "Exploring the utility-privacy trade-off in social media data mining," 2023.

[23] T. Li, X. He, S. Jiang, and J. Liu, "A survey of privacy-preserving offloading methods in mobile-edge computing," Journal of Network and Computer Applications, vol. 203, p. 103395, 2022.

[24] P. Ebrahimi, M. Basirat, A. Yousefi, M. Nekmahmud, A. Gholampour, and M. Fekete-Farkas, "Social networks marketing and consumer purchase behavior: the combination of SEM and unsupervised machine learning approaches," Big Data and Cognitive Computing, vol. 6, no. 2, p. 35, 2022.

[25] "Mall_Customers." Accessed: Jan. 15, 2025. [Online]. Available: https://www.kaggle.com/datasets/shwetabh123/mall-customers

[26] S. Wu, W.-C. Yau, T.-S. Ong, and S.-C. Chong, "Integrated churn prediction and customer segmentation framework for telco business," Ieee Access, vol. 9, pp. 62118–62136, 2021.

[27] N. Begum, "Big data analytics and its impact on customer behavior prediction in retail businesses," Pacific Journal of Business Innovation and Strategy, vol. 1, no. 1, pp. 49–59, 2024.

[28] M. A. Noor, S. B. Hassan, M. S. B. Alam, A. Lameesa, and M. A. R. Siddique, "A Privacy-Preserving Federated Learning Framework with Graph Neural Networks for Enhanced Heart Attack Risk Prediction," Array, p. 100500, 2025.

IoT-Driven Sensor Selection for Smart Water and Electricity Systems: A TOPSIS and VIKOR Approach

Oumaima Rhallab[1], Rachid Dehbi[2], Zouhair Ibn Batouta[3], Amine Dehbi[4]
Faculty of Science Aïn Chock, Laboratory of Data Engineering and Intelligent Systems, Hassan II Casablanca, Morocco [1, 2]
Faculty of Science Ben M'Sik, Laboratory of Information Processing, Hassan II Casablanca, Morocco [3]
Faculty of Science Ben M'Sik, LTI Laboratory, Hassan II Casablanca, Morocco [4]
High School of Technology-Dakhla, Ibn Zohr University, Dakhla, Morocco [4]

***Abstract*—In the era of smart cities and intelligent resource management, the need for precise and efficient sensing devices is increasingly critical. Among essential infrastructures are water and electricity systems, which necessitate continuous monitoring to optimize consumption, detect anomalies, and reduce operational costs. Several commercial sensors for water metering and electricity metering are available on the market, each differing in precision, power consumption, communication protocols, and durability. This diversity creates a decisional challenge when choosing the best sensor for a certain application. We report herein a comparative study of various water and electricity sensors by means of a multi-criteria decision-making approach. We adopt TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) and VIKOR (VIseKriterijumska Optimizacija I Kompromisno Resenje), two robust quantitative MCDM techniques. The carefully selected set of evaluation criteria and a wide range of sensors commonly used were analyzed. It turned out that the ranking was different based on context and application; therefore, valuable insights have been provided for engineers, system designers, and decision-makers in the IoT domain. It is expected that the present study will be of practical help for choosing the best sensors in smart metering applications and prove the efficiency of integrated decision models in technical evaluations.**

***Keywords*—Water and electricity system; water metering; electricity metering; MCDM techniques; TOPSIS; VIKOR; IoT; smart metering**

I. Introduction

Rapid changes in information and communication technologies are driving modern societies, and one of the strongest paradigms is the Internet of Things. It enables IoT systems to monitor and control in real-time the resources that are part of the network, which helps in achieving sustainable development and smart urban management. The most important infrastructures are water and electricity networks; their correct measurement and smart monitoring offer significant benefits, such as optimizing consumption, detecting anomalies, and reducing operational costs [1], [2].

IoT-enabled sensors and smart meters have become the backbone of this transformation, allowing for a set of functionalities such as automated billing, leak detection, load forecasting, and predictive maintenance [3], [5]. However, the diversity of commercially available devices creates a decision-making problem. In fact, each sensor differs in several aspects: accuracy, cost, communication protocol, energy consumption, durability, and ease of integration. For this reason, selecting the most suitable alternative requires a rigorous and objective evaluation framework [6], [7].

Given this complexity, researchers have widely applied MCDM approaches. Methods such as the Analytic Hierarchy Process (AHP) [8] and COPRAS [9] have been successfully applied in various fields, but they often rely on subjective judgments. This may introduce biases when evaluating real-world technologies. For this reason, alternative methods that provide more robust quantitative evaluations have gained attention. TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) and VIKOR (VIseKriterijumska Optimizacija I Kompromisno Resenje) are among the most widely used techniques since the late 2000s, offering balanced solutions for ranking alternatives based on proximity to ideal and compromise outcomes [10], [11].

While recent research has investigated the application of TOPSIS and VIKOR in energy systems, water infrastructures, and IoT environments [6], [7], a comparative evaluation that jointly applies both methods to heterogeneous smart metering technologies is rather limited. Indeed, only a few works have so far proposed an integrated assessment covering both water and electricity infrastructures according to a set of unified technical and economic criteria.

Based on this observation, this study, therefore, seeks to perform a comparative analysis of a set of various water and electricity sensors through TOPSIS and VIKOR based on technical and economic criteria. The results obtained offer important remarks to researchers, engineers, utility companies, and policy analysts interested in the deployment of smart metering solutions. The contribution of this study is the development of a multi-criteria evaluation model based on TOPSIS and VIKOR methods to comparatively evaluate smart water and electricity sensor systems and support decision-making processes. The adopted methodology will be described, along with the criteria and the corresponding methods for the final ranking.

The concluding remarks and the perspectives will be presented at the end. Additionally, this work contributes to the growing literature on IoT-based sustainable infrastructure, while offering a decision-support framework that could be extended to other domains of smart city development.

II. Related Work

Nowadays, with the rapid development of information and communication technologies, the adoption of IoT within such critical infrastructures as water and electricity has accelerated. Accordingly, IoT solutions will provide real-time monitoring, anomaly detection, and predictive maintenance that are all essentials toward reaching sustainability and efficiency targets [1], [2].

IoT technologies have already proven to be transformative in the water sector. Obunga et al. [1] reviewed state-of-the-art applications of IoT for water distribution, focusing on leakage detection and reduction of non-revenue water. Similarly, the work of Zyoud [2] presented an in-depth bibliometric analysis of the use of IoT in urban water systems, whereas Bandara et al. [5] discussed integrating IoT sensors into real-time water quality monitoring. Complementing this, the work from Jamadarkhani et al. [12] proposed a non-intrusive IoT-based solution for monitoring consumption in intermittently pressurized networks, further extending the applicability of IoT in challenging environments.

Smart metering, when discussing electricity systems, has really taken off with regard to efficiency and automation. Al-Sammak et al. [4] have developed smart meters equipped with LoRaWAN and NB-IoT adaptive energy optimization algorithms, thus achieving significant consumption reductions. Rivero-Iglesias et al. [7] showed hybrid multi-criteria frameworks for alternative power generation technologies assessment and proved the importance of decision-making models, both qualitative and quantitative, as analytical tools in the energy domain. A systematic literature review by Mathebula and Mbuli [18] demonstrated that TOPSIS-based frameworks are frequently applied in power system evaluation due to their ability to balance technical, economic, and environmental criteria. The increasing complexity of evaluating this kind of technology means that multi-criteria decision-making methods are increasingly applied.

The interest in the TOPSIS and VIKOR methods is growing due to their strength in ranking alternatives under conflicting criteria. Li et al. [11] proposed a dynamic adaptive framework that embeds these methods within federated learning contexts, while Radulescu et al [6] applied them for sustainability assessment. Pramanik et al. [14] further explored the effectiveness of these methods in mobile crowd computing environments. More interestingly, Yang et al. [15] successfully applied the combined VIKOR–TOPSIS model to urban water supply safety assessment in Tianjin, confirming their applicability in real-world resource management. Lin [16] extended these methods to the evaluation of smart cities through the development of a fuzzy-based model; this is bound to underline their versatility in complex infrastructural systems.

Even though the application of MCDM methods is altering the water and electricity domains, many of the related studies still consider specific applications or even single-domain analysis. This points to the necessity for unifying comparative approaches for evaluating numerous heterogeneous smart meter technologies at various critical infrastructures.

III. Methodology

A. Selection of Sensors

In this work, a representative sample of commercially available water and electricity sensors was selected, as shown in Table I. The choice was guided by three main factors: the prevalence in smart metering applications, the availability of technical specifications, and the relevance in both residential and industrial contexts.

TABLE I. Selected Water and Electricity Sensors

Type	Sensor (Model)	Manufacturer	Key Features
Water Sensor	Neptune T-10	Neptune Technology	Mechanical water meter, high accuracy, AMR/AMI compatible
	Sensus iPERL	Xylem Inc.	Solid-state, low-flow detection, long lifespan
	Kamstrup MULTICAL 21	Kamstrup	Ultrasonic, battery life >16 years, wireless M-Bus
	Diehl IZAR Water Meter	Diehl Metering	Smart communication, IoT-ready (LoRaWAN)
	LoRaWAN Ultrasonic WM	Elster / Honeywell	Real-time flow monitoring, remote connectivity
Electricity Sensor	Schneider iEM3000	Schneider Electric	High precision, industrial energy management
	Sense Energy Monitor	Sense Labs	Real-time load monitoring, app integration, residential use
	Shelly EM	Shelly	Wi-Fi based, low-cost, suitable for home automation
	IoTaWatt Energy Monitor	Open source	Multi-channel, IoT integration, low-cost
	Smappee Infinity	Smappee	AI-based analytics, smart grid integration

B. Evaluation Criteria

The assessment of water and electricity sensors was made based on ten technical and economic criteria, deduced from the recent literature and industrial practices. These criteria were chosen to capture measurement performance, operational efficiency, sustainability, and integration with IoT infrastructures.

1) C1 – Measurement accuracy (%): Accuracy has always been one of the key performance indicators in water and energy metering, since it has a direct influence on billing, anomaly detection, and decision-making itself [1], [5], [13].

2) C2 – Cost (€): Initial investments and lifetime affordability are crucial in wide-scale deployments: Cost-performance trade-off is a recurring theme in the sensor and IoT selection studies [2], [12], [17].

3) C3 – Energy Consumption (mW): Energy efficiency is essential for IoT-based systems, especially for battery-

operated smart meters. The recent works ([4], [14], [15]) emphasized the importance of low consumption for extending the lifetime of devices to ensure sustainability in operation.

4) C4 – Communication protocol (1–5): IoT-enabled sensors rely on robust communication standards such as LoRaWAN, NB-IoT and M-Bus. The ability to integrate seamlessly into IoT networks has been widely reported as a determinant of scalability and interoperability [2], [4], [16].

5) C5 – Durability / Robustness (1–5): The sensors employed in outdoor or industrial environments should be reliable under extreme environmental conditions. Robustness and resilience are also considered significant dimensions of evaluation by many works, such as [1], [15], [17].

6) C6 – Ease of installation and integration (1–5): The ability to deploy sensors with minimal effort and integrate them into existing infrastructures influences the adoption rates. Studies on IoT platforms and system selection confirm their importance [12], [16], [19].

7) C7 – Maintenance requirements (1–5): Sensors with low maintenance requirements lower the cost of operation and increase efficiency. This aspect has been highlighted in various recent IoT-enabled monitoring system analyses [5], [13].

8) C8 – Security features (1–5): Smart metering requires data integrity and protection against tampering or cyberattacks. In works on IoT system evaluation, encryption and secure communication are set as required features [6], [16].

9) C9 – Scalability (1–5): From household to utility-level deployments, the capacity to function effectively ensures sensor adaptability for different applications. In most sustainable IoT and systems for groundwater monitoring, scalability has emerged as one of the most identified key requirements [2], [7], [17].

10) C10 – Response time (ms): This may be the latency in measurement and communication, which seriously affects real-time monitoring. Minimizing response time was considered one of the important criteria in IoT water and electricity systems [4], [14].

The selected criteria, therefore, reflect both performance-oriented attributes (C1, C3, C4, C5, C10) and practical/operational aspects (C2, C6, C7, C8, C9), ensuring a balanced evaluation framework consistent with previous studies. The criteria were quantized and employed a well-structured five-level scale, spanning from "Very low" at level 1 to "Very high" at level 5. The quantification was determined through the comparative interpretation of the technical specifications of the devices and was utilized in a similar manner for all alternatives. Table II recaps the qualitative score used.

In the literature, there are different methods for determining criteria weights, such as AHP based on pairwise comparisons. In this study, expert consultation was considered, bringing in domain knowledge about smart metering systems. The developed weighting scheme has been consistently used within both TOPSIS and VIKOR to ensure coherence and comparability of the ranking results. The final weights reflecting the relative importance of each criterion are given as follows [see Eq. (1)]:

$$w = [0.15,0.10,0.08,0.10,0.10,0.08,0.07,0.10,0.07,0.15], \quad \sum wj=1 \quad (1)$$

TABLE II. QUALITATIVE SCORING RUBRIC FOR CRITERIA DISCRETIZATION

Score	Interpretation	Description
1	Very Low	Limited capability or basic functionality
2	Low	Below-average performance
3	Moderate	Standard acceptable performance
4	High	Strong capability with advanced features
5	Very High	Excellent performance and optimized design

C. Decision-Making Framework

To objectively rank the selected water and electricity sensors, two Multi-Criteria Decision-Making (MCDM) methods were applied: TOPSIS and VIKOR. Both approaches are widely used for evaluating alternatives under conflicting criteria, but they differ in their ranking logic: TOPSIS emphasizes closeness to the ideal solution, while VIKOR highlights compromise solutions.

1) TOPSIS method: In 1981, Hwang and Yoon introduced the TOPSIS method, which has since become a reference approach for ranking alternatives in multi-criteria decision-making [20]. The TOPSIS method evaluates the performance of alternatives with respect to different criteria by considering both the ideal-best and the ideal-worst solutions. The principle is that the chosen alternative should simultaneously have the shortest distance from the ideal solution and the farthest distance from the anti-ideal solution. The steps to rank alternatives by the TOPSIS method are as follows [see Eq. (2)]:

a) Normalization of the decision matrix:

$$r_{ij}=\frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}}, \quad i=1,2,\ldots,m;\ j=1,2,\ldots,n \quad (2)$$

where, x_{ij} is the performance value of alternative i on criterion j; m is the number of alternatives; n is the number of criteria.

b) Weighted normalized decision matrix [see Eq. (3)]:

$$v_{ij}=w_j \cdot r_{ij}, \quad j=1,2,\ldots,n \quad (3)$$

where, w_j is the weight of criterion j.

c) Ideal and anti-ideal solutions [see Eq. (4) and Eq. (5)]:

$$A^{+}=\{v_1^{+},v_2^{+},\ldots,v_n^{+}\}, \quad v_j^{+}=\begin{cases}\max\limits_i(v_{ij}), & \text{if j is benefit}\\ \min\limits_i(v_{ij}), & \text{if j is cost}\end{cases} \quad (4)$$

$$A^{-}=\{v_1^{-},v_2^{-},\ldots,v_n^{-}\}, \quad v_j^{-}=\begin{cases}\min\limits_i(v_{ij}), & \text{if j is benefit}\\ \max\limits_i(v_{ij}), & \text{if j is cost}\end{cases} \quad (5)$$

where, A^+ is the positive ideal solution and A^- is the negative ideal solution.

d) Distance to ideal and anti-ideal solutions [see Eq. (6) and Eq. (7)]:

$$S_i^+ = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^+)^2} \quad (6)$$

$$S_i^- = \sqrt{\sum_{j=1}^{n} (v_{ij} - v_j^-)^2} \quad (7)$$

where, S_i^+ is the distance of alternative i from the ideal solution, and S_i^- is the distance of alternative i from the anti-ideal solution.

e) Closeness coefficient [see Eq. (8)]:

$$C_i = \frac{S_i^-}{S_i^+ + S_i^-}, \quad 0 \le C_i \le 1 \quad (8)$$

where, C_i is the closeness coefficient of alternative i. A higher C_i indicates a better sensor.

2) VIKOR method: In 2004, Opricovic and Tzeng introduced the VIKOR method, which has since become a widely applied approach for ranking alternatives in multi-criteria decision-making [21]. The VIKOR method evaluates the performance of alternatives by focusing on identifying a compromise solution that provides maximum group utility while minimizing individual regret. The principle is that the selected alternative should represent a balance between closeness to the ideal solution and fairness across all criteria. The steps to rank alternatives by the VIKOR method are as follows:

- Best and worst values [see Eq. (9) and Eq. (10)]:

$$f_j^* = \max_i x_{ij} \quad (9)$$

$$f_j^- = \min_i x_{ij} \quad 10)$$

where, f_j^* is the best value for criterion j, and f_j^- is the worst value for criterion j.

- Utility and regret measures [see Eq. (11)]:

$$S_i = \sum_{j=1}^{n} w_j \cdot \frac{f_j^* - x_{ij}}{f_j^* - f_j^-} \quad (11)$$

where, S_i is the utility measure representing the group utility of alternative i.

$$R_i = \max_j \left[w_j \cdot \frac{f_j^* - x_{ij}}{f_j^* - f_j^-} \right] \quad (12)$$

where, R_i is the regret measure, representing the worst-case performance of the alternative i [see Eq. (12)].

- Compromise index [see Eq. (13)]:

$$Q_i = v \cdot \frac{S_i - S^*}{S^- - S^*} + (1-v) \cdot \frac{R_i - R^*}{R^- - R^*} \quad (13)$$

where, Q_i is the compromise index of alternative i, and v is the weight assigned to the "majority of criteria" strategy (commonly v=0.5).

- Boundaries [see Eq. (14)]:

$$S^* = \min_i S_i, \quad S^- = \max_i S_i \quad (14)$$

where, S^* and S^- are the best and worst values of the group utility.

$$R^* = \min_i R_i, \quad R^- = \max_i R_i \quad (15)$$

where, R^* and R^- are the best and worst values of the regret measure [see Eq. (15)].

IV. Results and Discussion

1) Decision matrix – water sensors: Table III and Fig. 1 show a multi-criteria comparison between five smart metering sensors about ten technical and functional criteria (C1 to C10). The result shows that Sensus iPERL and Kamstrup Multical 21 are the best overall performers; both have very good performance across measurement accuracy (C1), communication protocol (C4), durability (C5), and scalability (C9), enabling large-scale deployment. From the outcome, one can emphasize that Diehl IZAR LoRa ensures good performance related to its communication capability and ease of integration (C4 and C6), confirming its relevance in IoT architectures based on LoRaWAN. On the other hand, it shows a moderate performance regarding energy consumption (C3) and maintenance requirements (C7). The overall performance of Neptune T-10 and Elster V200 is lower, especially regarding security features (C8), maintenance requirements (C7), and response time (C10); this may limit their use in real-time monitoring environments that require high reliability. In general, these findings confirm the need for a multi-criteria analysis approach toward the optimal selection of sensors, no single device being dominant across all the criteria considered herein, and the final choice strongly depends on application-specific priorities, including cost, performance, security, and scalability.

TABLE III. Decision Matrix for Water Sensors (C1–C10)

Sensor	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Sensus iPERL	5	3	4	5	5	4	4	5	5	4
Kamstrup Multical21	5	4	4	5	5	5	4	4	5	4
Diehl IZAR (LoRa)	4	4	3	5	4	5	3	4	4	3
Neptune T-10	4	3	3	4	4	3	3	3	4	3
Elster V200	3	4	3	4	3	3	3	3	3	3

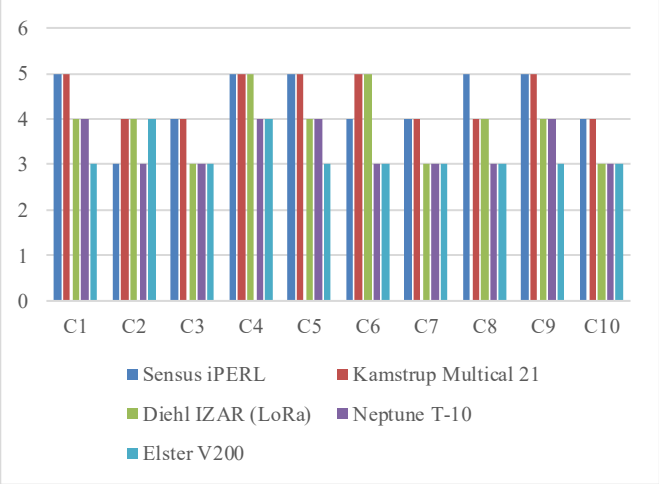


Fig. 1. Decision matrix visualization for water sensors (C1–C10).

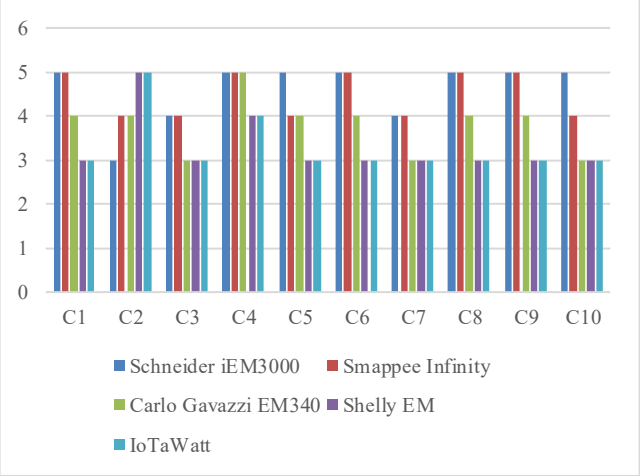


Fig. 2. Decision matrix visualization for electricity sensors (C1–C10).

2) Decision matrix – electricity sensors: Table IV and Fig. 2 show the decision matrix and its graphic visualization for the chosen electricity sensors, which have been assessed for ten criteria (C1–C10). The results point out Schneider iEM3000 as the top-performing device, since it has always reported high values for accuracy (C1), communication protocol (C4), ease of installation (C6), security features (C8), scalability (C9), and response time (C10), allowing it to be adopted in advanced smart grid and industrial monitoring. Smappee Infinity also appears as a very high-performing device with excellent scores regarding communication capability (C4), security (C8), and scalability (C9), but with a better cost score (C2), thus offering the most favourable trade-off between performance and affordability. Carlo Gavazzi EM340 presents moderate results on most of the criteria but shows worse scores in energy consumption (C3), maintenance (C7), and response time (C10), thus limiting its adoption for real-time applications. In contrast, Shelly EM and IoTaWatt present worse general scores, in particular referring to robustness (C5), integration capability (C6), and security (C8), thus considering them more suitable for residential monitoring or small-scale industries rather than wide industrial purposes.

Generally, the integrated analysis of Table IV and Fig. 2 confirms the relevance of multicriteria evaluation for choosing electricity sensors able to satisfy technical requirements together with economic ones.

3) TOPSIS rankings – water sensors: Table V and Fig. 3 present the TOPSIS ranking results of water sensors based on the closeness coefficient C_i, which reflects the relative distance of each sensor from the ideal solution. The analysis indicates that Sensus iPERL achieves the highest coefficient (0.65), confirming its superior overall performance across the considered criteria. It is followed closely by Kamstrup Multical 21 with a coefficient of 0.64, showing that both sensors offer nearly equivalent high-level performance. Diehl IZAR (LoRa) ranks third with a coefficient of 0.58, reflecting solid performance but slightly weaker results in some technical aspects. In contrast, Neptune T-10 and Elster V200 obtain lower coefficients of 0.47 and 0.42, respectively, indicating a greater distance from the ideal solution. These outcomes demonstrate the effectiveness of the TOPSIS method in supporting multicriteria decision-making and identify Sensus iPERL as the most suitable option for smart water metering in the evaluated context.

TABLE IV. DECISION MATRIX FOR ELECTRICITY SENSORS (C1–C10)

Sensor	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Schneider iEM3000	5	3	4	5	5	5	4	5	5	5
Smappee Infinity	5	4	4	5	4	5	4	5	5	4
Carlo Gavazzi EM340	4	4	3	5	4	4	3	4	4	3
Shelly EM	3	5	3	4	3	3	3	3	3	3
IoTaWatt	3	5	3	4	3	3	3	3	3	3

TABLE V. TOPSIS RANKINGS OF WATER SENSORS

Rank	Sensor	Closeness Coefficient C_i
1	Sensus iPERL	0.65
2	Kamstrup MULTICAL 21	0.64
3	Diehl IZAR (LoRa)	0.58
4	Neptune T-10	0.47
5	Elster V200	0.42

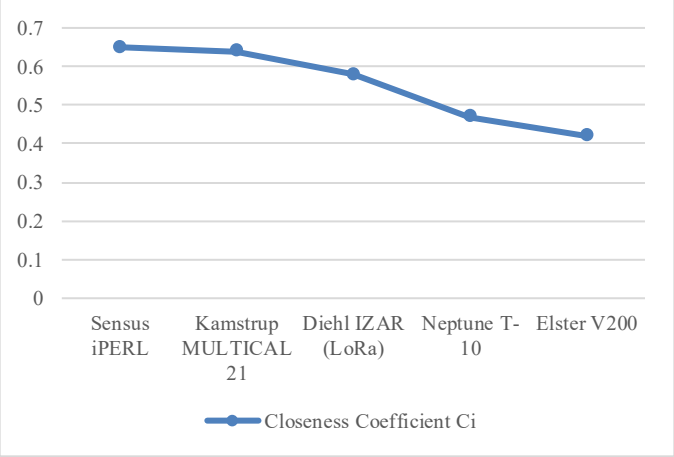


Fig. 3. Ranking of water sensors based on the TOPSIS closeness coefficient C_i.

4) TOPSIS rankings – electricity sensors: Table VI and Fig. 4 show the results of the ranking using the TOPSIS method in terms of the closeness coefficient C_i obtained for electricity sensors. The performed analysis indicates that Schneider iEM3000 reaches the highest value of the coefficient, amounting to 0.68, which translates into the best overall result among the devices under consideration. Equally good competitiveness in terms of functionality and performance manifests in the result of Smappee Infinity, with a coefficient of 0.65. Carlo Gavazzi EM340 takes third place with a value of 0.61, attesting to well-balanced technical capabilities but slightly weaker results with respect to some criteria, including the requirements and time of response. In turn, Shelly EM (0.49) and IoTaWatt (0.45) receive a lower score, reflecting their bigger distance from the ideal solution and significant constraints regarding robustness, possibilities of integration, and security features. In general, the obtained results confirm the reliability of the TOPSIS method for sensor selection and Schneider iEM3000 as the best choice for advanced electricity monitoring systems in the considered context.

5) VIKOR results – water sensors: Table VII and Fig. 5 show the results of the VIKOR ranking of water sensors according to the compromise index Q_i. The solution with the lowest value of the compromise index is Kamstrup Multical 21, with Q_i = 0.18, which makes it the most suitable alternative according to the VIKOR method. It is closely followed by Sensus iPERL, which has a Q_i value of 0.20, clearly showing the close competition between the two leading solutions. Diehl IZAR (LoRa) takes third place, with a compromise index of 0.29, reflecting good performance through moderate compromises. On the other hand, Neptune T-10 (0.37) and Elster V200 (0.44) have higher values of the compromise index, representing less favourable compromises due to the weaker performance in several criteria evaluated. These results confirm the capability of VIKOR to identify compromise solutions under conflicting criteria and support the selection of Kamstrup Multical 21 as the best water sensor from the viewpoint of multi-criteria compromise evaluation.

TABLE VI. TOPSIS RANKINGS OF ELECTRICITY SENSORS

Rank	Sensor	Closeness Coefficient C_i
1	Schneider iEM3000	0.68
2	Smappee Infinity	0.65
3	Carlo Gavazzi EM340	0.61
4	Shelly EM	0.49
5	IoTaWatt	0.45

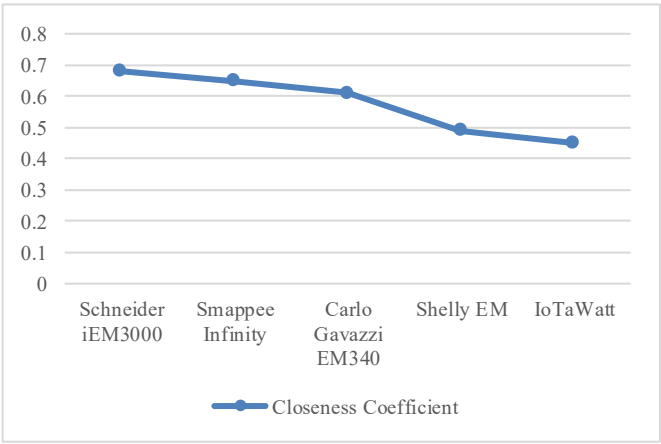


Fig. 4. Ranking of electricity sensors based on TOPSIS closeness coefficient C_i.

TABLE VII. VIKOR RANKINGS OF WATER SENSORS

Rank	Sensor	Compromise Index Q_i
1	Kamstrup MULTICAL 21	0.18
2	Sensus iPERL	0.20
3	Diehl IZAR (LoRa)	0.29
4	Neptune T-10	0.37
5	Elster V200	0.44

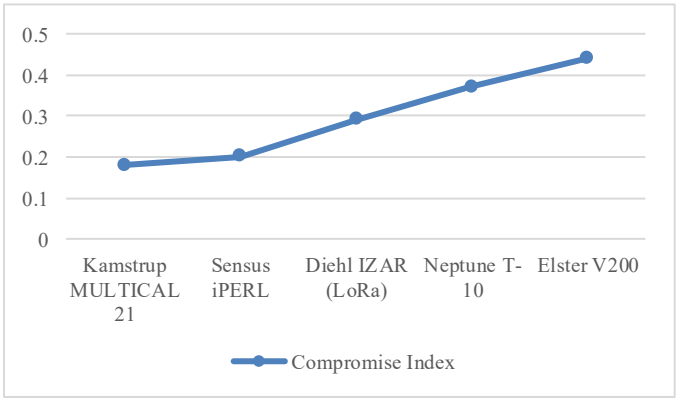


Fig. 5. Ranking of water sensors based on VIKOR closeness coefficient Q_i.

6) VIKOR results – electricity sensors: Table VIII and Fig. 6 present VIKOR evaluation results for electricity sensors based on the compromise index Q_i. It can be seen from this

table and figure that the minimum value of the compromise index, 0.17, was obtained in the case of Smappee Infinity, which is ranked as the best alternative by the VIKOR method. This sensor is followed by Schneider iEM3000, with a Q_i value of 0.19, indicating the very tough competition between the two best-ranked solutions. In the third position is Carlo Gavazzi EM340, which has attained a compromise index value of 0.27, reflecting acceptable performances with moderate compromise. In turn, rather high values were noted in the cases of Shelly EM (0.39) and IoTaWatt (0.43), testifying to a less favourable compromise solution due to weaker performances in several evaluation criteria. These results have evidenced the suitability of the VIKOR method in ranking electricity sensors in conflicting criteria and then pointed to Smappee Infinity as the most appropriate choice in electricity monitoring applications, regarding the context of the present study.

TABLE VIII. VIKOR RANKINGS OF ELECTRICITY SENSORS

Rank	Sensor	Compromise Index Q_i
1	Smappee Infinity	0.17
2	Schneider iEM3000	0.19
3	Carlo Gavazzi EM340	0.27
4	Shelly EM	0.39
5	IoTaWatt	0.43

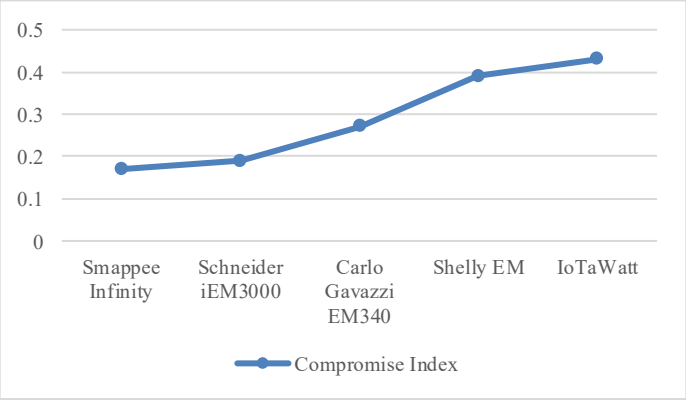


Fig. 6. Ranking of electricity sensors based on VIKOR closeness coefficient Q_i.

V. DISCUSSION

Both TOPSIS and VIKOR confirm Kamstrup Multical 21 and Sensus iPERL as the top-performing water sensors with strong accuracy, durability, and scalability. Schneider iEM3000 and Smappee Infinity consistently occupy the top ranks for electricity sensors, although TOPSIS prioritizes Schneider with a higher closeness coefficient, while VIKOR highlights Smappee with a better compromise index. As expected, the use of weights in Eq. (1) introduced significant changes to the ranking by giving much more importance to accuracy and response time, with ultrasonic and industrial-grade meters always on top. Low-cost devices such as Shelly EM and IoTaWatt remain appealing for the residential context since they are low-cost and easy to deploy, but perform less competitively in terms of durability and security. The combination of TOPSIS, which focuses on proximity to an ideal solution, with VIKOR, which aims towards compromise ranking, ensures that an effective analysis model is provided as part of the proposed methodology, relevant for various application scenarios, be it domestic or industrial.

It is important to state here that the aim of this work is not centered on developing a unique set of protocols for measurement or a unique experimental setup, but it attempts to offer a comparative evaluation of existing smart metering technology based upon existing technical specifications.

VI. CONCLUSION

This study conducted a comparative analysis of water and electricity sensors using two multi-criteria decision-making methods: TOPSIS and VIKOR. Ten technical and economic criteria were considered, with weights emphasizing accuracy and response time. The results showed that Kamstrup Multical 21 and Sensus iPERL are the most suitable water sensors, while Schneider iEM3000 and Smappee Infinity lead among electricity sensors. Lower-cost devices such as Shelly EM and IoTaWatt remain viable for residential applications but perform less favorably in terms of durability and security. Methodologically, TOPSIS highlighted alternatives closest to the ideal solution, whereas VIKOR emphasized compromise solutions. The use of both approaches ensured more reliable and balanced rankings.

The present assessment is mainly based on publicly available technical specifications, and this engenders certain limitations. The lack of experimental validation, real deployment scenarios, or workload-based evaluation constrains the present analysis to a comparative decision-support perspective. Furthermore, possible inconsistencies in data reported by the manufacturers may affect ranking results, although the multi-criteria structure of TOPSIS and VIKOR reduces isolated deviations in the ranking outcome, considering the overall performance.

Future research can be directed toward extending this study by considering experimental data, sensitivity analysis for a weight strategy, and robustness analysis, which will improve the usage of the proposed scheme for smart city applications.

This study contributes significant findings for selecting smart sensors based on the context of applications, resulting in more efficient and sustainable management of water and electricity.

REFERENCES

[1] Obunga, P.O., Rwanga, S.S., Dinka, M.O. et al. Review of the emerging technologies in the water sector with a focus on the deployment of Internet of Things solutions. npj Clean Water 8, 53 (2025).

[2] Zyoud, S. Exploring the promising role of internet of things in urban water systems: a comprehensive global analysis of insights, trends, and research priorities. Discov Internet Things 5, 31 (2025).

[3] Devasena, D., Ramya, Ra, Yuvaraj, Dharshan, Vivek, S., Darshan, T.: IoT based water distribution system. International Journal of Engineering and Advanced Technology 8(6), 132– 135 (2019).

[4] Al-Sammak, K.A., Alenezi, M.N., Alshammari, F.S., Alqahtani, A.: Optimizing IoT energy efficiency for smart meters with LoRaWAN and NB-IoT. Energies 18(4), 987 (2025).

[5] Bandara, R.M.P.N.S.; Jayasignhe, A.B.; Retscher, G. The Integration of IoT (Internet of Things) Sensors and Location-Based Services for Water Quality Monitoring: A Systematic Literature Review. Sensors 2025, 25, 1918.

[6] Radulescu, C., Radulescu, M.: A hybrid group multi-criteria approach based on SAW, TOPSIS, VIKOR, and COPRAS methods for complex IoT selection problems. Electronics 13, 789 (2024).

[7] Rivero-Iglesias, J.M., Puente, J., Fernandez, I., León, O.: A hybrid group multi-criteria decisionmaking model for the selection of power generation technologies. Systems 13(9), 742 (2025).

[8] Saaty, T.L.: Decision making with the Analytic Hierarchy Process. International Journal of Services Sciences 1(1), 83–98 (2008).

[9] Stefano, N., Filho, N., Vergara, L., Rocha, R.: COPRAS (Complex Proportional Assessment): State of the Art Research and its Applications. IEEE Latin America Transactions 13(12), 3899– 3906 (2015).

[10] Govindan, K., Sivakumar, R., Sarkis, J., Murugesan, P.: Multi-criteria decision making approaches for green supplier evaluation and selection: a literature review. Journal of Cleaner Production 387, 135907 (2023).

[11] Li, L., Shi, W., Chen, S., Liu, J., Huang, J., Liu, P.: A dynamic adaptive and resource-allocated selection method based on TOPSIS and VIKOR in federated learning. Neural Processing Letters 56, 11493 (2024).

[12] Diao, X., Rownak, M.R., Olatubosun, S., Vaddi, P.K., Smidts, C.: A multiple-criteria sensor selection framework based on qualitative physical models. Advanced Engineering Informatics 65(B), 103228 (2025).

[13] Jayaraman, P., Nagarajan, K.K., Partheeban, P., Krishnamurthy, V.: Critical review on water quality analysis using IoT and machine learning models. International Journal of Information Management Data Insights 4(1), 100210 (2024).

[14] Essamlali, I., Nhaila, H., El Khaili, M.: Advances in machine learning and IoT for water quality monitoring: A comprehensive review. Heliyon 10(6), e27920 (2024).

[15] Ullah, I., Ali, F., Sharafian, A., Ali, A., Yasir, N.H.M., Bai, X.: Optimizing underwater connectivity through multi-attribute decision-making for underwater IoT deployments using remote sensing technologies. Frontiers in Marine Science 11, 1468481 (2024).

[16] Ilieva, G., Yankova, T.: IoT system selection as a fuzzy multi-criteria problem. Sensors 22(11), 4110 (2022).

[17] Pocco, V., Chucuya, S., Huayna, G., Ingol-Blanco, E., Pino-Vargas, E.: A multi-criteria decision-making technique using remote sensors to evaluate the potential of groundwater in the arid zone basin of the Atacama Desert. Water 15, 1344 (2023).

[18] Mathebula, J., Mbuli, N.: Application of TOPSIS for Multi-Criteria Decision Analysis (MCDA) in Power Systems: A Systematic Literature Review. *Energies* 18(13), 3478 (2025)

[19] Tran, N.-T.: Application of the multi-criteria analysis method MAIRCA, SPOTIS, COMET for the optimisation of sustainable electricity technology development. EUREKA: Physics and Engineering 1, 180–188 (2024).

[20] Hwang, C.L., Yoon, K.: Multiple Attribute Decision Making: Methods and Applications. Springer, Berlin, Heidelberg (1981).

[21] Opricovic, S., Tzeng, G.H.: Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. European Journal of Operational Research 156(2), 445–455 (2004)

A Feasibility Study on Synthetic RGB-NIR Image Generation for Oil Palm Fresh Fruit Bunch Grading

Nor Surayahani Suriani[1], Norzali Hj Mohd[2], Shaharil Mohd Shah[3], Siti Zarina Muji[4], Fadilla Atyka Nor Rashid[5]
Department of Electronics Engineering-Faculty of Electrical and Electronics Engineering,
Universiti Tun Hussein Onn Malaysia, Johor, Malaysia[1, 2, 3, 4]
Center for Artificial Intelligence Technology, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, Selangor, Malaysia[5]

***Abstract*—Accurate ripeness of grading oil palm fruit bunches (FFBs) is essential for optimizing oil quality and harvesting decisions. While near-infrared (NIR) imaging provides useful spectral cues for ripeness assessment, its adoption in field conditions is limited by sensor cost and system complexity. This study presents a low-cost alternative by generating synthetic NIR images from RGB inputs using a U-Net-based image translation model and integrating the generated NIR with RGB channels for ripeness classification. Five deep learning models, including a custom CNN, ResNet-50, EfficientNet-B0, DenseNet-201 and MobileNetV3, were evaluated under RGB-only and RGB + synthetic NIR configurations using identical training protocols. Experimental results demonstrate consistent performance improvements when synthetic NIR was incorporated. EfficientNet-B0 achieved the highest overall accuracy of 90.3%, while MobileNetV3 obtained the highest macro-averaged F1-score of 85.4%, indicating strong and balanced classification across ripeness classes. Confusion matrix analysis further revealed complementary strengths between the models, where EfficientNet-B0 showed stronger robustness in late-stage maturity detection, and MobileNetV3 provided improved discrimination of early-stage ripeness. The results demonstrate that synthetic NIR augmentation enhances classification performance and training stability without requiring specialized imaging hardware.**

***Keywords*—Generative AI; deep learning; U-Net image translation; EfficientNet-B0; MobileNetV3**

I. Introduction

The oil palm (Elaeis gueneensis) is one of the most economically important oil-yielding crops globally, whose crude palm oil quality largely depends on the ripeness degree of fresh fruit bunches (FFBs) at harvest. Optimizing the stage of maturity to harvest FFBs from oil palm maximizes oil production, benefits free fatty acid concentration and minimizes post-harvest losses. Nevertheless, commercial plantations still mostly rely on visual inspection by human graders to determine ripeness, which is subjective, labour-intensive and prone to inconsistency due to variations in experience, lighting conditions and field environments.

Many years ago, computer vision and machine learning techniques were increasingly explored to automate oil palm FFB grading [1]. Conventional approaches rely heavily on RGB images captured using standard cameras, extracting visual cues such as color distribution, texture and fruitlet appearance [2]. While RGB-based methods are cost-effective and easy to deploy, their performance is often affected by illumination changes, occlusion and limited spectral information, particularly when distinguishing between adjacent ripeness stages [3]. To address these limitations, near-infrared (NIR) imaging has been introduced as NIR reflectance is sensitive to internal biochemical properties, related to moisture content, oil accumulation and fruit maturity [4].

Despite their advantages, RGB-NIR imaging systems face practical challenges in large-scale plantation deployment. The acquisition of multispectral data typically requires specialized sensors, higher costs and controlled data collection protocols [5]. Moreover, the availability of large, well-annotated RGB-NIR datasets for oil palm FFB grading remains limited, which restricts the effective training of deep learning models and may lead to overfitting. These constraints motivate the exploration of alternative strategies that can enhance model performance without substantially increasing data acquisition complexity.

Synthetic image generation has emerged as a promising solution to address data scarcity in computer vision applications. By generating artificial yet realistic images, synthetic data can augment a limited real dataset to improve feature learning and enhance generalization performance of classification models. Recent advances in deep generative models have enabled the synthesis of images across different spectral domains, including the generation of NIR representations from RGB inputs and vice versa [6]. While synthetic data generation has been successfully applied in domains such as medical imaging, remote sensing and industrial inspection. The feasibility and effectiveness of oil palm FFB grading using combined RGB-NIR information remain underexplored.

Therefore, this study investigates the feasibility of synthetic RGB-NIR image generation for oil palm fresh fruit bunch grading. Specifically, this work evaluates whether synthetically generated RGB-NIR images can enhance classification accuracy when used to augment real image datasets. The study focuses on assessing the impact of synthetic data on model performance rather than proposing a new grading standard with emphasis on practical applicability under constrained data conditions. The main contributions of this study are: (1) an investigation of synthetic RGB-NIR image generation as a data augmentation strategy for oil palm FFB grading, (2) a comparative analysis of classification performance using real images versus combined real and synthetic datasets, and (3) an empirical assessment of the feasibility of deploying synthetic RGB-NIR data to improve grading accuracy without additional sensor requirements.

The findings of this work provide insights into the potential role of synthetic multispectral data in advancing automated oil palm ripeness grading systems and support future research toward cost-effective and scalable solutions for plantation level deployment.

The remainder of this study is organized as follows. Section II reviews related works on oil palm ripeness detection, RGB-based computer vision methods and multispectral imaging approaches. Section III describes the proposed methodology, including synthetic NIR image generation from RGB inouts uisng U-Net model and ripeness classification using deep learning architectures, namely a custom CNN, ResNet50, EfficientNet-B0, DenseNet-201 and MobileNetV3. Section IV presents the experimental results and discussion, including qualitative evaluation of synthetic NIR images and comparative analysis of classification performance between RGB-only and RGB combined with synthetic NIR inputs. Finally, Section V concludes the study and outlines directions for future work.

II. Related Works

Automated classification of oil palm Fresh Fruit Bunches (FFBs) ripeness has attracted significant research interest due to its direct impact on palm oil yield and quality. Traditionally, FFB grading has been performed manually by trained field graders. But this approach is subjective, inconsistent and labour-intensive. As a result, computer vision and machine learning methods have been explored to standardize and automate the grading process [7].

Early works in oil palm FFB classification predominantly employed RGB image analysis and handcrafted features, such as colour and texture [8]. For instance, studies have employed variations of colour space transformation and texture measures to extract ripeness cues from surface appearance with classifiers like ANN, SVM, K-Nearest Neighbors and advanced deep learning model achieves from moderate to high classification accuracy under controlled environments [9] . These methods confirm that colour indices and textural differences correlate with FFB maturity, yet performance often degrades under real plantation environments with variable illumination, viewpoints and occlusion. In addition, such handcrafted features require careful tuning and may not generalize well across datasets.

To overcome these constraints, deep learning techniques have been adopted for ripeness detection based solely on RGB imagery. Recent work using convolutional neural networks (CNNs) and real-time object detection frameworks such as YOLO has demonstrated improved robustness by automatically learning hierarchical features from large image datasets [10]. CNN-based classifiers and YOLO-based detection pipelines have achieved competitive ripeness classification performance beyond traditional RGB-feature-based methods [11]. However, despite advantages in representation learning, RGB alone provides only limited spectral information, which can be insufficient for distinguishing close ripeness stages, especially when colour differences are subtle.

In parallel with RGB-based techniques, multispectral and near-infrared(NIR) imaging has been explored to enhance ripeness assessment [12]. NIR reflectance is sensitive to internal biochemical properties such as moisture content, cell structure and oil accumulation to offer complementary information to visible bands. A recent review found that combining visible and NIR spectra significantly improves fruit ripeness classification accuracy compared to visual inspection alone. The optical spectrometers that incorporate visible and NIR bands have demonstrated improved ripeness discrimination among ripe and overripe categories for FFB samples [13]. These findings indicate that multispectral techniques can potentially overcome limitations of RGB imaging, but they typically reuire specialized sensors that are costly and complex to deploy at scale in plantation contexts.

The emergence of hyperspectral imaging (HSI) further extends spectral coverage by capturing continuous reflectance across numerous narrow bands. HSI can reveal subtle biochemical and structural changes during fruit maturation that are not discernible with RGB or multispectral imaging alone [14]. Research on wavelength selection from hyperspectral datasets has shown that optimized spectral bands can effectively differentiate between ripeness levels in oil palm FFBs [15]. Hyperspectral devices have been employed in other fruit quality assessments, such as fruit quality evaluation [16], where reconstruction from RGB images to hyperspectral representations has shown promise in augmenting spectral information using deep learning models.

Despite the potential benefits of multipectral and hyperspectral information, practical data acquisition remains a key barrier. Hyperspectral cameras are expensive and bulky, limiting their use in operational plantation environments. Furthermore, large and well annotated RGB-NIR or RGB-HSI paired datasets for oil palm grading are scarce, posing challenges for training deep models and evaluating their generalization performance under diverse field conditions.

To address data scarcity, synthetic image generation and domain translation models have emerged as promising strategies. Deep generative models such as Generative Adversarial Networks (GANs) [17], [18] and image-to-image translation frameworks (e.g. Pix2Pix) have been used in related agricultural domains [19] to synthesize NIR or hyperspectral representations from standard RGB inputs [20]. Meanwhile, synthetic NIR generation from RGB imagery has been applied successfully in crop monitoring tasks, showing that synthetic NIR can approximate spectral information without requiring expensive hardware. Additionally, generative models have been applied to synthetic hyperspectral data generation to enhance maturity classification in other crops.

Despite progress in agricultural synthetic image generation, there are clear research gaps relevant to oil palm FFB grading:

1) Lack of paired RGB-NIR datasets specific to oil palm plantations, which limits effective learning for cross-domain translation and multispectral fusion techniques. Existing datasets primarily provide RGB imagery with limited spectral diversity, and the acquisition of matched NIR data remains sparse.

2) Limited exploration of synthetic spectral generation for oil palm ripeness tasks. Since studies have explored synthetic NIR in other crops and synthetic hyperspectral data for other agricultural quality assessments, there has been no

comprehensive investigation into the feasibility of synthetic RGB-NIR images for enhancing oil palm FFB classification accuracy.

3) Insufficient evaluation under real-world plantation conditions. Most existing work is conducted under controlled lighting or constrained datasets, which do not fully capture the variability and complexity of outdoor agricultural environments.

4) Underutilized hyperspectral to RGB/NIR reconstruction frameworks in the context of oil palm grading. Although hyperspectral reconstruction from RGB has been shown to enhance spectral richness in other produce quality tasks, its applicability and effectiveness for oil palm FFb remain under-examined.

Addressing these gaps, the present study examines the feasibility of synthetic RGB-NIR image generation as a data augmentation strategy for oil palm FFB grading and evaluates whether such synthetic modalities can enhance classification performance without requiring additional hardware. This represents a step toward developing cost-effective, scalable and generalizable automated grading systems suitable for real plantation deployment.

III. Methodology

This study adopts a feasibility-driven methodological framework to investigate whether synthetic near-infrared (NIR) information generated from RGB images can enhance oil palm fresh fruit bunch (FFB) grading. The proposed approach consists of two sequential stages: (1) synthetic NIR image generation from RGB data using a U-Net model, and (2) deep learning based classification of FFB ripeness using RGB images with and without synthetic NIR augmentation. The overall objective is not to replicate true NIR measurements, but to assess whether NIR-like representations inferred from RGB data can provide complementary information that improves classification performance under practical data acquisition constraints.

A. RGB Image Acquisition and Annotation

All image data used in this study consists exclusively of RGB images captured in oil palm plantations under natural outdoor conditions. Image acquisition was performed using a standard digital camera without any near-infrared or multispectral sensors. Images were collected across multiple harvesting sessions to capture variability in illumination, background clutter, fruit bunch orientation, and occlusion. Each image corresponds to a single oil palm FFB and was manually annotated into predefined ripeness categories, such as underripe, unripe, ripe, and overripe. The grading groundtruth were determined by experienced plantation personnel following standard harvesting guidelines. These annotated RGB images constitute the sole real data source for both synthetic image generation and classification experiments. The image dataset was obtained from the Pertubuhan Peladang Parit Raja. The total data set of 3476 images for four categories of oil palm fruits is shown in Fig. 1.



Fig. 1. Oil palm fruit images underripe, unripe, (top) ripe, and overripe (bottom) from left to right.

B. Synthetic NIR Image Generation

The second phase of the simulation work was the generation of synthetic NIR images. Before model development, all RGB images underwent preprocessing to ensure consistency and suitability for deep learning models.

1) Image preprocessing: The images were resized to a fixed spatial resolution to accommodate network input requirements and to standardize feature representation. Pixel intensity values were normalized to improve numerical stability during training. The dataset was then partitioned into training, validation, and testing subsets using stratified sampling to preserve the original class distribution across splits. No NIR images were used or referenced during preprocessing, as the study relies entirely on RGB inputs.

2) U-Net architecture: Synthetic NIR image generation was performed using a U-Net convolutional neural network, which is well-suited for dense pixel-level prediction tasks due to its encoder-decoder architecture and skip connections. The U-Net model was designed to take a three-channel RGB image as input and produce a single-channel synthetic NIR image as output. Fig. 2 shows the U-Net architecture.

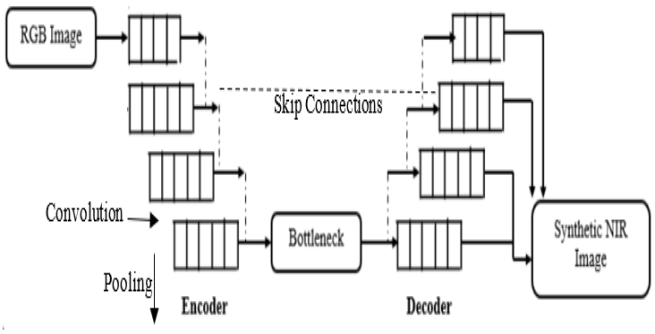


Fig. 2. U-Net model architecture for synthetic NIR image generation, illustrating the encoder-decoder structure and skip connections.

The encoder extracts hierarchical features related to surface texture, color gradients, and spatial structure. Meanwhile, the decoder reconstructs a spatially aligned representation that approximates NIR-like characteristics. Skip connections between corresponding encoder and decoder layers enable the preservation of fine-grained spatial details. The important parts were capturing fruitlet boundaries and structural patterns relevant to ripeness assessment.

Since no real NIR images were collected, the U-Net model was trained using a self-supervised and prior-guided learning strategy rather than direct pixel-wise supervision. The training objective was to encourage the network to generate physically plausible and structurally meaningful NIR-like representations based on known correlations between visible and NIR responses reported in hyperspectral and multispectral studies of agricultural produce.

The learning process was guided by a combination of reconstruction consistency constraints, smoothness regulation, and edge-preserving objectives to maintain spatial coherence and prevent unrealistic artifacts [21]. Spectral priors derived from the hyperspectral to RGB/NIR relationship were implicitly incorporated to guide the network toward a representation that emphasizes reflectance characteristics associated with fruit maturity, such as texture homogeneity and intensity variation. This training strategy aligns with the feasibility-oriented nature of the study, as the aim is to infer complementary information rather than reproduce true NIR measurements.

3) Synthetic NIR augmentation strategy: Once trained, the U-Net model was applied to the RGB images in the training set to generate corresponding synthetic NIR images. These synthetic images were then used to construct different data configurations for classification experiments. In the baseline configuration, only RGB images were used for ripeness classification.

In the augmented configuration, the synthetic NIR images were concatenated with the RGB channels to form four-channel input tensors. Additional experiments examined the use of synthetic NIR images alone and the use of synthetic NIR to augment minority classes, enabling a systematic evaluation of the contribution of synthetic spectral information under different scenarios.

C. Deep Convolutional Neural Network Model

Deep convolutional neural networks (CNNs) were employed for FFB grading due to their demonstrated effectiveness in agricultural image classification and visual feature learning. In this study, five representative CNN architectures were evaluated to comprehensively assess the impact of synthetic NIR augmentation across different network depths, parameter scales and computational characteristics, namely a custom baseline CNN, ResNet50, EfficientNet-B0, DenseNet-201 and MobileNetV3.

A baseline custom CNN architecture was first implemented to establish reference performance using both RGB-only and RGB augmented with synthetic NIR inputs. The network consists of sequential convolutional layers followed by batch normalization, nonlinear activation and max-pooling operations to progressively extract hierarchical spatial features associated with oil palm fruit texture, shape and surface patterns. This model provides a lightweight benchmark to evaluate the effectiveness of spectral augmentation under limited architectural complexity.

ResNet50 was adopted as a deep residual network benchmark for comparative analysis. The architecture employs identity shortcut connections that mitigate the vanishing gradient problem and enable stable optimization of deep networks. Owing to its strong representation capacity and extensive adoption in agricultural vision applications, ResNet50 serves as a reliable reference for evaluating the contribution of synthetic NIR information in a high-capacity model. The network was initialized using ImageNet-pretrained weights, and the final fully connected layer was replaced to match the number of FFB ripeness classes. For RGB + synthetic NIR experiments, the first convolutional layer was modified to accept four-channel input while preserving pretrained knowledge through channel-wise weight initialization.

To explore parameter-efficient and scalable architectures, EfficientNet-B0 was incorporated. EfficientNet-B0 employs compound scaling to jointly balance network depth, width and input resolution. This enables strong accuracy with significantly reduced parameter count compared to conventional CNNs. This characteristic makes EfficientNet-B0 particularly suitable for feasibility studies where both performance and computational efficiency are critical considerations.

DenseNet-201 was included to evaluate the effect of dense feature reuse and enhanced gradient propagation. Dense connections encourage feature sharing across layers and reduce redundant representations, which can be advantageous when training with limited datasets. Its inclusion allows assessment of whether dense connectivity benefits the exploitation of synthetic spectral features.

MobileNetV3 was selected as a lightweight architecture optimized for deployment in resource-constrained environments. It utilised depth-wise separable convolutions, inverted residual blocks and hardware optimization strategies to achieve favourable accuracy to complexity trade-offs. This model reflects potential real-world implementation scenarios in plantation monitoring systems where computational resources and power consumption are limited.

All networks were trained using identical dataset partitions, preprocessing pipelines and training hyperparameters to ensure fair and unbiased comparison. This multi-model evaluation framework enables systematic investigation of the consistency, robustness and generalizability of synthetic NIR augmentation across diverse CNN architectures, ranging from shallow baseline models to deep residual, densely connected and efficient oriented networks.

D. Performance Metrics

The performance of the proposed oil FFB grading framework was evaluated using standard classification metrics derived from the confusion matrix, including accuracy, precision, recall and F1-score. These metrics provide complementary perspectives on classification behavior and are

widely adopted in agricultural image analysis to assess both overall performance and class-specific reliability.

Overall classification accuracy measures the proportion of correctly classified samples relative to the total number of samples and is defined as:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

where TP, TN, FP and FN denote the numbers of true positives, true negatives, false positives and false negatives, respectively. While accuracy provides a general indication of model performance, it may be biased in the presence of class imbalance, which is common in real-world agricultural datasets.

Precision quantifies the proportion of correctly predicted positive samples among all predicted positives and is expressed as:

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall, also referred to as sensitivity, measures the proportion of correctly predicted positive samples among all actual positives and is given by:

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

The F1-score is defined as the harmonic mean of precision and recall, which provides a balanced assessment of classification performance by jointly considering false positives and false negatives.

$$F1-score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

In this study, the macro-averaged F1-score was emphasized to ensure equal contribution from each ripeness class, regardless of class frequency.

$$F1_{macro} = \frac{1}{C}\sum_{i=1}^{C} F1_i \quad (5)$$

where C denotes the total number of classes and $F1_i$ represents the F1-score of the i-th class, given by

$$F1_i = 2 \times \frac{Precision_{\ i} \times Recall_i}{Precision_i + Recall_i} \quad (6)$$

where $Precision_{\ i}$ and $Recall_i$ correspond to the precision and recall computed for class i, respectively.

Among the evaluated metrics, the macro-averaged F1-score is considered the most important performance indicator for this feasibility study. Unlike accuracy, macro F1-score is less influenced by dominant classes and more sensitive to improvements in underrepresented or visually ambiguous ripeness categories. This characteristic is particularly relevant for oil palm FFB grading, where misclassification between adjacent ripeness stages can have significant economic implications. Consequently, the macro-averaged F1-score was used as the primary metric for comparing RGB-only and RGB with synthetic NIR configurations, while accuracy and class-wise precision and recall were reported as supporting metrics.

IV. Results and Discussion

The performance of the proposed framework was evaluated by comparing classification results obtained using RGB-only inputs against those obtained using RGB augmented with synthetic NIR images. Across all evaluated models, the inclusion of synthetic NIR consistently improved classification accuracy relative to the RGB-only baseline. This improvement indicates that the synthetic NIR channel provides complementary information that enhances feature discrimination for oil palm fresh fruit bunch (FFB) grading.

Performance gains resulting from synthetic NIR augmentation were observed across all model types, suggesting that the effectiveness of synthetic NIR is not dependent on a specific network architecture. This consistency supports the feasibility of using synthetic spectral representation to enhance classification robustness.

A. Synthetic NIR Generation

Fig. 3 illustrates representative examples of synthetic near-infrared (NIR) images generated by the U-Net model for oil palm FFB at four ripeness stages: underripe, unripe, ripe and overripe. Although no real NIR images were acquired during data collection, the trained U-Net successfully learned a mapping from RGB inputs to synthetic NIR-like intensity representations which produce consistent spatial and textural patterns across different maturity levels.

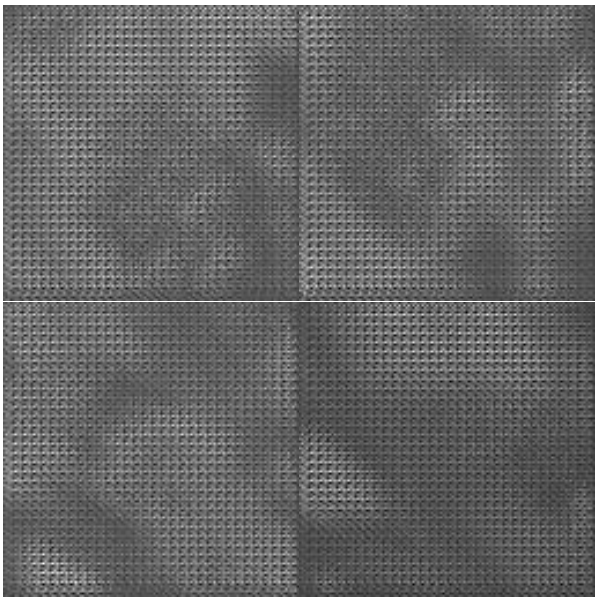


Fig. 3. Synthetic NIR of oil palm fruit images underripe, unripe (top), ripe, and overripe (bottom) from left to right.

Visually, the synthetic NIR images exhibit relatively smooth grayscale distributions with localized intensity variations that correspond to fruit surface structure and cluster density. Brighter regions indicate higher predicted reflectance, while darker areas represent lower reflectance. These intensity variations are not random noise but follow coherent spatial patterns aligned with the shape and arrangement of fruitlets within each bunch. This suggests that the network has learned meaningful morphological cues from RGB inputs.

Across ripeness stages, subtle differences in brightness distribution and texture consistency can be observed. The underripe and unripe samples display more uniform and compact intensity patterns, whereas the ripe and overripe

samples exhibit increased contrast and localized brightness variation. This behavior is consistent with expected physical changes during fruit maturation, such as surface texture evolution, moisture redistribution and oil accumulation, which typically influence near-infrared reflectance characteristics. Although the generated images do not represent true spectral measurements, they encode discriminative structural information that complements visible color features.

The synthetic NIR images preserve spatial continuity and edge consistency, indicating that the U-Net effectively maintains geometric fidelity during image translation. The absence of severe artifacts or discontinuities demonstrates stable learning and successful generalization across different fruit appearances. This stability is critical because downstream classification models rely on consistent spatial patterns rather than absolute pixel values. The synthetic NIR channel likely enhances feature separability by introducing additional intensity-based cues that are less sensitive to illumination variations and color ambiguity in RGB images alone. This is particularly beneficial for discriminating adjacent ripeness stages, which often exhibit overlapping color characteristics in visible imagery.

Overall, Fig. 3 confirms that the U-Net is capable of generating structurally meaningful synthetic NIR representations from RGB inputs, providing complementary information that strengthens the robustness and discriminative power of the FFB grading framework without requiring physical NIR sensors.

B. Training Convergence and Optimization Stability

Fig. 4 presents the training and validation loss curve of EfficientNet-B0 under RGB only (left) and RGB + synthetic NIR (right) input configurations. The convergence behavior provides important insight into optimization stability, learning efficiency and generalization capability of the proposed framework.

For the RGB-only configuration, the training loss decreases rapidly during the early iterations indicate effective initial feature learning. However, noticeable fluctuations and spikes are observed throughout the training process, suggesting sensitivity to limited data variability and potential instability in optimization. The validation loss follows a similar decreasing trend but remains relatively higher and more oscillatory implies moderate generalizetion uncertainty. The corresponding accuracy curve exhibits substantial variance across iterations and reflects the difficulty of learning discriminative features using RGB information alone, particularly for visually overlapping ripeness classes.

In contrast, the RGB + synthetic NIR configuration demonstrates smoother and more consistent convergence behavior. The training loss decreases steadily with fewer abrupt fluctuations and reaches a lower final value compared to the RGB-only case. In addition, the validation loss closely tracks the training loss with a reduced gap between the two curves, indicating improved generalization and reduced risk of overfitting. The validation accuracy stabilizes at higher levels with less variance, reflecting more reliable feature learning when synthetic spectral information is incorporated.

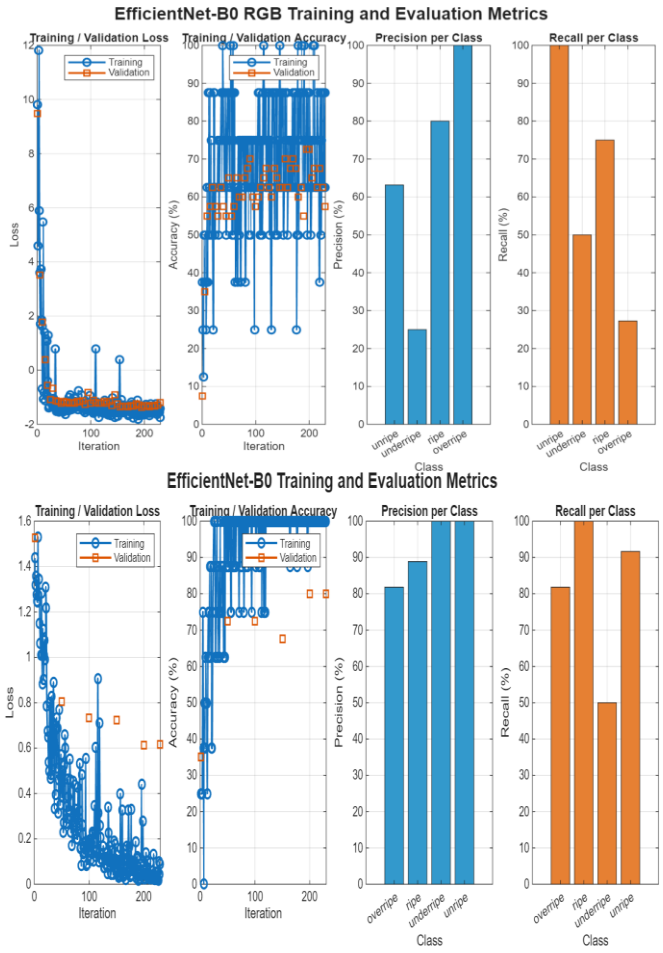


Fig. 4. Training and validation loss curves of EfficientNet-B0 for RGB-only (Top) and RGB + synthetic NIR images (Bottom).

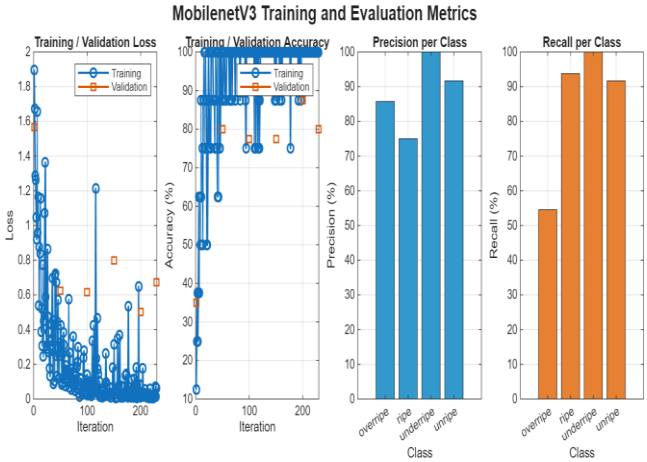


Fig. 5. Training and validation loss curves of MobileNetV3 for RGB + synthetic NIR images.

In addition to EfficientNet-B0, the convergence behavior of MobileNetV3 under the RGB + synthetic NIR configuration is illustrated in Fig. 5. The training loss exhibits a rapid decay during the early iterations. After 50 iterations, the loss stabilizes at a low magnitude with minor fluctuations, suggesting convergence toward a stable minimum and absence of optimization divergence. The gap between training and

validation curves remains small, indicating good generalization capability and limited overfitting. This behavior confirms that the lightweight architecture is capable of learning robust representations even with limited training data when augmented with synthetic spectral information.

The per-class precision and recall bars indicate strong discrimination for the underripe and unripe classes, with high recall values approaching 100%, which supports the improved macro-averaged F1-score reported in Table I. The synthetic NIR images likely enhance contrast and structural cues that are less sensitive to illumination variations present in RGB images, enabling the network to converge more efficiently and consistently.

Overall, the convergence characteristics of MobileNetV3 confirm that the optimization process is stable, efficient and generalizable when synthetic NIR is incorporated. The consistent learning behavior observed across both EfficientNet-B0 and MobileNetV3 demonstrates that the benefits of synthetic NIR augmentation are not architecture dependent and supports the robustness and scalability of the proposed framework.

C. Model Performance Comparison

Table II summarizes the classification performance of different deep learning models under two input configurations: RGB-only and RGB augmented with synthetic NIR. Overall, the results demonstrate that the inclusion of synthetic NIR improves both accuracy and macro-averaged F1-score for most evaluated architectures. This confirms the feasibility of using synthetic spectral information to enhance oil palm FFB grading.

TABLE I. THE MODEL PERFORMANCE COMPARISON

Input	Model	Accuracy (%)	Macro F1-score (%)
RGB only	CNN	75.8	70.1
	ResNet50	77.5	75.3
	EfficientNet-B0	85.3	80.5
	DenseNet-201	82.1	80.1
	MobileNetV3	80.3	82.5
RGB + Synthetic NIR	CNN	78.5	74.0
	ResNet50	78.1	75.7
	EfficientNet-B0	**90.3**	84.6
	DenseNet-201	85.4	80.9
	MobileNetV3	82.9	**85.4**

Among the RGB-only models, EfficientNet-B0 achieved the highest accuracy (85.3%) and macro F1-score (80.5%), indicating strong baseline discriminative capability. DenseNet-201 and MobileNetV3 also performed competitively, suggesting that both deep feature reuse (DenseNet) and lightweight architectures (MobileNetV3) are suitable for this task. The custom CNN and ResNet50 achieved lower macro F1-scores, reflecting the limitations of shallow architectures and the sensitivity of deeper residual networks to limited training data.

When synthetic NIR was incorporated, the performance improvements were observed across all models. The most notable improvement was achieved by EfficientNet-B0, whose accuracy increased from 85.3% to 90.3%, and the macro F1-score improved substantially from 80.5% to 84.6%. This demonstrates that synthetic NIR provides complementary information that can be effectively exploited by compound-scaled architectures. Similarly, MobileNetV3 achieved the highest macro F1-score (85.4%) under RGB+synthetic NIR, indicating that lightweight networks are capable of benefiting from synthetic spectral augmentation while maintaining computational efficiency.

Although ResNet50 and CNN exhibited more modest improvements, their consistent upward trends suggest that the benefit of synthetic NIR is not architecture-specific. This consistency strengthens the argument that synthetic NIR contributes meaningful additional representation rather than acting as random noise. In addition, the macro-averaged F1-score was emphasized as the primary metric because it balances class-wise performance and mitigates class imbalance effects. The achieved macro F1-scores of 84.6% (EfficientNet-B0) and 85.4% (MobileNetV3) indicate strong and balanced classification performance across ripeness categories, validating the feasibility of the proposed approach.

1) Class-level performance analysis: Table II and III present per-class precision, recall, and F1-score for EfficientNet-B0 and MobileNetV3 under the RGB + synthetic NIR configuration. These results provide deeper insight into class-wise behavior and error distribution.

TABLE II. PER-CLASS METRICS EFFICIENTNET-B0 FOR RGB+SYNTHETIC NIR IMAGE

Class	Precision	Recall	F1-score
Unripe	100	91.67	95.65
Underripe	100	50	66.67
Ripe	88.89	100	94.12
Overripe	81.82	81.82	81.82

TABLE III. PER-CLASS METRICS MOBILENETV3 FOR RGB+SYNTHETIC NIR IMAGE

Class	Precision	Recall	F1-score
Unripe	91.67	91.67	91.67
Underripe	100	100	100
Ripe	75	93.75	83.33
Overripe	85.71	54.55	66.67

For EfficientNet-B0, the unripe class achieved perfect precision (100%) and high recall (91.67%), resulting in a strong F1-score of 95.65%. This indicates that the model reliably identifies visually distinct unripe bunches. However, the underripe class exhibits a lower recall of 50%, despite perfect precision. This suggests that while predictions labeled as underripe are highly accurate, a significant portion of true underripe samples are misclassified into neighboring classes. This behavior reflects the intrinsic visual ambiguity between the underripe and ripe stages.

The ripe class shows excellent recall (100%) and a strong F1-score (94.12%), indicating that the synthetic NIR channel enhances the model's ability to capture maturity-related features. The overripe class demonstrates balanced precision and recall (both approximately 81.8%), reflecting moderate

confusion likely caused by texture degradation and environmental variability.

For MobileNetV3, the underripe class achieves perfect precision and recall (100%), indicating excellent class separability under this architecture. The unripe class also maintains balanced precision and recall (91.67%), that confirm stable discrimination. However, the overripe class shows reduced recall (54.55%), which suggests that lightweight models may struggle to capture subtle degradation patterns associated with over-ripeness, even with synthetic NIR augmentation. The ripe class maintains reasonable performance with an F1-score of 83.33%.

These class-wise results indicate that synthetic NIR improves discrimination, particularly for intermediate maturity stages, but challenges remain for visually overlapping classes such as underripe and overripe.

2) Comparison between EfficientNet-B0 and MobileNetV3: MobileNetV3 achieved the highest macro F1-score (85.4%), EfficientNet-B0 delivered the highest overall accuracy (90.3%). This reveals an important trade-off between balanced class performance and overall correctness. EfficientNet-B0 demonstrates stronger overall predictive power, whereas MobileNetV3 exhibits slightly better balance across classes, particularly for underripe detection.

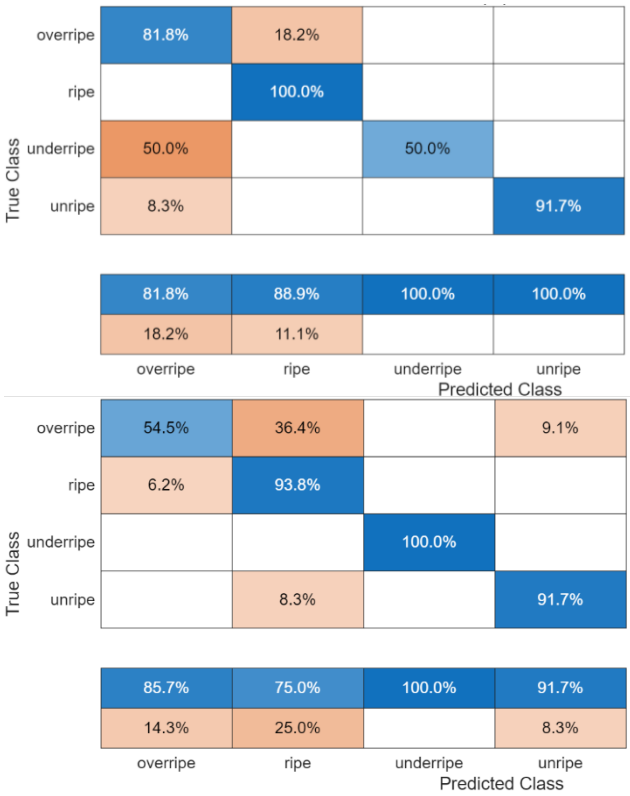


Fig. 6. Confusion matrix for efficientNet-B0 (top) and mobileNet-V3 (bottom).

Fig. 6 presents the normalized confusion matrices for EfficientNet-B0 and MobileNetV3 under the RGB + synthetic NIR configuration, providing insight into class-wise prediction behavior and misclassification patterns beyond overall performance metrics. For EfficientNet-B0, strong discrimination performance is observed for the ripe and unripe classes, where correct classification rates reach 100% and 91.7%, respectively. This indicates that the model effectively captures maturity-related texture and structural cues enhanced by synthetic NIR information. The overripe class also achieves a high recognition rate of 81.8%, with the remaining 18.2% primarily misclassified as ripe.

This confusion is expected due to gradual visual transitions between late ripeness stages, where texture degradation and color saturation differences may be subtle. The underripe class shows comparatively weaker recall, with only 50% correctly classified. While the remaining samples are misclassified as ripe. This reflects inherent ambiguity between early maturity stages and highlights the challenge of separating underripe and ripe fruit bunches based solely on visual cues, even with synthetic spectral augmentation.

In contrast, MobileNetV3 demonstrates excellent recognition for the underripe and unripe classes, achieving 100% and 91.7% correct classification rates, respectively. This suggests that the lightweight architecture effectively leverages synthetic NIR features to enhance early-stage ripeness discrimination. The ripe class also maintains strong performance at 93.8% with only minor confusion toward the overripe category. However, MobileNetV3 exhibits noticeably reduced performance for the overripe class, where only 54.5% of samples are correctly classified. A substantial proportion (36.4%) is misclassified as ripe, and a smaller fraction (9.1%) is confused with unripe. This indicates that MobileNetV3 may lack sufficient representational capacity to fully capture subtle degradation patterns associated with over-ripeness, despite benefiting from synthetic spectral augmentation.

Overall, the confusion matrix analysis reveals complementary strengths between the two architectures. EfficientNet-B0 provides stronger robustness in detecting later ripeness stages, particularly when overripe samples contribute to higher overall accuracy. Conversely, MobileNetV3 excels in discriminating early maturity stages and results in a slightly higher macro-averaged F1-score due to balanced class performance. These observations align with the quantitative results reported in Table I and the per-class metrics in Tables II and III.

From a deployment perspective, MobileNetV3 offers advantages in computational efficiency and suitability for embedded systems, making it attractive for real-time plantation applications. Meanwhile. EfficientNet-B0 may be preferable when maximizing classification accuracy is the primary objective and computational resources are less constrained. The close macro F1-scores between these two models further reinforce the robustness of the synthetic NIR augmentation strategy across different architectural paradigms.

3) Impact of synthetic NIR on feature discrimination: The observed improvements suggest that synthetic NIR provides complementary cues beyond visible color information. Although the synthetic NIR images are not true spectral measurements, they appear to encode structural and intensity variations that enhance feature separability in the learned

representations. This is particularly beneficial for distinguishing visually similar ripeness stages, where RGB features alone may be insufficient.

The performance gains achieved without any real NIR acquisition demonstrate the practical value of the proposed approach. By leveraging RGB-only data and learned spectral priors, the system avoids additional hardware cost and deployment complexity. Hence, aligned well with real-world plantation constraints.

4) Limitations and reliability considerations: Despite the encouraging results, certain limitations remain. First, the absence of real NIR ground truth prevents direct validation of the physical accuracy of the synthetic NIR images. Therefore, conclusions are based on downstream classification performance rather than spectral fidelity. Second, class imbalance and limited sample size may influence class-wise variability, particularly for underripe and overripe categories. Nevertheless, the consistent macro F1-score improvements across multiple architectures indicate that the observed gains are systematic rather than incidental. The strong performance of lightweight architectures further supports the feasibility of deploying the proposed framework in resource-constrained environments.

V. Conclusion

This study investigated the feasibility of enhancing oil palm FFB grading performance through synthetic near-infrared (NIR) image generation from RGB inputs using a U-Net architecture. Unlike conventional approaches that require dedicated multispectral or hyperspectral sensors, the proposed framework generates synthetic spectral representations from standard RGB images. This enable cost effective spectral augmentation without additional hardware requirements.

Quantitative evaluation across five deep learning architectures custrom CNN, ResNet50, EfficientNet-B0, DenseNet-201 and MobileNetV3, showed consistent performance gains when synthetic NIR was included. EfficientNet-B0 achieved the highest overall accuracy of 90.3%, while MobileNetV3 obtained the highest macro-averaged F1-score of 85.4%, indicating strong and balanced classification performance across ripeness classes. Confusion matrix analysis revealed complementary strengths between the two models, with EfficientNet-B0 exhibiting stronger robustness in late-stage maturity detection and MobileNetV3 excelling in early-stage discrimination.

The results confirm that synthetic NIR provides complementary discriminative information beyond visible RGB features, particularly for visually ambiguous maturity stages. The improvement is also consistent across diverse architectures, reinforcing the generalizability and robustness of the proposed approach. The strong performance achieved using lightweight architectures further supports the practical feasibility of real-time deployment in plantation environments. Future work will focus on validating synthetic NIR against spectral information and exploring advanced generative models to further enhance the potential of NIR for oil palm FBB grading quality and oil estimation. Overall, the proposed synthetic RGB-to-NIR generation framework demonstrates a viable pathway for improving oil palm FBB accuracy while maintaining low deployment cost and operational simplicity, hence offering strong potential for scalable agricultural monitoring applications.

Acknowledgment

This research was supported by Universiti Tun Hussein Onn Malaysia (UTHM) through TIER1 (vot Q874).

References

[1] R. Kurniawan, Samsuryadi, Mohamad, Fatma Susilawati, Wijaya, Hanna Oktafia Lingga and Santoso, Budi, "Advancing palm oil fruit ripeness classification using transfer learning in deep neural networks," Bulletin of Electrical Engineering and Informatics, vol. 14, no. 2, pp. 1126 - 1137, 2025.

[2] A. Septiarini, A. Sunyoto, H. Hamdani, F. Utaminingrum and H. Rahmania, "Machine vision for the maturity classification of oil palm fresh fruit bunches based on color and texture features," Scientia Horticulturae, vol. 286, 2021.

[3] D. Setiawan, Utomo, Pradita Eko Prasetyo and Alfalah, Muksin, "Detection of Oil Palm Fruit Ripeness through Image Feature Optimization using Convolutional Neural Network Algorithm," International Journal on Informatics Visualization, vol. 9, no. 2, pp. 674 - 682, 2025.

[4] M. Shiddiq, Saktioto, Salambue, Roni, Wardana, Fiqra, Dasta, Vicky Vernando, Harmailil, Ihsan Okta, Rabin, Mohammad Fisal, Arpyanti, Nisa and Wahyudi, Dilham, "Multispectral imaging and deep learning for oil palm fruit bunch ripeness detection," Bulletin of Electrical Engineering and Informatics, pp. 4168 - 4181, 2024.

[5] A. S. Safiee, Najib, Muhammad Sharfi, Mohamad, Mujahid and Che Zan, Nazriyah, "Integrated Approaches for Palm Oil Fresh Fruit Bunch Ripeness Classification Using Near-Infrared Optical Sensors, Artificial Data and Case-Based Reasoning," in 6th Symposium on Intelligent Manufacturing and Mechatronics, 2026.

[6] I. Sa, Lim, Jongyoon, Ahn, Hoseok and Macdonald, Bruce, "deepNIR: Datasets for Generating Synthetic NIR Images and Improved Fruit Detection System Using Deep Learning Techniques," Sensors, vol. 22, no. 13, 2022.

[7] R. R. Hutapea, Hanun, Taqiyya and Irwansyah, Edy, "Comparison of Palm Oil Fruit Ripeness Detection and Classification Methods Using Deep Learning," ICIC Express Letters, vol. 16, no. 7, pp. 797 - 804, 2025.

[8] S. C. Valentine, Alfred, Rayner, Fui, Florence Sia, Shamrie Sainin, Mohd and Iswandono, Zamhar, "Classification of Oil Palm Fresh Fruit Bunches (FFB) Based on Its Maturity Colour Using Convolutional Neural Network (CNN) Approach," in 9th International Conference on Computational Science and Technology, 2022.

[9] P. Laddawong, Pianroj, Yutthapong, Chotikawanid, Piyanart, Punvichai, Teerasak, Jumrat, Saysunee, Kham-Ouam, Atitaya and Muangprathub, Jirapond, "Federated Learning for Grading Oil Palm Fruit Ripeness in the Oil Palm Industry," Applied Computational Intelligence and Soft Computing, vol. 1, 2025.

[10] M. Rosbi and Omar, Zaid, "Segmentation of Outdoor Oil Palm Fruit Images for Ripeness Detection using YOLO Algorithm and Features Thresholding," in Sustainable and Integrated Engineering International Conference, 2022.

[11] M. H. Junos, Mohd Khairuddin, Anis Salwa, Abu Talip, Mohamad Sofian, Kairi, Muhammad Izhar and Siran, Yosri Mohd, "Improved hybrid feature extractor in lightweight convolutional neural network for postharvesting technology: automated oil palm fruit grading," Neural Computing and Applications, vol. 36, no. 32, pp. 20473 - 20491, 2024.

[12] Z. Ünal, "Image preprocessing techniques applied on nir images for fruit bruise detection," in 3rd International Conference on Research of Agricultural and Food Technologies, 2024.

[13] N. I., Seminar K.B, Irzaman and Budiastra I.W., "Improving the accuracy of near-infrared (NIR) spectroscopy method to predict the oil content of

oil palm fresh fruits," in IOP Conference Series: Earth and Environmental Science, 2020.

[14] C. Sun, Zhang, Lei, Zhai, Lixiang, Shen, Tingting, Cai, Jianrong, Zou, Xiaobo and Guo, Zhiming, "Automatic early bruise detection in strawberry fruit by hyperspectral imaging and deep learning techniques," Postharvest Biology and Technology, vol. 232, 2026.

[15] B. O.M., Shariff A.M, Mahmud A.B., Shafri H. and Alfatni M, "Oil palm fruit grading using a hyperspectral device and machine learning algorithm," in 7th IGRSM International Conference and Exhibition on Remote Sensing and GIS, 2014.

[16] X. Yu, Liu, Shuangyin and Wang, Chenjiaozi, "Detection of fungal disease in citrus fruit based on hyperspectral imaging," Information Processing in Agriculture, vol. 12, no. 4, pp. 456 - 465, 2025.

[17] R. M. Desai, Bhole, Varsha Yogesh and Deshpande, Manoj M., "Generative Adversarial Networks and Their Variants for Precise Fruit Recognition," in 4th International Conference on Sustainable Expert Systems, 2024.

[18] S. V and Kasturi K, "Structural Invariant Feature Segmentation Based Apple Fruit Disease Detection Using Deep Spectral Generative Adversarial Networks," SN Computer Science, vol. 5, no. 5, 2024.

[19] D. Doria Usta, Hundelshaussen, Ricardo, Martínez López, Carlos, Salgado Chamorro, Delio, López Martínez, César, Coimbra Leite Costa, João Felipe and Arcari Bassani, Marcel, "From RGB to Synthetic NIR: Image-to-Image Translation for Pineapple Crop Monitoring Using Pix2PixHD," Technologies, vol. 13, no. 12, 2025.

[20] Q. Wang, Lu, Jinzhu, Wang, Yuanhong and Miao, Fajun, "In situ nondestructive identification of citrus fruit ripeness via hyperspectral imaging technology," Plant Methods, vol. 21, no. 1, 2025.

[21] Q. Jiang, Zhou, Tao, Yu, Dongjian, Song, Yupeng, Yao, Shaowen, Wang, Punming and Jin, Xin, "GSU-Fusion: A Text-guided Symmetric U-Net Framework for Infrared and Visible Image Fusion," Information Fusion, vol. 127, 2026.

The Acquiring Optimal Models of Random Forest and Support Vector Machine Through Tuning Hyperparameters in Classifying the Imbalanced Data

Dwija Wisnu Brata[1], Arif Djunaidy[2]*, Daniel Oranova Siahaan[3], Samingun Handoyo[4]
Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia [1, 3]
Department of Information Systems, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia [2]
Department of Information Systems, Brawijaya University, Malang, Indonesia [1]
Data Science Study Program, Brawijaya University, Malang, Indonesia [4]

***Abstract*—Machine learning models most often misclassify the positive class in the dataset with class imbalance. Besides, a sophisticated model involves the hyperparameters that need to be tuned to the optimal values. The study aims to tune hyperparameters of random forest (RF) and support vector machine (SVM) models using 5-fold cross-validation data, to build the best RF and SVM for two data scenarios: the original and oversampling training data, and to compare the models' performances in either the training or testing data. The RF hyperparameters: the instance number in the leaf node and tree depth of the RF, were acquired (500, 10), respectively. Whereas, the SVM hyperparameters: the values of gamma and constant, were acquired (0.001, 500), respectively. The benchmark models achieved around 98% across the accuracy, precision, recall, and F1 score metrics. However, it performed worse on the Mathew's Correlation Coefficient (MCC) and Area Under the Curve (AUC): 0.0000 and 0.5000, respectively. The models trained on the class-imbalance dataset failed to predict the positive class. Although the best RF and SVM models trained on the oversampled dataset perform worse than both benchmark models across four standard metrics, the RF best model shows improvements of approximately 7% (from 0.000 to 0.067) and 11% (from 0.500 to 0.612) while the SVM best model show slightly different improvements of approximately 6% (from 0.000 to 0.056) and 11% (from 0.500 to 0.611) in MCC and AUC, respectively. Both the RF and SVM models improve in predicting the positive class, and the best RF model performs slightly better.**

***Keywords*—Area under the curve; cross-validation folds; Matthew's correlation coefficient; optimal hyperparameters; oversampling technique**

I. Introduction

Choosing the right decision from among the decision candidates will lead to the appropriate policy decision-maker. Decision-makers often face binary choices across fields, from economics to the medical sciences. Financial analysts evaluated the company profile, that is, features related to the practice of good accounting, to recommend the company's status, which is fraudulent or not fraudulent [1]. Nutritionists assessed the baby's status as stunted or normal [2]. Midwives decide whether a pregnant woman should give birth through surgery or give birth normally [3]. Unfortunately, binary-class datasets, particularly medical datasets, often exhibit class imbalance. The severity of imbalance with proportions of 80% vs. 20% or 90% vs. 10% is still classified as a standard imbalance class, as Liu et al. stated. [4], but the proportion of 95% vs. 5% or larger gap is categorized as an extreme imbalance class, as stated by Mienye and Sun[5]. Because class imbalance can lead machine learning models to perform poorly at predicting the positive class, the imbalance should be addressed immediately to produce a balanced class dataset.

Machine learning linear models, including logistic regression, linear discriminant analysis, Bayesian, and naïve Bayes, employ a linear hyperplane as the separating boundary to classify instances from different classes [6]. Unfortunately, the linear separating boundary has many possible solution lines, and the optimal solution line is not unique. A Decision tree categorized as a model called a random forest (RF), is a choice for better performance than a decision tree classification model [8]. Some studies have reported satisfactory performance of the RF model, including Speiser et al. [9], Alam et al. [10], and El-Sappagh et al. [11]. Although RF models are relatively resistant to overfitting, some hyperparameters, such as the tree depth and the minimum number of instances required to split a leaf, strongly influence the optimal RF model. Tuning RF hyperparameters should receive significant attention to obtain the best RF model. Another classification model type, the support vector machine (SVM), is a popular model with superior performance on nonlinear hyperplane boundaries [12]. Nevertheless, determining an optimal SVM model is difficult [13]. The SVM with a radial basis function (RBF) kernel has two hyperparameters: a constant term, which is the inverse of the regularization term in the SVM objective function, and a gamma parameter, which controls the width of the RBF kernel [14]. As with RF modeling, tuning the SVM hyperparameters should receive significant attention. The works conducted by Kumar et al. [15], Bektaş [16], and Ganaie et al. [17] are some examples of applications of the SVM classification model. Again, both hyperparameters of the SVM with the RBF kernel must be tuned systematically to obtain the optimal SVM model. The performance evaluation metric is another critical factor in determining the best model [18]. A confusion matrix was used to compute performance metrics [19]. Using different metrics for model performance evaluation can lead to different best models, particularly in imbalanced-class classification.

*Corresponding author.

There are three central issues encountered in developing the best RF and SVM models: oversampling to address class imbalance, tuning the model's hyperparameters using 5-fold cross-validation, and evaluating model performance using six metrics. The imbalanced-class problem is addressed using oversampling, as employed by Kovács [20] and Xu et al. [21], which can significantly improve model performance in classifying instances from the positive class. The hyperparameters of both models were tuned systematically using k-fold cross-validation, as in Passos and Mishra [22] and Nematzadeh et al. [23]. They randomly partitioned the training set into k folds, yielding k pairs of training and validation sets. The performance of both the optimal RF and SVM models will be evaluated by not only four famous metrics, that is, accuracy, precision, recall, and F1 score [24], but also MCC and AUC metrics that should measure the shifting prediction of models to the positive class[25].

The research goals are to develop the best RF and SVM models by oversampling for class imbalance and tuning the models' hyperparameters using cross-validation data. The RF's optimal hyperparameters (minimum number of instances per tree leaf and tree depth) and the SVM's optimal hyperparameters (constant and gamma values) were determined by averaging the k-fold validation accuracies obtained via grid search. The models with optimal hyperparameters of both RF and SVM were trained using either the training set or the oversampled training set. The best model's performance is evaluated on the training and test data using six metrics. Furthermore, the important features of the best RF model are explored. The rest of the sections are organized as follows: reviewing the related works given in Section two, the material data and proposed methods are explored and explained in Section three, the results consisting of the hyperparameter tuning, the model's performance, and the discussion presented in Section four, and finally, Section five presents the conclusion.

II. Related Works

The dataset characteristics, such as high dimensionality and imbalanced classes, tend to influence the machine learning model's performance. In fact, two issues can arise across all aspects of human life, including business (e.g., predicting fraudulent firms [26]), industry (e.g., manufacturing quality prediction [27]), and public health (e.g., cancer classification [28]). Fraud business, unusual products, and cancer patients are minority groups that control the class imbalance in the dataset. Some efforts have been made to handle class imbalance to improve classifier performance. The techniques range from simple methods with basic principles to creating a balanced class through random sampling without replacement, employing oversampling or under sampling methods [29] to sophisticated techniques that were innovated either by hybridization between simulated annealing algorithm for under-sampling and machine learning model [30] or ensemble techniques that concern on combining individual techniques such as under sampling technique, Real Ada-boost, cost-sensitive weight modification, and adaptive boundary decision strategy to acquire classifier models with better performance[31].

The deployment of artificial intelligence across most aspects of human life has made machine learning methods a popular approach and a challenging area of research. In general, machine learning techniques are grouped into unsupervised and supervised methods based on whether a target feature is available [32]. A popular method in unsupervised learning is clustering, which groups similar instances into the same group [33]. The other unsupervised method, namely the ranking method (e.g., the Analytic Hierarchy Process), is often used in decision support systems [34]. When the target feature has a numerical scale, the appropriate model is called a regression model, which can be used to predict [35] or forecast future value [36]. The supervised predictive modeling technique that uses the target feature category is known as a classification model [37]. The categorical target features can be binary [38] or multiclass [39].

Classification models for binary classification can have either a linear decision boundary, such as logistic regression [40] and linear discriminant analysis [41], or a nonlinear one, such as a decision tree [42] and a support vector machine [43]. The decision tree implemented in various fields outperformed the other machine learning models. The ensemble of decision tree models creates a random forest (RF) model that not only mitigates the disadvantages of the decision tree model, such as reducing the risk of overfitting, but also achieves higher performance than the DT model [44]. The RF classifier performed very satisfactorily for the brain-stroke classification by Subudhi et al. [45] and for Arabic sentiment analysis of social media related to COVID-19 conspiracy theories by Al-Hashedi et al. [46]. Nevertheless, similar to the DT model, the RF model also has hyperparameters that should be tuned systematically.

Another classification model with a different approach is the support vector machine (SVM), which uses a hyperplane as the decision boundary [47]. When the decision boundary is complex and nonlinear, SVMs outperform logistic regression [48] and k-means clustering [49]. It also performs comparably to neural network ensembles and boosted shallow trees [50]. The superior performance of both RF and SVM has prompted numerous researchers to compare them, including fraud detection in credit card transactions by Hussain et al. [51], classification of invasive and expansive species by Sabat-Tomala et al. [52], and optimization modeling of callus growth and development in Cannabis sativa done by Hesami and Jones [53]. Based on the above-cited research, RF and SVM show comparable performance, with the difference between the two models not significant. The model's hyperparameters should be adequately tuned to improve the model's performance. Efforts to find the RF-optimal hyperparameters were carried out using various approaches, including the grid search technique [54], Bayesian optimization [55], and an RF model hybridized with a genetic algorithm [56]. On the other hand, the SVM/s hyperparameters, i.e., gamma value and kernel parameter, became the focus of SVM's hyperparameter tuning. Al-Mejibli et al. [57] investigated SVM performance across hyperparameters (e.g., gamma values) and kernel parameters. Wainer and Fonseca [58] employed grid search, random search, Bayesian optimization, simulated annealing, particle swarm optimization, Nelder Mead, and others as methods for SVM tuning hyperparameters.

High-dimensional features and class imbalance are not only common challenges in building machine learning models but can also lead to poor classifier performance. The

implementation of artificial intelligence across most aspects of modern human life has brought machine learning to the forefront as a popular method and a central challenge in the field. It assumes that both RF and SVM will perform satisfactorily, high-performing classifiers for predicting instances with an unknown label when trained with optimal model hyperparameters, so their hyperparameters must be tuned systematically. Comparing the optimal RF and SVM performance remains a challenging research topic, especially for class-imbalanced datasets.

III. The Material and Proposed Method

The dataset containing 52159 patient records for joint replacement surgery is a preprocessed subset of a data mining project. The original data were published as public data and were collected from a Taiwanese hospital in 2021. The target feature, the outcome, had binary classes (infected or uninfected) and should be influenced by 29 predictors consisting of categorical and numerical features. Table I presents the categorical features and their associated label distributions. Table II presents the numerical features and their corresponding basic statistics, i.e., the minimum, maximum, and variance for each feature.

Table I presents all features followed by their class distribution. The target feature (outcome) has two labels: not infected (class_0) or infected (class_1), with a label distribution of [51280, 879], indicating that the target feature is highly class-imbalanced. The not-infected class dominated the infected class, accounting for 98% of the dataset. This means that simply guessing an unknown instance label, without using a classifier, will achieve 98% accuracy. There are 17 categorical predictors in the dataset. where their distribution is 11 features with two classes, two features with three classes, and the remaining features with 4, 6, 14, and 16 classes. Predictors with two classes dominate the others, and their class frequencies are heterogeneous. Nevertheless, the predictor features have a more balanced class distribution than the target feature.

TABLE I. The Categorical Features and Their Label Distribution

Feature Name	Distribution Label	Labels number	Feature Name	Distribution Label	Labels number
Outcome	[51280, 879]	2	Commercial_ALBC	[47355, 4804]	2
Psychoses	[51999, 160]	2	Drain	[38550, 13609]	2
Alcohol Abuse	[51541, 618]	2	Paralysis	[52025, 134]	2
Coagulopathy	[51802, 357]	2	SEX	[34786, 17279]	2
Solid Tumor without Metastasis	[50089, 2070]	2	Non_commercial_ALBC	[30418, 21741]	2
Peptic Ulcer Disease, excluding bleeding	[49244, 2915]	2	Anemia	[51510, 629, 20]	3
Rheumatoid Arthritis/collagen	[49890, 2269]	2	Psychiatric disorder	[51114, 986, 59]	3
Cancer history	[50001, 1964, 193, 1]				4
Diagnosis	[40786, 7023, 2038, 1310, 906, 96]				6
elx_index	[29700, 9366, 6165, 3474, 1768, 898, 464, 188, 74, 32, 18, 9, 2, 1]				14
cci_index	[35002, 8003, 4343, 2301, 1172, 656, 289, 131, 117, 84, 29, 13, 12, 4, 2, 1]				16

TABLE II. The Numerical Features and their Simple Statistics

Feature name	Min.	Max.	Var.
CBC_Platelet	16.1	992	1509.13
AGE	12	99	148.93
LOS	1	72	8.35
OP_time_minute	2	1539	1030.98
OP_time_hour	0.03	25.65	0.29
BUN	1.8	140.67	25.76
GOT	4	15643	4824.68
GPT	2	506	171.28
ALB	2	7.01	0.17
Na	22.1	169.32	3.07
K	2.3	8.13	0.17
UA	1.1	18.74	0.8

The predictors with numerical scales and their simple summaries are displayed in Table II. There seems to be a clear issue with a commensurate measure. Among the features, there are not only significant gaps in the range statistic but also very heterogeneous variances. In addition, the incomparable unit scales are resolved by converting them into standardized normal scores. Finally, all numerical predictors have an N (0,1) distribution and the range of [-3, 3].

The study builds the best RF and SVM classifiers, with the model development stages covering 4 main processes sequentially: addressing the class imbalance with oversampling, using cross-validation data in the hyperparameters tuned, training the model candidates employing the obtained hyperparameters, and assessing the model's performance on six performance metrics for both the training and testing data. Before splitting the dataset into training and testing data. The preprocessing step converts the original scale to Z-scores to ensure a commensurate measure is obtained. Fig. 1 displays the steps for building the SVM classifier model; the steps for building the RF classifier model are similar, except for performing PCA and using both standardized and categorical data as inputs.

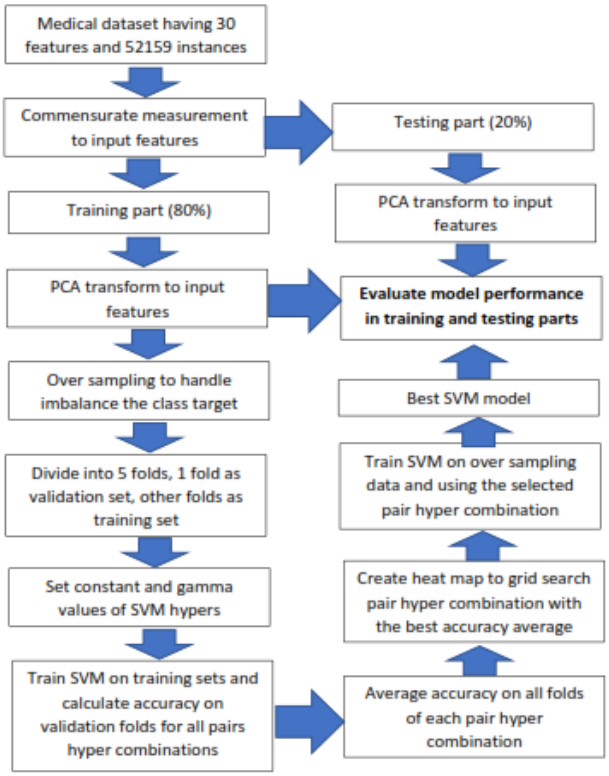


Fig. 1. The steps of the building SVM classifier.

A. The K-Folds Cross-validation to Create Pairs of Training and Validation Sets

Splitting a dataset into training and test data is a fundamental step in machine learning modeling, as the test data is needed to assess how well the model performs. Usually, the dataset is randomly split into 80% for training and 20% for testing. Most machine-learning models involve hyperparameters that should be carefully tuned [59]. The k-fold cross-validation method is a popular approach that requires substantial computational effort to obtain the optimal hyperparameters of machine learning models [60]. Fig. 2 describes splitting the dataset into training and test sets, dividing the training set into 5 folds, and finally creating pairs of training and validation sets. In detail, the stages in producing the training and validation sets are as follows.

- The validation set was the first fold, and all succeeding folds were the training set.
- The validation set was the second fold, and the other folds were the training set.
- The validation set was the third fold, and the other folds were the training set.
- Finally, the validation set was the k-fold, and all preceding folds were the training set.

After the domain values of the hyperparameters that cover the optional ones are determined, the model's candidate with a hyperparameter pair is trained on the training set, and performance is evaluated on the validation set using the accuracy metric. The trained model with the highest average accuracy was selected via grid search [61].

Training part					Testing part
fold_1	fold_2	fold_3	. . .	fold_k	
val_1	Training set_1				
fold_1	**val_2**	fold_3	. . .	fold_k	
fold_1	fold_2	**val_3**	. . .	fold_k	
. . .	. . .	. . .	. . .	. . .	
Training set_k				**val_k**	

Fig. 2. Formatting the dataset to produce pairs of the training and validation sets.

B. Random Forest Classifier Model

The Random Forests (RF) method was proposed by Leo Breiman in 2001. It is an ensemble method designed explicitly for decision tree classifiers. There are two sources of randomness: bagging and a random input vector. Bagging means that each tree was grown using a bootstrapped sample of the data. A random input vector was used at each decision node, where the best split was chosen from a random sample of m features or a subset of attributes instead of all attributes [62].

Fig. 3 depicts the steps for building the RF model. They include selecting the M-dimensional predictors from the dataset, setting m < M, randomly selecting the m predictors, and calculating the information gains (IG). Furthermore, determine the splitting feature with the highest IG value. The RF algorithm was presented as follows[63]:

for $b = 1$ to B,

- Take the bootstrapped dataset (Db) with size N from the D dataset.
- Grow the decision tree (Tb) based on Db, and repeat the steps for each decision node until the leaf node.
  - Select m of M features randomly with (m < M)
  - Compute the IG of the m features
  - Select the best splitting feature and the IG
  - Divide the node into two child nodes
- The RF output was yielded by the majority vote.

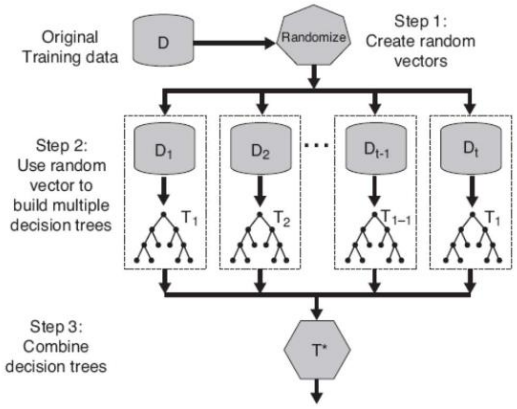


Fig. 3. The concept of developing the RF model.

The RF's hyperparameters, including the tree depth and the number of instances per tree leaf node, should be determined fairly. The grid search method selects the hyperparameter pair with the highest average accuracy obtained from evaluating the trained RF on 5-fold validation data.

C. Modeling Support Vector Classifier Model

The conventional classifier models, namely logistic regression (LR) and Fisher linear discriminant (FDL), do not have a unique decision boundary due to their lack of well-definedness. An example of a well-defined classifier model is a support vector machine (SVM). The decision boundary of the SVM is produced by maximizing the distance between the boundary and the nearest instances. The SVM model is known as the maximum-margin classifier.

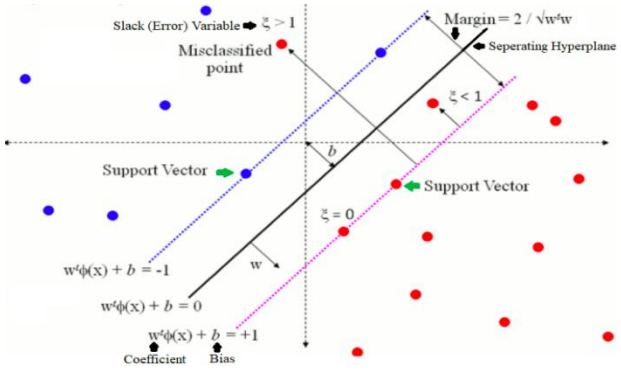


Fig. 4. The SVM model including the decision boundary and support vectors.

The margin of an SVM classifier is the smallest distance between the decision boundary and any training instance, called a support vector[64]. Fig. 4 describes the relationship among the components of an SVM model, including the decision boundary (separating hyperplane), decision margin, and support vectors. A support vector is a training point that lies on the decision-margin line. The SVM concept involves finding support vectors that minimize the distance between the two decision-margin lines. In other words, the SVM task is the same as maximizing the vectors of x nearest to the decision boundary[65].

Fig. 4 depicts three kinds of points in the SVM modeling: points well-classified are $(y_i(w^T \cdot x + w_0) \geq 1)$, points lying within the margin are $(0 < y_i(w^T \cdot x + w_0) < 1)$, and misclassified points are $(y_i(w^T \cdot x + w_0) < 0)$. Eq. (1) describes the three types of points in the SVM model.

$$y_i(w^T \cdot x + w_0) \geq 1 - \xi_i \quad (1)$$

where $\xi_i \geq 0$, is the slack variable. The problem formulation in SVM modeling maximizes the margin and minimizes the slack variable. Mathematically, the formulation is presented by Eq. (2) as follows:

$$\text{minimize } J(\boldsymbol{w}, w_0, \xi) = \frac{1}{2}||w||^2 + C\sum \xi_i \quad (2)$$

Subject to

$$y_i(w^T \cdot x_i + w_0) \geq 1 - \xi_i \; \alpha\nu\delta \; \xi_i \geq 0$$

Consider Eq. (2) and rewrite it by adding the Lagrange multiplier to produce Eq. (3) as follows:

$$\mathcal{L}(w, b, \alpha, \xi) = \frac{1}{2}w^T w + C\sum_i \xi_i - \sum_i \alpha_i(y_i(w^T x_i + b) - 1 + \xi_i) - \sum_i \beta_i \xi_i \quad (3)$$

Where are $\alpha_i \geq 0$, and $\beta_i \geq 0, \forall i = 1,2,3,\ldots,N$. Minimize $\mathcal{L}(w, b, \alpha, \xi)$ through setting $\nabla \mathcal{L}(w, b, \alpha, \xi) = 0$. Furthermore, by rewriting the dual problem formulation given in Eq. (4) as follows [66]:

$$\mathcal{L}(\boldsymbol{w}, b, \alpha) = \sum_i \alpha_i - \frac{1}{2}\sum_i \sum_j \alpha_i \alpha_j y_i y_j x_i x_j \quad (4)$$

Subject to:

$$\sum_i \alpha_i y_i = 0 \; and \; 0 \leq \alpha_i \leq C \quad \forall i = 1,2,3,\ldots,N$$

The inverse of the regularization (C constant) is a crucial hyperparameter for the SVM. The C value describes the behavior as follows: as C increases, the model overfits. In contrast, as C decreases, the model underfits.

Consider the last term of Eq. (4) that can be replaced with a kernel function that serves as a nonlinear mapping from a low-dimensional to a high-dimensional feature space, thereby resulting in a simple form of the optimization problem. The method is known as the kernel trick. Furthermore, the formula for a kernel function is given as follows:

$k(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$ where k and φ respectively are the kernel and the feature map. The decision function of the SVM model in the space of feature-map is stated as follows:

$$y(x) = w^T \emptyset(x) + b \;\; \omega\eta\varepsilon\rho\varepsilon \;\; w = \sum_{n=1}^{N} a_n t_n \emptyset(x_n)$$

An instance x with unknown class is classified by involving a kernel with the formula as follows:

$y(x) = \sum_{n=1}^{N} a_n t_n k(x, x_n) + b$ where t_n is the target class, and the bias b is calculated by the formula as follows:

$$b = \frac{1}{N_{\mathcal{M}}} \sum_{n \in \mathcal{M}} (t_n - \sum_{m \in S} a_m t_m k(x_n, x_m))$$

Both formulas of w and b are obtained by satisfying the constraints of the Karush Kuhn Tucker (KKT).

Whereas the Radial Basis Function (RBF) kernel is presented as follows:

$$k(x, x^l) = exp\left(-\frac{||x - x^l||^2}{2\sigma^2}\right) \quad (5)$$

The denominator in Eq. (5) is referred to as the gamma value only for the RBF kernel. Gamma is a model hyperparameter for the RBF kernel. The gamma value has a behavior: when gamma increases, the SVM overfits, and when gamma decreases, the SVM underfits.

D. Classifier Model Performance Metrics

The metrics for evaluating the classifier's performance are calculated using a confusion matrix. The confusion matrix elements are presented in Eq. (6) as follows:

$$confusion\ matrix = \begin{bmatrix} TN & FN \\ FP & TP \end{bmatrix} \quad (6)$$

Where are TN, FN, FP, and TP representing true negatives, false negatives, false positives, and true positives, respectively

[67]. The study employs the six metrics to evaluate the classifier model's performance, presented in Eq. (7) as follows:

$$\begin{aligned} Accuracy &= \frac{(TP+TN)}{(TP+TN+FP+FN)} \\ Precision &= \frac{TP}{(TP+FP)} \\ Recall &= \frac{TP}{(TP+FN)} \\ F1\ score &= \frac{2\times Recall\times Precision}{(Recall+Precision)} \\ MCC &= \frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)\times(TP+FN)\times(TN+FP)\times(TN+FN)}} \\ AUC &= \int_0^1 ROC(x)dx \end{aligned} \tag{7}$$

Performance metrics, including accuracy, precision, recall, and F1 score, are widely used to evaluate classifier models in binary classification with balanced classes. Two other metrics, namely Matthews Correlation Coefficient (MCC) and Area Under the ROC Curve (AUC), were employed to evaluate the classifier's performance more comprehensively [68].

The MCC metric is widely used to evaluate the performance of classifier models in biomedical research. Both MCC and AUC are elective metrics for reaching a consensus on the best practices for the development and validation of predictive models for personalized medicine[69]. The AUC value ranges from 0.0 to 1.0, describing the capability of the binary classifier to separate instances of the positive class (class_1) from instances of the negative class (class_0).

IV. Performance Metrics of Classification Model

A commensurate measure of the dataset's numerical features is the first issue encountered during preprocessing. The numerical predictors with the heterogeneous unit scale are converted to standardized Z-scores with a mean of zero and a variance of one. The advantage of standardized data is that it is straightforward to identify the extreme values. The value of a feature of an instance is considered an outlier when the absolute Z-score exceeds 3 in the instance where an outlier value of any feature is dropped from the dataset. Although the action entails a risk of dropping an instance from the minority class, further observation to identify the factors underlying the occurrence of extreme features was not straightforward. The former version of the dataset used an incomparable unit scale, and there were no outliers in either the training or testing data.

A. Do Over Sampling and Format the Training and Validation Subsets

The dataset with all predictor features was randomly split into training (80%) and testing (20%) sets. The random forest (RF) and support vector machine (SVM) models were developed using a training set and five pairs of training and validation subsets for hyperparameter tuning. The test set serves as out-of-sample data to evaluate the performance of the best model, i.e., the one with the optimal hyperparameters. Because the dataset is imbalanced, the issue must be addressed by oversampling. For simplicity, the research oversampled the training data only once before dividing the training set randomly into five cross-validation folds. The target feature has a class-label distribution of [class_0, class_1] = [36155, 611], indicating an imbalance. Oversampling was performed on the lower-frequency class (class_1) until its frequency matched that of class_0. Fig. 5 shows that both classes had the same frequency.

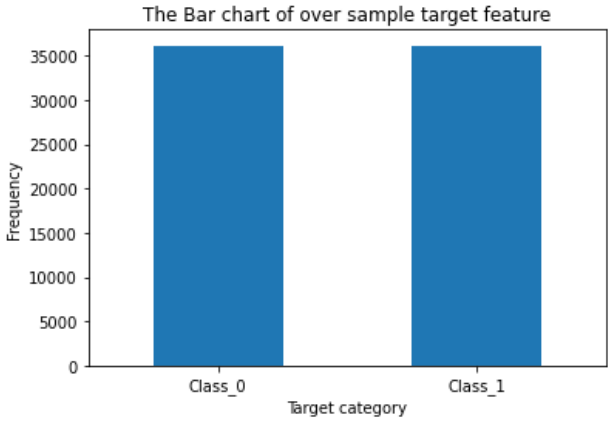


Fig. 5. The class distribution after oversampling.

By applying oversampling, the class_1 frequency increased from 611 to 36155, bringing the total number of instances in the oversampled training set to 72310. Oversampling introduces a trade-off in model training, and hyperparameter tuning requires longer processing time. Furthermore, the oversampled training set was randomly split into five folds, producing five pairs of the training and validation subsets. The number of instances in each validation subset was 14462. The training subset was used to train model candidates with a given associated hyperparameter pair, whereas the validation subset was used to calculate the model's accuracy. Each pair of hyperparameters was used to train a model candidate on five training subsets, producing five accuracy values on five validation subsets. The average accuracy indicates the model's performance on the pair of hyperparameters.

B. Tuning Hyperparameters of Random Forest Classifier

The research builds an RF classifier model that considers not only tree depth but also the number of instances per leaf node as hyperparameters. Both are expected to affect the RF's performance significantly. Their ranges are first determined by trial and error to cover the optimal value. Furthermore, the first hyperparameter was set to [300, 500, 700, 1000, 1200], and the second one to [7, 8, 9, 10, 12, 14]. The RF model candidate was trained by evaluating each hyperparameter pair on four folds of training data, then calculating the accuracy metric on the remaining fold as validation data. There are not only five pairs of training and validation data but also five pairs of hyperparameters; as a result, each pair of training and validation data is used 5 times to train an RF model. The accuracy metric of each trained RF model is calculated on its corresponding validation data. The model's accuracy for a pair of hyperparameters was calculated and recorded. The average accuracy represents the RF model's performance across the pair of hyperparameters. Fig. 6 is a heatmap of the average accuracy across 30 cells.

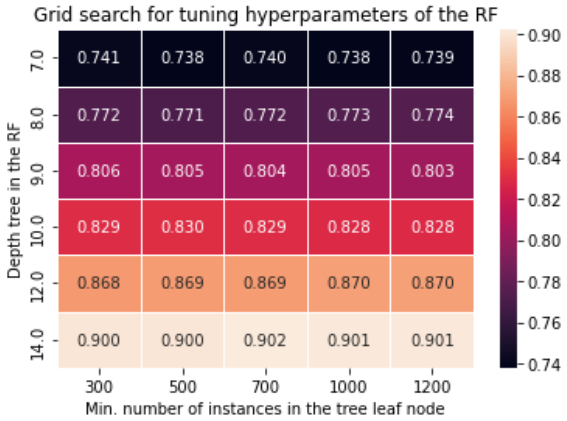


Fig. 6. The average accuracy presented in a heatmap.

Fig. 6 shows the RF's performance: average accuracy increases with tree depth. For example, at a tree depth of 10, the average accuracy increases by around 6% when the tree depth increases by around two points. On the other hand, a different pattern emerges when the tree depth exceeds 10, with the RF's performance increasing less than at the preceding depth level. There is only a marginal increase in the RF's performance of approximately 4% as the tree depth increases from 10 to 12 points. The phenomenon indicates that the model overfits when the RF tree depth exceeds 10. The number of instances per leaf node does not significantly affect the shift in mean accuracy. RF performance at a tree depth of 10 is in the range [0.828, 0.830]. The highest mean accuracy occurs with the cell hyperparameter pair (500, 10). The obtained hyperparameter pair is used to train the RF model on the oversampled training data. The confusion matrix of the optimal RF model is presented for the training and testing data, as presented in Table III:

TABLE III. THE OPTIMAL RF MODEL CONFUSION MATRIX

Actual value	Prediction training		Prediction testing	
	Class_0	Class_1	Class_0	Class_1
Class_0	26329	9826	6512	2511
Class_1	36	575	84	85

Table III displays the confusion matrices for the optimal RF model on both the training and test data with the hyperparameter pair (500, 10). The best RF incorrectly classified 9826 of 36155 instances in the negative class (class_0) on the training data and incorrectly classified 2511 of 9023 instances in class_0 on the testing data. On the other side, the best RF incorrectly classified 36 of 611 instances in the positive class (class_1) in the training data and incorrectly classified 84 of 169 instances in class_1 in the testing data.

C. Tuning Hyperparameter of the Support Vector Classifier Model

In developing the support vector machine (SVM) classifier, the constant and gamma hyperparameters are tuned using cross-validation. The optimal hyperparameter range was determined carefully. First, the SVM model candidate was trained with a given constant value and various gamma values to identify the gamma value with the highest accuracy, and select it as the selected gamma value. Furthermore, the SVM model was retrained with the selected gamma and various constant values. Finally, the sets of gamma and constant values are obtained as [0.0001, 0.0005, 0.001, 0.005, 0.01] and [100, 500, 1000, 3000, 5000, 10000]. For each hyperparameter combination (30 pairs), an SVM model candidate was trained on the training set (the union of the four folds), and the accuracy on the associated validation set (the remaining fold) was calculated. The average accuracy across all folds and gamma-constant pairs is presented as a heat map in Fig. 7.

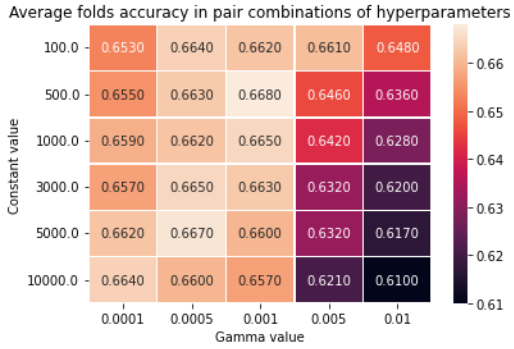


Fig. 7. The heat map of the SVM average accuracy.

Fig. 7 shows the average accuracy of the SVM model candidates in gamma coordinates and constant hyperparameter coordinates. They appear to be randomly distributed and compete tightly with one another. The highest accuracy was achieved with the pair [0.001, 500] for the gamma and constant values, respectively. The hyperparameter pair for the SVM model achieved an average accuracy of 0.668. Furthermore, the hyperparameter pair (0.001, 500) was used to train an SVM candidate model on the oversampled training data, and the confusion matrices for both the training and test sets are presented in Table IV.

TABLE IV. THE SVM MODEL CONFUSION

Actual value	Prediction training		Prediction testing	
	Class_0	Class_1	Class_0	Class_1
Class_0	22813	13342	5461	3562
Class_1	192	419	65	104

Table IV displays the confusion matrix of the best SVM, in which the classifier model incorrectly classified 13342 of 36155 instances of the negative class (class_0) in the training data and incorrectly classified 3562 of 9023 instances of class_0 in the testing data. In addition, the best SVM model incorrectly classified 192 of 611 instances of the positive class (class_1) in the training data and incorrectly classified 65 of 169 instances of class_1 in the testing data. The best SVM model performs better in classifying the class_1 instance in the testing data than the best RF model does. Nevertheless, the best RF outperformed the best SVM on the training data. The best SVM model's average accuracy, displayed in the heat map, showed a random distribution, suggesting it has no evidence of overfitting. Nevertheless, it should be noted that training the SVM model on the 5-fold validation data took longer than training the RF model.

D. The Models' Performance Comparison and the Important Features

The comparison of the best RF and SVM models was conducted in the six performance metrics, i.e., accuracy, precision, recall, F1 score, MCC, and AUC, on both the training and testing data. In addition, the acquired optimal hyperparameters were employed to train both models, using the original training data (rather than the oversampled training data) as the benchmark. Table V displays the benchmark and the best model's performance on both the training and testing data. Table V consists of two primary columns: the first presents RF performance, and the second presents SVM performance. Each primary column contains four columns: the first and second columns present the benchmark model performance, and the two remaining columns display the best model's performance.

TABLE V. THE BEST RF AND SVM PERFORMANCE COMPARISON METRICS

Performance measures	Random Forest (RF)				Support Vector Machine (SVM)			
	Benchmark		*Best model*		Benchmark		Best model	
	Training	*Testing*	*Training*	*Testing*	Training	Testing	Training	Testing
Accuracy	0.984	0.982	0.732	0.718	0.983	0.982	0.632	0.605
Precision	0.980	0.960	0.980	0.970	0.970	0.970	0.970	0.980
Recall	0.980	0.980	0.730	0.720	0.980	0.980	0.630	0.610
F1-score	0.980	0.970	0.830	0.820	0.970	0.980	0.760	0.740
MCC	0.150	0.000	0.190	0.067	0.000	0.000	0.085	0.056
AUC	0.512	0.500	0.835	0.612	0.500	0.500	0.658	0.611

On the training data, the RF benchmark model achieved 98% across four metrics: accuracy, precision, recall, and F1 score. Furthermore, it achieved the MCC and AUC metrics of 15% and 51.2%, respectively. The best RF model performs worse than the RF benchmark across four metrics. However, it achieves the same precision as the RF benchmark and improves by 4% and 32.3% in MCC and AUC metrics, respectively.

The RF model's performance is assessed on the test data; the RF benchmark outperforms the best RF in three metrics, namely accuracy, recall, and F1 score, by 26.4%, 26%, and 15%, respectively. However, the best RF outperforms the RF benchmark in precision, MCC, and AUC, with improvements of 1%, 6.7%, and 11.2%, respectively. The RF benchmark achieves an MCC of 0% and an AUC of 50%, which is equivalent to a coin toss; thus, the RF benchmark model has no performance. Mathematically, an MCC of 0% is obtained when the numerator in the MCC formula equals 0, i.e., when both the true positives (TP) and false positives (FP) are 0. Although the best RF model performed below the RF benchmark on three standard metrics, it significantly outperformed MCC and AUC, which were 6.7% and 61.2%, respectively. Oversampling increased the MCC and AUC by 6.7% and 11.2%, respectively.

When evaluated on the training set, the benchmark SVM model achieved the same precision (97%) as the best SVM model, and the two models showed a performance difference of 35.1%, 35%, and 21% for accuracy, recall, and F1 score, respectively. Furthermore, the best SVM outperformed the benchmark by 8.5% and 15.8% in MCC and AUC, respectively. When evaluating the SVM models on the test set, the benchmark SVM model outperformed the best SVM by 37.7%, 37%, and 24% in accuracy, recall, and F1 score, respectively. In contrast, the best SVM outperformed the benchmark model in precision, MCC, and AUC, with improvements of 1%, 5.6%, and 11.1%, respectively.

The SVM benchmark's performance on the test set is nearly identical to that of the RF benchmark. Both benchmark models perform poorly, as demonstrated by an MCC of 0 and an AUC of 0.5. This explains why they cannot correctly predict labels for unknown positive-class instances on the test set. Extensive attempts have been made to add hyperparameters to produce a well-performing classifier, but to no avail. The best-performing RF model is similar to the best SVM model on both the training and test sets. On the training set, the best RF performs slightly better than the best SVM, with differences of 10%, 1%, 10%, 7%, 10.5%, and 17.7% for accuracy, precision, recall, F1 score, MCC, and AUC, respectively. On the test dataset, the best RF achieves slightly higher accuracy, recall, F1 score, MCC, and AUC than the best SVM, with differences of 11.3%, 11%, 8%, 1.1%, and 0.1%, respectively. The best SVM outperforms the best RF on all five metrics, except precision, by 1%.

The results confirmed that oversampling to address class imbalance during machine learning model training does not always improve model performance. This is similar to the work done by [20], [29], and [30]. The MCC or AUC metrics clearly explain performance improvements. The MCC values increased by 6.7% and 5.6% for the best RF and SVM models, respectively, whereas the AUC values for both models were approximately 11%. Although the best RF model performs slightly better than the best SVM model, hyperparameter optimization for the RF model requires careful attention. Otherwise, there is a potential for overfitting problems[55]. Additionally, the benchmark model identified other findings warranting significant attention. The benchmark model achieves approximately 98% performance across both the training and test sets in four metrics: accuracy, precision, recall, and F1 score. However, both benchmark models performed poorly in the test set, as indicated by an MCC value of 0 and an AUC value of 0.5, where their performance was only equal to the probability of a

balanced coin toss[25]. The event implicitly suggests that using four widely used metrics to evaluate model performance on imbalanced binary classes is misguided.

In machine learning modeling, assessing model performance on test data is crucial because it confirms the model's ability to classify out-of-sample data. This study presents model performance assessments on both training and test data to explicitly demonstrate that assessing model performance based on training data can lead to biased conclusions. The model performed worse on the test data, but the pattern of performance metrics was similar when it did not overfit [52]. Optimally fitting an RF model is constrained by the difficulty in determining the hyperparameter domain that covers optimal values. However, training the RF model on cross-validation data requires less time. On the other hand, the hyperparameter space of an SVM model is straightforward to determine, but training it on cross-validation data is more time-consuming.

Another advantage of building an RF model is that it is a key feature that warrants exploration. The degree of importance of each feature can be used to determine the dominant features or for dimension reduction[10]. Developing a model that involves a few predictor features is preferable because the system mechanism underlying the model can be explained and interpreted well[42]. Fig. 8 presents the degree of essential features produced by the best RF model.

The bar chart in Fig. 8 presents the ranking of important features in increasing order. The numerical features appear to dominate the importance measure, serving as the splitting features in developing the RF model. The first five important features are CBC_Platelet, AGE, K, GOT, and LOS, with importance levels of approximately 10%, 9%, 8%, 8%, and 8%, respectively. This indicates that the five features account for approximately 43% of the RF model's development. The next group comprises four features, each with a degree of importance of approximately 6%, followed by Na and GPT features with approximately 5%. There were five features with a degree of importance of roughly 3%. The importance of the first 16 critical features is approximately 92%. From a dimension-reduction perspective, 16 features account for approximately 92% of the dataset's total variance. The variance explained by feature importance represents the contribution of the original features to the RF model. On the other hand, the variance explained by the PCA transformation is the sum of the variances of the principal components, which are mutually orthogonal and account for the dataset's variance. Note that all 12 numerical features are considered important.

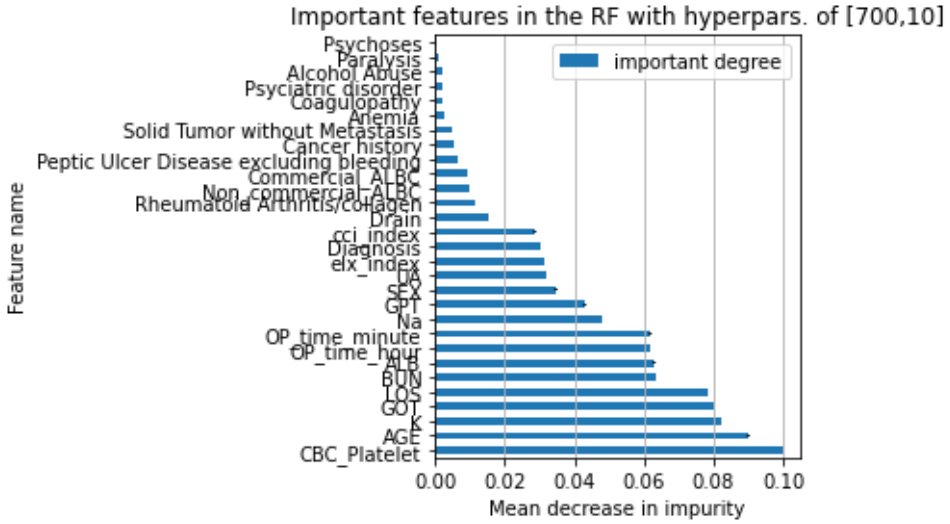


Fig. 8. The best RF model's important features.

V. Conclusion

Oversampling significantly increases the size of the training data, doubling it. This consumes substantial time, particularly when training the SVM model on cross-validation data for hyperparameter tuning. The optimal hyperparameters acquired for the RF model are (500, 10) for the number of instances per tree leaf and the tree depth, respectively. The optimal hyperparameters acquired for the SVM model are (0.001, 500) for the gamma and the constant, respectively. Both RF and SVM benchmarks achieve satisfactory performance on 4 metrics: accuracy, precision, recall, and F1 score, all at 98%. However, they perform worse on 2 metrics: MCC and AUC, with 0% and 50%, respectively. In contrast, the best RF and SVM performed similarly on 6 metrics. Oversampling reduces both the best models across four metrics: accuracy, precision, recall, and F1 score, but it increases on two metrics, namely MCC and AUC. The best RF and SVM outperformed the benchmark on the MCC and AUC metrics. The performance gaps between the best RF and the RF benchmark are (6.7% and 11.2%) for the MCC and AUC metrics, respectively. Whereas the performance gaps between the best SVM and the SVM benchmark are (5.6% and 11.1%) for the MCC and AUC metrics, respectively. In addition, the best RF produces important features that can serve as predictors for other models. In the future, the challenging research topics should focus on building an RF model using the original feature scale, employing MCC or AUC as selection criteria to obtain optimal hyperparameters, and conducting oversampling on the cross-

validation data. They are expected to significantly improve model performance.

REFERENCES

[1] A. W. Widodo, S. Handoyo, G. Irianto, N. W. Hidajati, D. S. Susanti, and I. N. Purwanto, "Finding the Best Performance of Bayesian and Naïve Bayes Models in Fraudulent Firms Classification through Varying Threshold," International Journal of Emerging Technology and Advanced Engineering, vol. 12, no. 12, pp. 94–106, Dec. 2022, doi: 10.46338/ijetae1222_11.

[2] S. Syahrial, R. Ilham, Z. F. Asikin, and St. S. I. Nurdin, "Stunting Classification in Children's Measurement Data Using Machine Learning Models," Journal La Multiapp, vol. 3, no. 2, 2022, doi: 10.37899/journallamultiapp.v3i2.614.

[3] W. Cnota et al., "The Killer Placenta" — a threat to the lives of young women giving Birth by cesarean section," Ginekol Pol, vol. 93, no. 4, 2022, doi: 10.5603/GP.a2021.0235.

[4] T. Liu, W. Fan, and C. Wu, "A hybrid machine learning approach to cerebral stroke prediction based on the imbalanced medical dataset," Artif Intell Med, vol. 101, 2019, doi 10.1016/j.artmed.2019.101723.

[5] I. D. Mienye and Y. Sun, "Performance analysis of cost-sensitive learning methods with application to imbalanced medical data," Inform Med Unlocked, vol. 25, 2021, doi: 10.1016/j.imu.2021.100690.

[6] A. V. Barmak, Y. V. Krak, E. A. Manziuk, and V. S. Kasianiuk, "Information technologyof separating hyperplanes synthesisfor linear classifiers," Journal of Automation and Information Sciences, vol. 51, no. 5, 2019, doi: 10.1615/JAutomatInfScien.v51.i5.50.

[7] Priyanka and D. Kumar, "Decision tree classifier: A detailed survey," International Journal of Information and Decision Sciences, vol. 12, no. 3, 2020, doi: 10.1504/ijids.2020.108141.

[8] T. Lan, H. Hu, C. Jiang, G. Yang, and Z. Zhao, "A comparative study of decision tree, random forest, and convolutional neural network for spread-F identification," Advances in Space Research, vol. 65, no. 8, 2020, doi: 10.1016/j.asr.2020.01.036.

[9] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," Expert Systems with Applications, vol. 134. 2019. doi: 10.1016/j.eswa.2019.05.028.

[10] M. Z. Alam, M. S. Rahman, and M. S. Rahman, "A Random Forest based predictor for medical data classification using feature ranking," Inform Med Unlocked, vol. 15, 2019, doi: 10.1016/j.imu.2019.100180.

[11] S. El-Sappagh, F. Ali, T. Abuhmed, J. Singh, and J. M. Alonso, "Automatic detection of Alzheimer's disease progression: An efficient information fusion approach with heterogeneous ensemble classifiers," Neurocomputing, vol. 512, 2022, doi: 10.1016/j.neucom.2022.09.009.

[12] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," Neurocomputing, vol. 408, 2020, doi: 10.1016/j.neucom.2019.10.118.

[13] W. Pannakkong, K. Thiwa-Anont, K. Singthong, P. Parthanadee, and J. Buddhakulsomsiri, "Hyperparameter Tuning of Machine Learning Algorithms Using Response Surface Methodology: A Case Study of ANN, SVM, and DBN," Math Probl Eng, vol. 2022, 2022, doi: 10.1155/2022/8513719.

[14] X. Ding, J. Liu, F. Yang, and J. Cao, "Random radial basis function kernel-based support vector machine," J Franklin Inst, vol. 358, no. 18, 2021, doi: 10.1016/j.jfranklin.2021.10.005.

[15] A. Kumar, N. Sinha, and A. Bhardwaj, "A novel fitness function in genetic programming for medical data classification," J Biomed Inform, vol. 112, 2020, doi: 10.1016/j.jbi.2020.103623.

[16] J. Bektaş, "EKSL: An effective novel dynamic ensemble model for unbalanced datasets based on LR and SVM hyperplane-distances," Inf Sci (N Y), vol. 597, 2022, doi: 10.1016/j.ins.2022.03.042.

[17] M. A. Ganaie and M. Tanveer, "KNN weighted reduced universum twin SVM for class imbalance learning," Knowl Based Syst, vol. 245, 2022, doi: 10.1016/j.knosys.2022.108578.

[18] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," BMC Genomics, vol. 21, no. 1, 2020, doi: 10.1186/s12864-019-6413-7.

[19] G. Canbek, T. Taskaya Temizel, and S. Sagiroglu, "BenchMetrics: a systematic benchmarking method for binary classification performance metrics," Neural Comput Appl, vol. 33, no. 21, 2021, doi: 10.1007/s00521-021-06103-6.

[20] G. Kovács, "An empirical comparison and evaluation of minority oversampling techniques on a large number of imbalanced datasets," Applied Soft Computing Journal, vol. 83, 2019, doi: 10.1016/j.asoc.2019.105662.

[21] Z. Xu, D. Shen, T. Nie, Y. Kou, N. Yin, and X. Han, "A cluster-based oversampling algorithm combining SMOTE and k-means for imbalanced medical data," Inf Sci (N Y), vol. 572, 2021, doi: 10.1016/j.ins.2021.02.056.

[22] D. Passos and P. Mishra, "A tutorial on automatic hyperparameter tuning of deep spectral modelling for regression and classification tasks," Chemometrics and Intelligent Laboratory Systems, vol. 223. 2022. doi: 10.1016/j.chemolab.2022.104520.

[23] S. Nematzadeh, F. Kiani, M. Torkamanian-Afshar, and N. Aydin, "Tuning hyperparameters of machine learning algorithms and deep neural networks using metaheuristics: A bioinformatics study on biomedical and biological cases," Comput Biol Chem, vol. 97, 2022, doi: 10.1016/j.compbiolchem.2021.107619.

[24] R. Yacouby and D. Axman, "Probabilistic Extension of Precision, Recall, and F1 Score for More Thorough Evaluation of Classification Models," 2020. doi: 10.18653/v1/2020.eval4nlp-1.9.

[25] D. Chicco and G. Jurman, "The Matthews correlation coefficient (MCC) should replace the ROC AUC as the standard metric for assessing binary classification," BioData Min, vol. 16, no. 1, 2023, doi: 10.1186/s13040-023-00322-4.

[26] S. Handoyo, Y. P. Chen, G. Irianto, and A. Widodo, "The varying threshold values of logistic regression and linear discriminant for classifying fraudulent firm," Mathematics and Statistics, vol. 9, no. 2, 2021, doi: 10.13189/ms.2021.090207.

[27] H. Zhou, K. M. Yu, Y. C. Chen, and H. P. Hsu, "A Hybrid Feature Selection Method RFSTL for Manufacturing Quality Prediction Based on a High Dimensional Imbalanced Dataset," IEEE Access, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3059298.

[28] A. S. Sumant and D. V. Patil, "Parallel Ensemble Feature Subset Selection based Deep Learning Approach for Imbalanced High Dimensional Cancer Datasets," Indian Journal of Computer Science and Engineering, vol. 13, no. 6, 2022, doi: 10.21817/indjcse/2022/v13i6/221306034.

[29] T. Wongvorachan, S. He, and O. Bulut, "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining," Information (Switzerland), vol. 14, no. 1, 2023, doi: 10.3390/info14010054.

[30] A. S. Desuky and S. Hussain, "An Improved Hybrid Approach for Handling Class Imbalance Problem," Arab J Sci Eng, vol. 46, no. 4, 2021, doi: 10.1007/s13369-021-05347-7.

[31] W. Lu, Z. Li, and J. Chu, "Adaptive Ensemble Undersampling-Boost: A novel learning framework for imbalanced data," Journal of Systems and Software, vol. 132, 2017, doi: 10.1016/j.jss.2017.07.006.

[32] P. Roy and C. Chowdhury, "A Survey of Machine Learning Techniques for Indoor Localization and Navigation Systems," Journal of Intelligent and Robotic Systems: Theory and Applications, vol. 101, no. 3, 2021, doi: 10.1007/s10846-021-01327-z.

[33] Marji, S. Handoyo, I. N. Purwanto, and M. Y. Anizar, "The effect of attribute diversity in the covariance matrix on the magnitude of the radius parameter in fuzzy subtractive clustering," J Theor Appl Inf Technol, vol. 96, no. 12, 2018.

[34] I. N. Purwanto, A. Widodo, and S. Handoyo, "System for selection starting lineup of a football players by using analytical hierarchy process (AHP)," J Theor Appl Inf Technol, vol. 96, no. 1, 2018.

[35] S. Handoyo, Marji, I. N. Purwanto, and F. Jie, "The fuzzy inference system with rule bases generated by using the fuzzy C-means to predict regional minimum wage in Indonesia," International Journal of Operations and Quantitative Management, vol. 24, no. 4, 2018.

[36] L. Liu, N. Jia, L. Lin, and Z. He, "A Cohesion-Based Heuristic Feature Selection for Short-Term Traffic Forecasting," IEEE Access, vol. 7, 2019, doi: 10.1109/ACCESS.2018.2889814.

[37] W. H. Nugroho, S. Handoyo, and Y. J. Akri, "An influence of measurement scale of predictor variable on logistic regression modeling and learning vector quntization modeling for object classification," International Journal of Electrical and Computer Engineering, vol. 8, no. 1, 2018, doi: 10.11591/ijece.v8i1.pp333-343.

[38] Marji and S. Handoyo, "PERFORMANCE OF RIDGE LOGISTIC REGRESSION AND DECISION TREE IN THE BINARY CLASSIFICATION," J Theor Appl Inf Technol, vol. 100, no. 15, 2022.

[39] M. Koklu and I. A. Ozkan, "Multiclass classification of dry beans using computer vision and machine learning techniques," Comput Electron Agric, vol. 174, 2020, doi: 10.1016/j.compag.2020.105507.

[40] S. Handoyo, N. Pradianti, W. H. Nugroho, and Y. J. Akri, "A Heuristic Feature Selection in Logistic Regression Modeling with Newton Raphson and Gradient Descent Algorithm," International Journal of Advanced Computer Science and Applications, vol. 13, no. 3, 2022, doi: 10.14569/IJACSA.2022.0130317.

[41] N. Han et al., "Transferable Linear Discriminant Analysis," IEEE Trans Neural Netw Learn Syst, vol. 31, no. 12, 2020, doi: 10.1109/TNNLS.2020.2966746.

[42] Y. T. Mursityo, I. Rupiwardani, W. H. N. Putra, D. S. Susanti, T. Handayani, and S. Handoyo, "Relevant Features Independence of Heuristic Selection and Important Features of Decision Tree in the Medical Data Classification," Journal of Advances in Information Technology, vol. 15, no. 5, pp. 591–601, 2024, doi: 10.12720/jait.15.5.591-601.

[43] P. H. Huynh and V. H. Nguyen, "A Novel Ensemble of Support Vector Machines for Improving Medical Data Classification," Engineering Innovations, vol. 4, 2023, doi: 10.4028/p-h0cef4.

[44] M. M. Chen and M. C. Chen, "Modeling road accident severity with comparisons of logistic regression, decision tree, and random forest," Information (Switzerland), vol. 11, no. 5, 2020, doi: 10.3390/INFO11050270.

[45] A. Subudhi, M. Dash, and S. Sabut, "Automated segmentation and classification of brain stroke using expectation-maximization and random forest classifier," Biocybern Biomed Eng, vol. 40, no. 1, 2020, doi: 10.1016/j.bbe.2019.04.004.

[46] A. Al-Hashedi et al., "Ensemble Classifiers for Arabic Sentiment Analysis of Social Network (Twitter Data) towards COVID-19-Related Conspiracy Theories," Applied Computational Intelligence and Soft Computing, vol. 2022, 2022, doi: 10.1155/2022/6614730.

[47] M. Aslani and S. Seipel, "Efficient and decision boundary aware instance selection for support vector machines," Inf Sci (N Y), vol. 577, 2021, doi: 10.1016/j.ins.2021.07.015.

[48] A. Widodo and S. Handoyo, "The classification performance using logistic regression and support vector machine (SVM)," J Theor Appl Inf Technol, vol. 95, no. 19, 2017.

[49] T. Y. Wen and S. A. Mohd Aris, "Hybrid Approach of EEG Stress Level Classification Using K-Means Clustering and Support Vector Machine," IEEE Access, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3148380.

[50] G. Bologna and Y. Hayashi, "A Comparison Study on Rule Extraction from Neural Network Ensembles, Boosted Shallow Trees, and SVMs," Applied Computational Intelligence and Soft Computing, vol. 2018, 2018, doi: 10.1155/2018/4084850.

[51] S. K. Saddam Hussain, E. Sai Charan Reddy, K. Gangadhar Akshay, and T. Akanksha, "Fraud Detection in Credit Card Transactions Using SVM and Random Forest Algorithms," in Proceedings of the 5th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), I-SMAC 2021, 2021. doi: 10.1109/I-SMAC52330.2021.9640631.

[52] A. Sabat-Tomala, E. Raczko, and B. Zagajewski, "Comparison of support vector machine and random forest algorithms for invasive and expansive species classification using airborne hyperspectral data," Remote Sens (Basel), vol. 12, no. 3, 2020, doi: 10.3390/rs12030516.

[53] M. Hesami and A. M. P. Jones, "Modeling and optimizing callus growth and development in Cannabis sativa using random forest and support vector machine in combination with a genetic algorithm," Appl Microbiol Biotechnol, vol. 105, no. 12, 2021, doi: 10.1007/s00253-021-11375-y.

[54] C. G. Siji George and B. Sumathi, "Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction," International Journal of Advanced Computer Science and Applications, vol. 11, no. 9, 2020, doi: 10.14569/IJACSA.2020.0110920.

[55] D. Sun, J. Xu, H. Wen, and D. Wang, "Assessment of landslide susceptibility mapping based on Bayesian hyperparameter optimization: A comparison between logistic regression and random forest," Eng Geol, vol. 281, 2021, doi: 10.1016/j.enggeo.2020.105972.

[56] M. Daviran, A. Maghsoudi, R. Ghezelbash, and B. Pradhan, "A new strategy for spatial predictive mapping of mineral prospectivity: Automated hyperparameter tuning of random forest approach," Comput Geosci, vol. 148, 2021, doi: 10.1016/j.cageo.2021.104688.

[57] I. S. Al-Mejibli, J. K. Alwan, and D. H. Abd, "The effect of gamma value on support vector machine performance with different kernels," International Journal of Electrical and Computer Engineering, vol. 10, no. 5, 2020, doi: 10.11591/IJECE.V10I5.PP5497-5506.

[58] J. Wainer and P. Fonseca, "How to tune the RBF SVM hyperparameters? An empirical evaluation of 18 search algorithms," Artif Intell Rev, vol. 54, no. 6, 2021, doi: 10.1007/s10462-021-10011-5.

[59] W. H. Nugroho, S. Handoyo, H. C. Hsieh, Y. J. Akri, Zuraidah, and D. Dwinita Adelia, "Modeling Multioutput Response Uses Ridge Regression and MLP Neural Network with Tuning Hyperparameter through Cross Validation," International Journal of Advanced Computer Science and Applications, vol. 13, no. 9, 2022, doi: 10.14569/IJACSA.2022.0130992.

[60] T. Yan, S. L. Shen, A. Zhou, and X. Chen, "Prediction of geological characteristics from shield operational parameters by integrating grid search and K-fold cross validation into stacking classification algorithm," Journal of Rock Mechanics and Geotechnical Engineering, vol. 14, no. 4, 2022, doi: 10.1016/j.jrmge.2022.03.002.

[61] G. N. Ahmad, H. Fatima, Shafiullah, A. Salah Saidi, and Imdadullah, "Efficient Medical Diagnosis of Human Heart Diseases Using Machine Learning Techniques with and Without GridSearchCV," IEEE Access, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3165792.

[62] Z. Chai and C. Zhao, "Enhanced random forest with concurrent analysis of static and dynamic nodes for industrial fault classification," IEEE Trans Industr Inform, vol. 16, no. 1, 2020, doi: 10.1109/TII.2019.2915559.

[63] L. Gigović, H. R. Pourghasemi, S. Drobnjak, and S. Bai, "Testing a new ensemble model based on SVM and random forest in forest fire susceptibility assessment and its mapping in Serbia's Tara National Park," Forests, vol. 10, no. 5, 2019, doi: 10.3390/f10050408.

[64] A. Murugan, S. A. H. Nair, and K. P. S. Kumar, "Detection of Skin Cancer Using SVM, Random Forest and kNN Classifiers," J Med Syst, vol. 43, no. 8, 2019, doi: 10.1007/s10916-019-1400-8.

[65] Y. Zhang, D. Xiao, and Y. Liu, "Automatic Identification Algorithm of the Rice Tiller Period Based on PCA and SVM," IEEE Access, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3089670.

[66] D. D. Shankar and A. S. Azhakath, "Minor blind feature based Steganalysis for calibrated JPEG images with cross validation and classification using SVM and SVM-PSO," Multimed Tools Appl, 2020, doi: 10.1007/s11042-020-09820-7.

[67] A. W. Widodo, S. Handoyo, I. Rupiwardani, Y. T. Mursityo, I. N. Purwanto, and H. Kusdarwati, "The Performance Comparison between C4.5 Tree and One-Dimensional Convolutional Neural Networks (CNN1D) with Tuning Hyperparameters for the Classification of Imbalanced Medical Data," International Journal of Intelligent Engineering and Systems, vol. 16, no. 5, pp. 748–759, 2023, doi: 10.22266/ijies2023.1031.63.

[68] D. Chicco and G. Jurman, "A statistical comparison between Matthews correlation coefficient (MCC), prevalence threshold, and Fowlkes–Mallows index," J Biomed Inform, vol. 144, 2023, doi: 10.1016/j.jbi.2023.104426.

[69] S. Boughorbel, F. Jarray, and M. El-Anbari, "Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric," PLoS One, vol. 12, no. 6, 2017, doi: 10.1371/journal.pone.0177678.

K-Nearest Neighbors Algorithm for Short-to-Medium Term Directional Stock Price Forecasting: An Analysis of Thailand's Banking Sector

Passawan Noppakaew*, Parit Wanitchatchawan, Kanchana Phuhoy, Natthasorn Seubwong
Faculty of Science-Department of Mathematics, Silpakorn University, Nakhon Pathom, 73000, Thailand

***Abstract*—This research explores the efficacy of a parsimonious K-Nearest Neighbors (KNN) framework for short-to-medium term stock direction forecasting, focusing specifically on the banking sector within Thailand's SET50 index. Prior preliminary analysis, aimed at determining the optimal prediction horizon, indicated that a 60-day forecast yielded the most effective results, establishing the scope of this study as medium-term prediction. The objective of this analysis is to determine if a 60-day directional movement can be effectively captured using a minimalist feature set limited to the current day's Opening Price and the previous day's 14-day Simple and Exponential Moving Averages. Employing a rolling-window validation methodology on seven key banking stocks during H1 2025, the KNN model demonstrated significant predictive capability. The average accuracy across the selected banking stocks reached 82.0%, with standout performance for TISCO and SCB. While results varied across stocks, our findings substantiate the theoretical and practical sufficiency of a simplicity-first approach. The research demonstrates that in high-noise emerging markets, feature sparsity and instance-based logic serve as an essential defense against overfitting, providing institutional practitioners with a transparent and robust alternative to complex methodologies.**

***Keywords*—Stock trend prediction; K-Nearest Neighbors; technical analysis**

I. Introduction

The intricate, volatile, and non-linear nature of financial markets presents a formidable and long-standing challenge [1]. Accurately forecasting stock price movements has been a desirable objective for both investors and academics [2], [3], as marginal improvements in predictive accuracy can offer significant financial advantages. Based on the limitations of traditional linear models for forecasting stock movements, alternative approaches have been explored. Machine learning (ML) which is a sub-field of artificial intelligence (AI) has emerged as a powerful paradigm for identifying the complex latent structures within financial time-series data [4], [5], [6]. See also [7], [8], [9], [10] for alternative approaches using deep learning techniques.

Unlike conventional models, ML algorithms excel at identifying non-linear patterns and hidden correlations within high-dimensional datasets [11]. This capability has led to widespread applications across various domains, from natural language processing and computer vision to intelligent recommendation engines [12]. The research efforts within ML-based financial forecasting can be broadly divided into two main approaches: one focusing on price prediction and the other on market directional forecasting. The first approach treats the problem as a regression task, attempting to forecast the exact future value or closing price of a stock. While academically rigorous, this method encounters significant challenges due to the high level of stochastic noise in market data. The second approach, which is the focus of this study, frames the problem as a classification task, predicting the directional movement, such as upward or downward trends, of the stock price. This classification approach offers distinct practical advantages. For a trader or investor, their decision is often binary, i.e., buy or sell, not the prediction of an exact price point. A model that correctly classifies the direction of a trend, even if it misses the magnitude, provides a clearer and more actionable signal for formulating trading strategies. The focus on direction rather than price is often more robust and less susceptible to the noise that plagues regression models.

Among the suite of ML tools, the K-Nearest Neighbors (KNN) algorithm, a non-parametric and instance-based learning method, has attracted significant attention due to its simplicity, transparency, and effectiveness [13]. KNN operates on the principle of feature similarity, classifying a new data point based on the majority class of its k nearest neighbors in the feature space. Due to its reliance on local similarity rather than global parameter estimation, KNN-based frameworks are effectively utilized to mitigate overfitting in noisy financial time series, particularly when coupled with strategic feature selection [14]. Its utility has been validated in diverse fields, such as the trading system proposed by Teixeira and de Oliveira [15], who utilized a high-dimensional feature space of 22 technical indicators to achieve significant results. However, recent advancements in emerging market research suggest a strategic shift toward binary directional classification as a more robust objective than point-based price estimation [6]. Specifically, in the context of the Stock Exchange of Thailand (SET), evidence indicates that focusing on market direction allows for higher predictive stability despite the inherent volatility of the Thai banking sector [6].

Consequently, this study diverges from the feature-rich precedent by investigating a minimalist, direction-oriented hypothesis. Rather than adopting the extensive 22-indicator framework, the current research examines whether effective directional forecasting for the Thai banking sector is achievable through a deliberately restricted feature set, thereby enhancing both model transparency and practical interpretability. This study investigates the hypothesis that a simple KNN classifier can effectively perform directional forecasting even with minimal input data. The analytical focus is narrowed specifically to

*Corresponding author.

stocks within the Banking sector, as listed in Thailand's SET50 Index for the H1 2025 period, namely BBL, KBANK, KTB, KTC, SCB, TISCO, and TTB. This sector presents a particularly interesting case for study due to several factors: bank stocks often exhibit a degree of sensitivity to macroeconomic indicators, their performance can be interrelated, reflecting broader economic health, and they represent a significant component of the overall market index, making their predictability valuable for investors. Within this focused scope, the objective is to predict whether the closing price 60 days in the future will be up (higher) or down (lower) relative to the current day's opening price. This prediction utilizes a deliberately minimalist feature set consisting of: 1) the current day's Opening Price, 2) the 14-day Simple Moving Average (SMA) from the previous day, and 3) the 14-day Exponential Moving Average (EMA) also from the previous day. The use of opening prices serves to enhance economic realism by simulating decision-making at the start of the trading session. By evaluating this model specifically on these key banking stocks, this research aims to determine if the minimalist KNN approach provides a practical and effective basis for directional forecasting within this vital market sector. To distinguish the unique contributions of this study from existing KNN-based forecasting literature, this study pivots toward a parsimonious framework that challenges the prevailing necessity of model complexity in high-noise environments. Methodologically, this study demonstrates that a minimalist feature set can maintain high predictive accuracy by prioritizing robustness over complexity. Such an approach serves as a strategic defense against the overfitting typically encountered in algorithmically dense architectures within high-noise environments. Beyond its empirical setting, the choice of the Thai banking sector serves as a strategic economic bellwether. Grounded in the theoretical consensus that financial systems are primary engines of national stability [16], [17], this research leverages the heightened sensitivity of these securities to macroeconomic shifts as a rigorous stress test for model reliability [18]. Finally, the research offers significant practical utility by providing a transparent, instance-based logic for institutional practitioners. Unlike complex models, the KNN framework offers superior interpretability, allowing users to substantiate forecasts through historical analogies–a vital requirement for real-world trading deployment.

The remainder of this study is organized into four primary sections to provide a comprehensive overview of the research framework: Section II details the research methodology, specifically focusing on the data selection process within the Thai banking sector and the implementation of the deliberately minimalist feature set. Section III reports the empirical results and predictive performance across the selected constituents. Finally, Section IV provides a synthesis of key findings, summarizing the practical utility of the parsimonious model and offering strategic recommendations for subsequent research directions.

II. Methodology

This study employs a quantitative methodology to ascertain the effectiveness of a KNN model for financial market classification. The process involves data acquisition from a major emerging market index, precise variable features, and a model training and validation process.

A. Dataset and Scope

Instead of serving as a nominal constraint for this study, the focus on the Thai banking sector represents a strategic decision to test the proposed model within a pivotal economic bellwether [17]. In an emerging market such as Thailand, the banking industry acts as a direct proxy for broader national stability, characterized by its acute sensitivity to monetary policy shifts, interest rate volatility, and macroeconomic fluctuations [16], [18]. This inherent sensitivity creates a high-variance and information-rich environment which is an ideal test for evaluating whether medium-term technical indicators can maintain predictive reliability under shifting conditions. To uphold the analytical reliability of this research and shield the model from the stochastic noise inherent in low-volume securities, this study deliberately focuses on banking constituents within the SET50 Index. This approach ensures the study is grounded in the market's most liquid and institutionally significant securities.

For this study, the banking constituents were identified from the official SET50 list [19] for the H1 2025 period starting from January 1, 2025, to June 30, 2025, namely BBL, KBANK, KTB, KTC, SCB, TISCO, and TTB, as shown in Table I. Concentrating on this sector allows a focused examination of the model's performance within a group of stocks often influenced by similar macroeconomic factors.

TABLE I. The Banking Sector in SET50 Index (H1 2025: Jan 1 - Jun 30, 2025)

No.	Symbol	Company Name
1	BBL	BANGKOK BANK PUBLIC COMPANY LIMITED
2	KBANK	KASIKORNBANK PUBLIC COMPANY LIMITED
3	KTB	KRUNG THAI BANK PUBLIC COMPANY LIMITED
4	KTC	KRUNGTHAI CARD PUBLIC COMPANY LIMITED
5	SCB	SCB X PUBLIC COMPANY LIMITED
6	TISCO	TISCO FINANCIAL GROUP PUBLIC COMPANY LIMITED
7	TTB	TMBTHANACHART BANK PUBLIC COMPANY LIMITED

All necessary historical time-series data (including daily Open, High, Low, Close, and Volume) for these selected banking stocks were programmatically retrieved using the Yahoo Finance (yfinance) library. To establish an appropriate prediction horizon for the main study, a preliminary experiment was conducted. This involved evaluating the KNN model's average prediction accuracy across various forward-looking horizons (n days), ranging from 3 to 120 days, while keeping the feature set constant. The results of this sensitivity analysis indicated that model balances between achieving near-maximal predictive performance and maintaining a forecast window appropriate for short-to-medium-term trading strategies at $n = 60$ days (details in Appendix A, Appendix B, and Appendix C). Therefore, a 60-day prediction horizon was adopted for the final evaluation. The final out-of-sample prediction (testing) phase, using this 60-day target, was conducted on data from January 1, 2025, to June 30, 2025.

B. Study Design and Variable Engineering

This research is designed as a binary classification task. The objective is to predict the directional movement of a stock's price over a fixed, 60-day forward-looking horizon.

1) Target variable (dependent variable): For any given trading day t, the target variable $Y(t)$ is defined as the future price direction. It is assigned a binary value based on a comparison between the closing price on day t and the closing price 60 days later ($t + 60$):

- Class 1 (Up) if $Close_{t+60} > Close_t$,
- Class 0 (Down) if $Close_{t+60} \leq Close_t$,

where, $Close_t$ represents the closing price on the prediction day (day t) and $Close_{t+60}$ is the closing price 60 days later. This definition is designed to directly model the practical trading decision of determining whether buying a stock at today's closing price will yield a higher closing price 60 days from now.

2) Feature set: Unlike models that rely on a high-dimensional feature space, this study adopts a deliberately minimal approach. Intricate feature engineering is sidestepped in favor of utilizing fundamental and widely recognized technical indicators common in financial analysis. Specifically, the feature set is constructed solely from the current day's Opening Price, alongside the previous day's 14-day Simple Moving Average (SMA) and 14-day Exponential Moving Average (EMA). Further details on these moving averages are provided in [20]. This deliberate simplicity allows for a focused evaluation of the core predictive power inherent in these basic trend metrics.

The resulting feature set $X(t)$, used to predict the outcome for day t consists of only three variables:

- Open Price ($Open_t$) is the opening price on the day of prediction.
- SMA-14 ($SMA_{14(t-1)}$) is the 14-day Simple Moving Average, calculated using closing prices up to the previous trading day ($t - 1$).
- EMA-14 ($EMA_{14(t-1)}$) is the 14-day Exponential Moving Average, also calculated using data up to the previous trading day ($t - 1$).

The 14-day lookback period for the moving averages was selected to align with the empirically determined optimal prediction horizon (see Appendix A, Appendix B, and Appendix C) and reflects common practice in short-to-medium-term technical analysis. Fine-tuning the moving average periods was considered beyond the scope of this study, which focuses on the effectiveness of the chosen minimalist feature types. This specific feature construction is critical, as it ensures that only information known at the time of prediction (the morning of day t) is used, thereby preventing any lookahead bias.

3) Model training and validation: The KNN algorithm was selected as the core classifier. To simulate realistic market conditions and ensure model robustness, a rolling-window validation procedure was implemented.

For each of the 7 stocks, and for each day in the test period (Jan-Jun 2025), the model was trained as follows:

- Data Window: A historical data window of approximately 1 year was retrieved relative to the prediction day (e.g., for a prediction on June 1, 2025, data from June 1, 2024, to May 31, 2025, will be used).
- K-Value Optimization: This historical window was split into a training set and a testing (validation) set (3:1 ratio). The KNN model was run iteratively to find the optimal hyperparameter k (the number of neighbors) that yielded the highest accuracy on this validation set.
- Prediction: The full historical window was then used as the training set, and the optimized k and the model created by the historical data window ($\approx$ 1 year) was used to make the single directional prediction, which is class 0 or 1, for the target day.

This process was repeated for every trading day in the test period, ensuring the model was always retrained on the most recent data available before making a new prediction. The final accuracy, precision and recall were calculated as the percentage of correctly predicted days over the entire 6-month test period. Minimalist approach provides a practical and effective basis for KNN-based directional forecasting.

III. Results

This section presents the empirical findings from applying the KNN classification model, configured with the minimalist feature set and a 60-day prediction horizon, to the selected banking stocks within Thailand's SET50 Index (H1 2025). The model's performance is evaluated on a per-stock basis using standard classification metrics, focusing primarily on prediction accuracy, recall, and precision, over the designated test period. The predictive performance of the KNN model across the selected banking stocks is detailed in Table II and visualized in Fig. 1. Overall, the model demonstrates considerable effectiveness for most stocks within this sector, although performance varies significantly between individual securities.

TABLE II. KNN Model Performance Metrics for SET50 Banking Stocks (H1 2025).

Stock Symbol	Accuracy (%)	Precision (%)	Recall (%)
BBL	61.0	40.5	44.7
KBANK	83.9	93.4	86.7
KTB	80.5	84.9	92.8
KTC	84.7	72.3	100.0
SCB	89.8	93.7	95.4
TISCO	94.9	96.3	98.1
TTB	78.8	91.3	80.2
Average	**82.0**	**81.8**	**85.4**

Both SCB and TISCO exhibit strong performance with high accuracy, precision and recall. Especially, TISCO emerges as the top performer, achieving exceptionally high scores across all metrics. This suggests that the model is highly reliable and effective at predicting the 60-day directional movement for these specific stocks. Whereas, several other banks, including KBANK, KTB, and KTC, also show robust results, generally achieving accuracies above 80%. Especially, KTC displays a perfect recall score, indicating the model successfully identified all actual upward movements within the test period for this stock. However, this comes with a lower precision compared to KBANK and SCB, implying a higher rate of false positives. KTB also shows a strong recall paired with moderate precision. TTB presents good precision, but

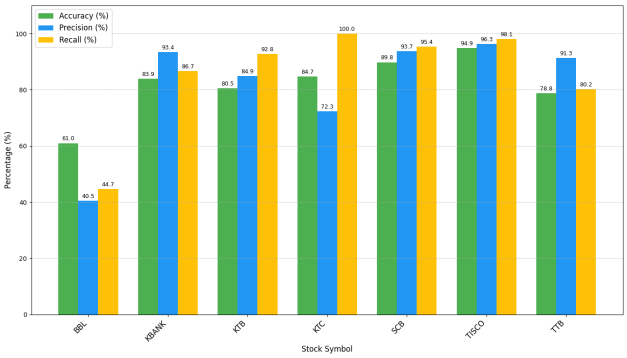


Fig. 1. Bar chart of KNN performance metrics for banking stocks.

lowers recall compared to the top performers. In contrast, BBL shows significantly weaker results across all aspects of the performance metrics. The model struggles considerably with predicting this stock's movement compared to the others in the sector.

Considering the average performance across the seven banking stocks, the KNN model with the selected features appears generally proficient for this sector. The high average recall suggests the model is good at capturing upward trends, while the respectable average precision indicates that when it predicts an upward movement, it is correct about 82% of the time on average.

IV. Conclusion

This research set out to explore the viability of a simplified approach to predict stock price direction within the Thai financial market. The study specifically investigated whether a KNN classifier, equipped with a deliberately minimalist set of technical indicators (Opening Price, 14-day SMA, 14-day EMA), could effectively forecast medium-term (60-day) upward and downward movements for stocks in the SET50 banking sector. The selection of the 60-day horizon was itself informed by a preliminary sensitivity analysis aiming for a balance between predictive power and practical trading relevance.

The study showed that the KNN model demonstrated considerable predictive capability, achieving an average accuracy of 82.0% across the sector. Performance was particularly strong for certain stocks like TISCO (94.9% accuracy) and SCB (89.8% accuracy), suggesting the model can be highly effective in specific cases. The findings indicate that even without complex feature engineering or computationally intensive algorithms, a basic KNN approach can capture meaningful predictive signals for medium-term directional changes, at least within this homogeneous sector. The high average recall (85.4%) further suggests a particular aptitude for identifying potential upward trends. However, the results were not uniform. The model's significantly lower performance for BBL underscores that predictive success, even with a consistent methodology, can be highly stock-specific.

The robust accuracy achieved through this minimalist framework underscores the theoretical and practical sufficiency of a simplified KNN approach, even in the absence of complex algorithmic benchmarks. These findings suggest that in high-noise environments like the SET50 banking sector, feature sparsity and low model complexity may actually serve as a primary defense against the overfitting typically observed in more elaborate architectures. By demonstrating that significant market signals can be effectively captured without the inherent computational overhead of complex models, this study provides a compelling case for a parsimonious modeling framework. Furthermore, the instance-based logic of KNN offers superior interpretability—a vital requirement for practitioners who prioritize understanding the historical analogies driving a prediction as much as the forecast itself.

Future research could productively expand upon this work by comparing the minimalist KNN approach against other classifiers (e.g., Logistic Regression, Naive Bayes) and applying the methodology to diverse market sectors and international markets. In addition, future research will investigate adaptive methods for selecting the optimal prediction horizon and moving average periods. Despite its limitations, this study affirms the potential utility of simple, interpretable models like KNN in the challenging domain of financial market prediction.

Acknowledgment

The authors wish to thank Department of Mathematics, Faculty of Science, Silpakorn University. We would also like to thank anonymous reviewers for valuable comments and suggestions.

References

[1] E. F. Fama, *Efficient capital markets: A review of theory and empirical work*, The Journal of Finance, vol. 25, no. 2, pp. 383–417, 1970.

[2] P. C. Tetlock, *Giving content to investor sentiment: The role of media in the stock market*, The Journal of Finance, vol. 62, no. 3, pp. 1139–1168, 2007.

[3] R. S. Tsay, *Analysis of Financial Time Series*, 3rd ed. Hoboken, NJ: John Wiley & Sons, 2010.

[4] W. Huang, Y. Nakamori, and S. Y. Wang, *Forecasting stock market movement direction with support vector machine*, Computers & Operations Research, vol. 32, no. 10, pp. 2513–2522, 2005.

[5] S. Basak, S. Kar, S. Saha, L. Khaidem, and S. R. Dey, *Predicting the direction of stock market prices using tree-based classifiers*, The North American Journal of Economics and Finance, vol. 47, pp. 552–567, 2019.

[6] K. Saetia and J. Yokrattanasak, *Stock Movement Prediction Using Machine Learning Based on Technical Indicators and Google Trend Searches in Thailand*, International Journal of Financial Studies, vol. 11, no. 1, Art. no. 5, 2023.

[7] E. W. Saad, D. V. Prokhorov, and D. C. Wunsch, *Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks*, IEEE Transactions on Neural Networks, vol. 9, no. 6, pp. 1456–1470, 1998.

[8] G. S. Atsalakis and K. P. Valavanis, *Surveying stock market forecasting techniques – Part II: Soft computing methods*, Expert Systems with Applications, vol. 36, no. 3, pp. 5932–5941, 2009.

[9] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, *Financial time series forecasting with deep learning: A systematic literature review: 2005–2019*, Applied Soft Computing, vol. 90, p. 106181, 2020.

[10] A. S. Baimuruev and A. B. Zharmagambetov, *Review of artificial intelligence methods and applications*, Journal of Physics: Conference Series, vol. 1889, no. 2, p. 022014, 2021.

[11] S. B. Kotsiantis, *Supervised machine learning: A review of classification techniques*, Informatica, vol. 31, pp. 249–268, 2007.

[12] M. I. Jordan and T. M. Mitchell, *Machine learning: Trends, perspectives, and prospects*, Science, vol. 349, pp. 255–260, 2015.

[13] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd ed. Upper Saddle River, NJ: Pearson, 2010.

[14] S. Lahmiri and S. Bekiros, *Randomness, informational entropy, and volatility interdependencies among major world markets: A k-nearest neighbors approach*, Chaos, Solitons & Fractals, vol. 132, Art. no. 109509, 2020.

[15] L. A. Teixeira and A. L. I. de Oliveira, *A method for automatic stock trading combining technical analysis and nearest neighbor classification*, Expert Systems with Applications, vol. 37, no. 10, pp. 6885–6890, 2010.

[16] R. G. King and R. Levine, *Finance and Growth: Schumpeter Might Be Right*, The Quarterly Journal of Economics, vol. 108, no. 3, pp. 717–737, 1993.

[17] R. Levine, *Financial Development and Economic Growth: Views and Agenda*, Journal of Economic Literature, vol. 35, no. 2, pp. 688–726, 1997.

[18] L. Menkhoff and C. Suwanaporn, *10 Years after the crisis: Thailand's financial system reform*, Journal of Asian Economics, vol. 18, no. 1, pp. 4–20, 2007.

[19] The Stock Exchange of Thailand (SET). (2025). *Constituents List SET50 & SET100*. [Online]. Available: https://www.set.or.th/en/market/information/securities-list/constituents-list-set50-set100 [Accessed: June 14, 2025].

[20] J. J. Murphy, *Technical Analysis of the Financial Markets*. New York, NY: New York Institute of Finance, 1999.

APPENDIX

Before proceeding with the main analysis, it was crucial to empirically justify the selection of the prediction horizon (n). An arbitrary choice could weaken the study's validity. Therefore, a sensitivity analysis was conducted to evaluate the KNN model's performance across various forward-looking timeframes.

A. Methodology

The core experimental setup mirrored the main study's design, with only the prediction horizon (n) being varied.

- Horizons Tested: A range of plausible short-to-medium-term horizons was selected for comparison. The parameter n was defined using the following intervals: $3, 5, 7, 10, 14, 21, 30, 45, 60, 90, 120$ days.
- Target Variable Definition: For each tested horizon n, the binary target variable $Y(t)$ was redefined as:
  - Class 1 (Up) if $Close_{t+n} > Close_t$,
  - Class 0 (Down) if $Close_{t+n} \leq Close_t$.
- Constant Factors: The minimalist feature set (current Open Price, previous day's SMA-14, previous day's EMA-14) and the KNN algorithm remained constant across all tests.
- Optimal k Selection: For each stock and each prediction day, the optimal number of neighbors (k) was determined. This involved:
  - Splitting the 1-year historical data starting from January 1, 2024, to December 31, 2024 into a 3:1 training/validation set.
  - Iterating through k values from 1 to 30.
  - Selecting the k that yielded the highest accuracy on the validation set for that specific stock and that specific training window.
- Performance Evaluation: Using the optimized k, predictions were made for the target day within the rolling window procedure. This entire process was repeated for all 50 stocks across the full test period (October 1, 2024, to December 31, 2024) for each of the selected horizons ($n = 3, 5, ..., 120$). The performance metric was the average prediction accuracy, precision and recall calculated across all 50 stocks for each horizon.

B. Results

The key metrics tracked were average accuracy, precision, and recall. The results are presented in Table III and visualized in Fig. 2.

TABLE III. AVERAGE PERFORMANCE METRICS VS. PREDICTION HORIZON (n DAYS).

Prediction	Average Metrics (%)		
Horizon (n days)	Accuracy	Precision	Recall
3	61.0	56.0	47.0
5	65.0	65.0	54.0
7	68.0	67.0	66.0
10	69.0	67.0	64.0
14	70.0	70.0	67.0
21	73.0	74.0	70.0
30	75.0	75.0	74.0
45	75.0	78.0	73.0
60	77.0	78.0	78.0
90	77.0	79.0	78.0
120	79.0	80.0	80.0

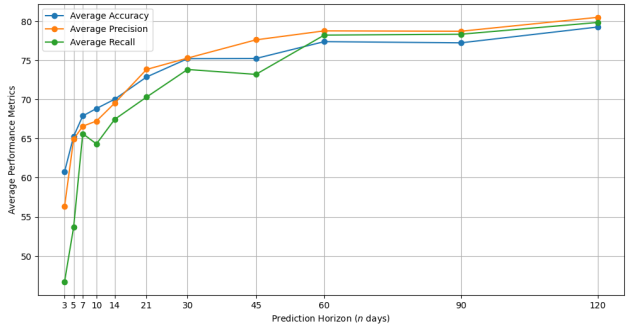


Fig. 2. Average performance metrics vs. prediction horizon (n days) up to 120 days.

As depicted in Fig. 2, all three performance metrics generally exhibit an upward trend across the entire tested range. Notably, the rate of improvement decelerates considerably after approximately $n = 45$ days. Average Precision, in particular, begins to plateau around $n = 60$ days, showing only marginal increases thereafter up to $n = 120$. While the highest average accuracy $\approx 79\%$ was observed at $n = 120$, the gain compared to $n = 60$ which has the average accuracy $\approx 77\%$ is relatively small, especially considering the doubling of the forecast period from approximately two months to four months. Moreover, $n = 120$ is beyond the typical scope of short-to-medium-term trading. Therefore, a horizon of $n = 60$ days (approximately two months) was chosen for this study. This decision represents a deliberate trade-off. It captures the vast majority of the predictive performance achievable while maintaining a forecast window that remains relevant and actionable for medium-term position adjustments.

C. Conclusion

Based on this empirical evidence, the 60-day prediction horizon ($n = 60$) was selected as the most suitable timeframe for the main classification task in this study. The sensitivity analysis in Fig. 2 supports this choice, indicating significantly diminishing improvement rates in performance gains for horizons substantially longer than 60 days.

Multi-Criteria Methodology for Selecting Communication Protocols in M2M Environments

Oleg Iliev
Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Sofia, Bulgaria

***Abstract*—The continuous expansion of Machine-to-Machine (M2M) communication and Internet of Things (IoT) ecosystems has significantly increased the complexity of selecting appropriate communication protocols and data flow management systems. Contemporary M2M deployments operate across heterogeneous functional domains, including sensor networks, transactional systems, and real-time streaming environments, each imposing distinct and often conflicting non-functional requirements such as latency, reliability, scalability, and resource efficiency. This study proposes a domain-oriented multi-criteria decision-making methodology for structured protocol selection in M2M environments. The framework integrates the Analytic Hierarchy Process (AHP) for context-dependent weighting of evaluation criteria with the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) for quantitative ranking of alternative technologies. A structured domain taxonomy is introduced to dynamically align evaluation priorities with functional deployment characteristics, and 99th percentile latency (L_{p99}) is incorporated as a primary performance indicator to capture tail-behavior effects critical for M2M reliability. Beyond ranking computation, the methodology formalizes a reproducible analytical workflow linking empirical measurements to domain-specific decision outcomes and incorporates a sensitivity-analysis perspective to assess ranking robustness under variations in criterion weights. The proposed framework establishes a transparent and adaptable decision-theoretic foundation for context-aware communication protocol selection in heterogeneous M2M scenarios.**

Keywords*—Machine-to-Machine communication; Internet of Things; multi-criteria decision making; AHP; TOPSIS; communication protocol selection; non-functional requirements

I. Introduction

Machine-to-Machine (M2M) communication has evolved from a specialized industrial paradigm into a foundational architectural component of modern cyber-physical systems, smart infrastructures, and large-scale Internet of Things (IoT) deployments. The rapid growth of connected devices, heterogeneous hardware platforms, and distributed data-processing architectures has significantly increased the complexity of communication design decisions. Contemporary M2M environments operate under diverse functional constraints, including energy efficiency, transactional consistency, ultra-low latency, and high-throughput streaming, each imposing distinct non-functional requirements.

The proliferation of communication technologies-ranging from request–response protocols such as HTTP/1.1 and HTTP/2 [1] to broker-oriented systems such as MQTT and AMQP, and high-throughput distributed streaming platforms such as Apache Kafka-has transformed protocol selection into a multi-dimensional engineering problem. Each protocol is optimized for specific design assumptions, transport-layer characteristics, and performance objectives. However, when deployed outside their intended operational domain, these technologies may exhibit significant degradation in latency, reliability, or scalability [2].

The protocol selection problem is further complicated by the conflicting nature of non-functional requirements (NFRs). Minimizing latency may increase resource consumption; maximizing reliability may reduce throughput; and optimizing scalability may introduce architectural complexity. In practice, many M2M deployments rely on protocol benchmarking studies that compare isolated performance metrics without embedding them into a unified analytical framework [3]. As a result, architectural decisions are frequently driven by ad hoc evaluations rather than systematic, reproducible methodologies.

This study addresses this gap by proposing a structured, domain-oriented decision-making methodology grounded in Multiple-Criteria Decision-Making (MCDM) theory. The proposed framework integrates the Analytic Hierarchy Process (AHP) [4] for domain-specific weighting of evaluation criteria with the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [5] for quantitative ranking of alternative technologies.

The present work formalizes a reproducible methodological workflow that can be systematically applied across heterogeneous M2M scenarios. Its primary contribution lies in the definition of a consistent analytical framework in which heterogeneous communication protocols can be evaluated, compared, and ranked according to domain-specific priorities.

While the current study focuses on the conceptual structure, mathematical formulation, and decision-theoretic foundations of the model, its empirical application to concrete deployment scenarios constitutes a complementary line of research that extends the methodology toward domain-specific validation.

The objective is not to identify a universally optimal protocol, but rather to determine the most suitable communication solution for a given functional context. By explicitly linking domain taxonomy, measurable evaluation criteria, and structured multi-criteria analysis, this study provides a reproducible foundation for systematic protocol selection in M2M environments.

The remainder of this study is organized as follows: Section II reviews related work and identifies the main research gaps. Section III introduces the conceptual architecture and the domain taxonomy. Section IV formalizes the evaluation criteria and details the integrated AHP–TOPSIS workflow. Section V discusses the implications of the methodology, including a

sensitivity perspective. Finally, Section VI concludes the study and outlines directions for future research.

The main contributions of this study can be summarized as follows:

- Domain-Oriented Formalization: A structured taxonomy of three representative M2M domains (Sensor Networks, Transactional Systems, and Real-Time Streaming) is defined, explicitly linking functional requirements to measurable non-functional criteria.
- Integrated AHP–TOPSIS Methodology: A reproducible multi-stage computational framework is developed, combining domain-aware subjective weighting (AHP) with objective distance-based ranking (TOPSIS).
- Tail-Latency-Centric Evaluation: The explicit inclusion of 99th percentile latency (L_{p99}) as a primary evaluation criterion addresses a critical yet frequently underrepresented performance dimension in M2M systems.
- Hybrid-Architecture Compatibility: The methodology supports evaluation of both standalone and hybrid communication architectures, reflecting contemporary deployment practices.
- Sensitivity-Aware Decision Framework: A structured sensitivity-analysis perspective is incorporated, enabling evaluation of ranking stability under variations in domain-specific weight configurations.
- Reproducible Analytical Model: The proposed framework establishes a transparent mathematical workflow linking empirical performance metrics to domain-specific decision outcomes.

II. Literature Review and Related Work

The scientific discourse regarding communication protocols in M2M and IoT environments has historically been divided into two separate silos: empirical performance evaluation and theoretical decision science. This section explores the convergence of these fields and identifies the prevailing research gaps.

A. Comparative Analysis of Communication Protocols

Extensive surveys have established a baseline for understanding protocol behavior. Al-Fuqaha et al. [2] provided one of the most comprehensive classifications of IoT technologies, mapping protocols to the OSI model and highlighting the trade-offs between RESTful architectures and broker-based systems. Their work emphasized that the choice of protocol is tightly coupled with the transport layer (TCP vs. UDP), which dictates the underlying reliability and congestion control mechanisms.

Empirical studies, such as those by Thangavel et al. [6], have utilized specialized testbeds to compare MQTT and CoAP. Their findings demonstrated that MQTT's TCP-based nature provides superior reliability in stable networks, whereas CoAP's UDP-based approach offers lower overhead in constrained environments. However, these studies typically focus on a narrow set of criteria, such as latency or throughput, often ignoring the "tail latency" (L_{p99}) or the impact of message ordering violations on transactional integrity.

More recent experimental studies extend these comparisons by incorporating security mechanisms and realistic deployment constraints. Seoane et al. [7] evaluate the performance impact of security layers on CoAP and MQTT in IoT environments, highlighting non-negligible latency overhead under encryption. Similarly, Wytrebowicz et al. [8] provide a pragmatic comparison of messaging protocols that considers not only raw performance metrics but also implementation complexity and deployment characteristics.

B. Architectural Middleware and Data Flow Management

The role of Message-Oriented Middleware (MOM) has been critically examined by Curry [3], who identified persistence, delivery guarantees, and asynchronous decoupling as the three pillars of modern M2M communication. As systems scale, the focus shifts from individual device connectivity to the management of large-scale data streams. Platforms like Apache Kafka have introduced a paradigm shift towards "log-based" communication, which offers high durability but introduces significant complexity in terms of client-side resource consumption.

The literature highlights a clear trade-off: lightweight protocols (MQTT, CoAP) lack the robust data management features of heavy-duty streaming platforms (Kafka, Pulsar). Yet, performance-oriented studies rarely provide a mechanism for translating these qualitative architectural benefits into a structured decision-making framework.

C. Multi-Criteria Decision-Making (MCDM) in IoT

The application of MCDM methods to solve complex engineering trade-offs has gained significant traction. The Analytic Hierarchy Process (AHP), developed by Thomas Saaty, has been used to determine information priorities in resource allocation [4]. In the IoT context, AHP has been successfully applied to select routing protocols and cloud service providers. However, AHP alone is often criticized for its susceptibility to human subjectivity and the "rank reversal" phenomenon when many alternatives are added.

To address these limitations, researchers have explored hybrid models. The integration of AHP with TOPSIS [5] has been identified as a robust solution. Sharifian et al. [9] applied this combination to evaluate reactive routing protocols, while Durão et al. [10] used it for process selection in industrial IoT. These frameworks demonstrate that while AHP captures expert knowledge, TOPSIS provides a mathematically stable ranking based on the Euclidean distance to an "ideal solution" [11].

Recent literature reviews further demonstrate the growing adoption of MADM methodologies in Industry 4.0 decision problems. Zayat et al. [12] synthesize contemporary applications of MADM techniques across industrial systems, confirming their relevance for complex multi-criteria technological selection scenarios similar to M2M communication environments.

Table I summarizes representative studies across protocol surveys, empirical evaluations, and MCDM-based selection approaches, highlighting the absence of dynamic domain weighting and hybrid support in most prior work.

TABLE I. COMPACT SUMMARY OF RELATED WORK CHARACTERISTICS

Reference	Type	Domain Weights	Hybrid
Al-Fuqaha et al. [2]	Survey	No	No
Thangavel et al. [6]	Empirical	No	No
Seoane et al. [7]	Empirical	No	No
Wytrebowicz et al. [8]	Empirical	No	No
Sharifian et al. [9]	MCDM	Partial	No
Durão et al. [10]	MCDM	Static	No
Zayat et al. [12]	Review	N/A	No
Proposed work	MCDM	Yes (Dynamic)	Yes

D. Discussion and Identification of Research Gaps

Despite the richness of the existing literature, two major structural limitations can be identified. First, most MCDM-based approaches in IoT assume static criterion importance, implicitly treating evaluation weights as context-independent. Such an assumption neglects the fact that distinct M2M domains, such as sensor networks and real-time streaming systems, impose fundamentally different non-functional priorities. Second, existing studies typically evaluate communication protocols as mutually exclusive alternatives, without systematically considering hybrid architectural configurations.

The proposed methodology addresses these limitations by introducing a domain-oriented formalization of the decision space. Criterion weights are dynamically derived through AHP in accordance with functional context, thereby embedding domain specificity directly into the ranking process. Furthermore, the integration of tail latency (L_{p99}) as a primary evaluation parameter extends traditional benchmarking approaches that rely predominantly on average metrics. Finally, the formalized AHP–TOPSIS workflow establishes a reproducible mathematical bridge between empirical measurements and domain-aware decision outcomes.

III. DEFINING THE METHODOLOGICAL ARCHITECTURE

Selecting a communication protocol in an M2M environment is inherently a multi-criteria problem. It involves balancing throughput, latency, reliability, scalability, and resource efficiency [5]. The proposed architecture integrates domain-oriented reasoning with quantitative evaluation.

A. Three-Layer Decision-Making Process

The framework in Fig. 1 defines an iterative, three-layer process.

- The Contextual Layer (Domain & Criteria Definition): The process begins by identifying the functional domain and the specific non-functional requirements. This layer defines the “Goal” and the “Criteria” of the AHP hierarchy.
- The Analytical Layer (Evaluation via AHP-TOPSIS): At this stage, weights are calculated using AHP [4] to reflect domain priorities. These weights are then applied to normalized empirical data within the TOPSIS model [9].
- The Strategic Layer (Selection & Hybridization): The output is a closeness coefficient C_i. This value is used to rank technologies or to identify complementary protocols for a hybrid solution.

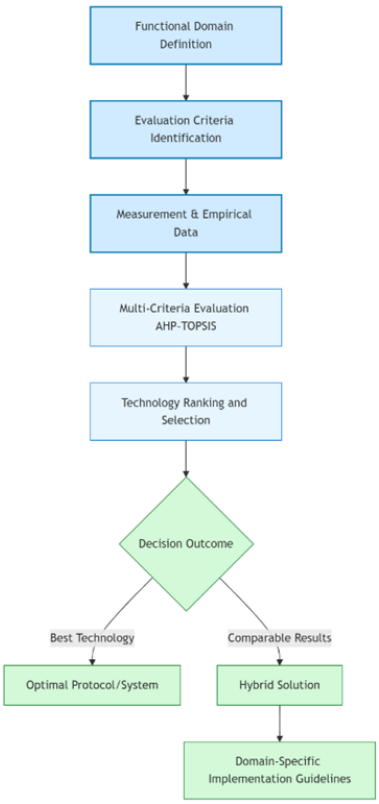


Fig. 1. Conceptual framework for selecting communication protocols and data flow management systems in M2M environments.

B. Core Design Principles

The methodology is guided by four fundamental principles:

- Domain Orientation: Priorities are not fixed but derived from the functional needs of the domain (e.g., Sensor vs. Real-Time).
- Quantitative Measurability: Every criterion must be expressed in a measurable unit (ms, msg/s, $\%$ loss).
- Mathematical Consistency: The use of the Consistency Ratio (CR) in AHP ensures that expert judgments are logical [4].
- Euclidean Distance Optimization: TOPSIS ensures that the chosen technology is not just ”the best” in one category, but the closest to the multidimensional “Ideal Solution” [11].

C. Taxonomy of M2M Domains

To enable dynamic weighting, we classify M2M environments into three domains:

- Sensor Networks: Focus on energy efficiency and aggregate reliability. Latency is secondary.
- Transactional Systems: Focus on strict message ordering, idempotency, and low tail latency to ensure system consistency.

- Real-Time Streaming: Focus on high throughput and minimal L_{p99} latency. Minimal loss is acceptable to maintain "freshness".

IV. Evaluation Criteria and Integrated Methodology

A. Formalization of Evaluation Criteria

A critical aspect of this methodology is the rigorous definition of criteria [13]. The selection process relies on multiple empirically measurable criteria, grouped into two main categories, as in Table II:

TABLE II. Formalized Evaluation Criteria for Protocol Selection.

Criterion	Description	Unit	Type
L_{avg}	Average message transmission latency	ms	Cost
L_{p99}	Latency at 99th percentile	ms	Cost
$Loss$	Share of lost messages	%	Cost
$Ordering$	Message ordering violations	%	Cost
$Reliability$	Successfully delivered messages	%	Benefit
$Throughput$	Channel throughput	msg/s	Benefit
$Scalability$	Behavior with increasing client count	-	Benefit
$Efficiency$	Efficiency in network and computational resource usage	-	Benefit

The relationship between domain and criteria is non-uniform: for instance, in sensor networks L_{avg} and $Reliability$ are prioritized, while in real-time systems L_{p99} and $Tput$ carry higher weights. This dependency is modeled via AHP-derived weights.

We categorize criteria into "Cost" (to be minimized) and "Benefit" (to be maximized), as given in Table III.

TABLE III. Mathematical Definitions of Selection Criteria

Criterion	Mathematical Definition	Type
L_{avg}	$\frac{1}{N}\sum_{i=1}^{N}(t_{recv,i} - t_{send,i})$	Cost
L_{p99}	$P(L \leq x) = 0.99$	Cost
$Loss$	$(1 - \frac{N_{recv}}{N_{sent}}) \times 100$	Cost
$Ordering$	$\frac{N_{out_of_order}}{N_{recv}} \times 100$	Cost
$Reliability$	$\frac{N_{ack}}{N_{sent}} \times 100$	Benefit
$Throughput$	$\frac{Total_Bytes}{Total_Time}$	Benefit
$Scalability$	$\frac{Perf_{N_clients}}{Perf_{1_client}}$	Benefit
$Efficiency$	$\frac{Payload_Size}{Total_Packet_Size}$	Benefit

This categorization ensures that during the normalization phase of TOPSIS, the mathematical direction of each variable is correctly handled [5].

B. The Integrated AHP-TOPSIS Computational Workflow

The core of the proposed methodology lies in the synergy between the subjective prioritization of AHP and the objective, distance-based ranking of TOPSIS. The synergy between AHP and TOPSIS is illustrated in Fig. 2. This dual-layered approach mitigates the inherent weaknesses of using either method in isolation. While AHP can be prone to inconsistencies in human judgment, TOPSIS provides a stable mathematical anchor. Conversely, while TOPSIS requires predefined weights, AHP provides a rigorous mechanism for deriving them based on domain-specific expertise.

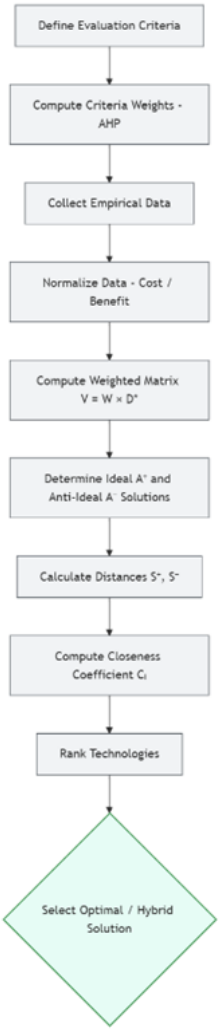


Fig. 2. Integrated AHP–TOPSIS evaluation process for technology selection.

1) Weight determination via the Analytic Hierarchy Process (AHP): The AHP phase begins with the decomposition of the decision problem into a hierarchy. At the summit is the objective (optimal protocol selection), followed by the criteria (latency, throughput, etc.), and finally the alternatives (HTTP/2, Kafka, MQTT).

The fundamental tool of AHP is the pairwise comparison matrix A. For a set of n criteria, the matrix $A = [a_{ij}]$ is a square $n \times n$ matrix where each element represents the relative importance of criterion i compared to criterion j based on the Saaty Fundamental Scale (ranging from 1 to 9) [4]. The matrix is reciprocal, such that $a_{ji} = 1/a_{ij}$ and $a_{ii} = 1$.

To derive the weight vector W, we solve the eigenvalue problem:

$$A \cdot w = \lambda_{max} \cdot w \tag{1}$$

where, λ_{max} is the principal eigenvalue. In practice, the geometric mean method is often used to approximate the weight vector:

$$w_j = \frac{\left(\prod_{j=1}^{n} a_{ij}\right)^{1/n}}{\sum_{i=1}^{n}\left(\prod_{j=1}^{n} a_{ij}\right)^{1/n}} \tag{2}$$

A critical requirement for the validity of AHP is the consistency check [4]. Since human judgment is stochastic and potentially contradictory (e.g., if $A > B$ and $B > C$, a human might mistakenly say $C > A$), we must calculate the Consistency Index (CI):

$$CI = \frac{\lambda_{max} - n}{n - 1} \tag{3}$$

The final validation is the consistency ratio ($CR = CI/RI$), where RI is the Random Index for a matrix of size n. If $CR \leq 0.1$, the weights are considered robust and logically consistent [4]. If not, the pairwise comparisons must be re-evaluated. This rigor is essential in M2M systems where a misaligned weight (e.g., underestimating the importance of tail latency in a transactional domain) could lead to catastrophic system failure.

2) Normalization and the construction of the decision space: Once the weights $W = (w_1, w_2, \ldots, w_n)$ are established, we move to the empirical evaluation of technologies. Let D be the decision matrix containing the raw performance data for m alternatives and n criteria. Because the criteria are measured in heterogeneous units (milliseconds, percentages, messages per second), they cannot be compared directly.

Non-linear normalization is employed [5] to transform all values into a dimensionless space $[0, 1]$. Unlike simple linear scaling, this approach preserves the relative significance of extreme values, which is vital for capturing the performance gaps between protocols like Kafka and HTTP/2.

For benefit criteria (e.g., Throughput, Reliability):

$$r_{ij} = \frac{x_{ij}}{\max_i(x_{ij})} \tag{4}$$

For cost criteria (e.g., Latency, Loss):

$$r_{ij} = \frac{\min_i(x_{ij})}{x_{ij}} \tag{5}$$

This transformation ensures that for all r_{ij}, a value closer to 1 always represents superior performance, regardless of whether the original metric was a "cost" or a "benefit".

3) The TOPSIS ranking algorithm: The Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) is based on the geometric principle that the chosen alternative should have the shortest Euclidean distance from the positive ideal solution (A^+) and the farthest distance from the negative ideal solution (A^-) [11].

First, we construct the weighted normalized decision matrix V:

$$v_{ij} = w_j \cdot r_{ij} \tag{6}$$

Next, we identify the ideal coordinates:

- Positive Ideal Solution (A^+): $A^+ = (v_1^+, v_2^+, \ldots, v_n^+)$, where $v_j^+ = \max_i(v_{ij})$.
- Negative Ideal Solution (A^-): $A^- = (v_1^-, v_2^-, \ldots, v_n^-)$, where $v_j^- = \min_i(v_{ij})$.

The separation measures for each alternative i are calculated using the n-dimensional Euclidean distance:

$$S_i^+ = \sqrt{\sum_{j=1}^{n}(v_{ij} - v_j^+)^2}, \quad S_i^- = \sqrt{\sum_{j=1}^{n}(v_{ij} - v_j^-)^2} \tag{7}$$

Finally, we compute the closeness coefficient (C_i):

$$C_i = \frac{S_i^-}{S_i^+ + S_i^-} \tag{8}$$

The value C_i serves as a universal performance index [5]. An alternative with $C_i = 1$ would represent a hypothetical "perfect" protocol, while $C_i = 0$ would represent the worst possible performer across all criteria.

C. Methodology Summary

For clarity and reproducibility, the complete methodological workflow can be summarized in the following sequential stages:

- Definition of functional domain.
- Identification of measurable evaluation criteria.
- Construction of AHP pairwise comparison matrix.
- Consistency validation of weight configuration.
- Normalization of empirical performance metrics.
- Computation of TOPSIS separation measures.
- Ranking via closeness coefficient C_i.

D. Illustrative Example

To demonstrate the operational logic of the proposed framework, consider a simplified scenario involving three communication alternatives: HTTP/2, MQTT, and Apache Kafka. Let the evaluation criteria be latency (L_{p99}), throughput, and reliability.

Assume that the AHP phase produces the following domain-specific weights:

$$W = (0.4, 0.35, 0.25)$$

After normalization and TOPSIS computation, assume the resulting closeness coefficients are:

$$C_{HTTP/2} = 0.62, \quad C_{MQTT} = 0.71, \quad C_{Kafka} = 0.84$$

In this illustrative scenario, Apache Kafka ranks first due to its superior throughput and reliability characteristics under the assigned domain priorities.

This example does not represent empirical validation, but serves to clarify the computational mechanics of the integrated AHP–TOPSIS workflow.

V. Discussion and Sensitivity Analysis

The results generated by the integrated AHP-TOPSIS methodology provide more than just a simple ranking; they offer a profound insight into the architectural trade-offs inherent in M2M communication. This section delves into the interpretation of the closeness coefficient and the stability of the model.

A. Conceptual Sensitivity Analysis

In multi-criteria decision-making frameworks, ranking stability is inherently dependent on the configuration of criterion weights. Since the AHP phase introduces domain-specific weighting derived from expert judgment, it is necessary to examine the structural robustness of the resulting ranking with respect to controlled perturbations of the weight vector.

Let w_k denote the weight of criterion k. A perturbation Δw_k modifies the weighted normalized matrix V, and consequently affects the separation measures S_i^+ and S_i^-. The perturbed closeness coefficient can be expressed as:

$$C_i' = \frac{S_i^-(\Delta w_k)}{S_i^+(\Delta w_k) + S_i^-(\Delta w_k)}.$$

Conceptually, three stability regimes may be distinguished:

- Dominant-Criterion Regime: Small perturbations in w_k induce ranking shifts, indicating high sensitivity to specific criteria.
- Balanced Regime: Moderate perturbations lead to proportional changes in C_i without rank reversal.
- Robust Regime: The ranking remains invariant under admissible perturbations, demonstrating structural stability of the decision space.

The explicit incorporation of a sensitivity-analysis perspective enhances the methodological credibility of the framework and provides decision-makers with insight into ranking resilience under evolving domain priorities.

B. Architectural Implications of the Closeness Coefficient

A high C_i value suggests that a protocol occupies a "sweet spot" in the multi-criteria space for a specific domain. For instance, in the real-time streaming domain, a protocol like Apache Kafka might achieve a high C_i not necessarily because it has the lowest latency, but because its superior throughput and scalability outweigh its latency costs when weighted through the AHP process.

Conversely, the model often reveals the limitations of "general-purpose" protocols. HTTP/2, while highly efficient for request-response cycles, frequently shows a declining C_i as the number of concurrent M2M connections increases, due to the overhead of its frame-management and the Head-of-Line (HoL) blocking issues at the application layer.

C. The Role of Hybridization in M2M Systems

One of the most significant findings of our methodology is that no single protocol is universally optimal. This supports the move towards hybrid communication architectures. By analyzing the distances S_i^+ and S_i^-, architects can identify protocols that are "specialists". If Protocol A has the best S^+ for reliability and Protocol B has the best S^+ for latency, a hybrid system can be designed where Protocol A handles control signaling and Protocol B handles the raw data payload.

D. Sensitivity Analysis: Testing Model Robustness

In any MCDM model, the final ranking is sensitive to the weights assigned in the AHP phase. To assess robustness, an OAT sensitivity-analysis procedure can be applied by varying the weight of a single criterion (e.g., Latency) by $\pm 20\%$ and examining the resulting changes in the C_i ranking [5]. This procedure can be used to identify stability zones in which the ranking remains invariant and regimes in which rank reversal occurs.

Conceptually, the methodology implies that: 1) Dominant Criteria: In the Transactional domain, the ranking is highly sensitive to the $Ordering$ and $Reliability$ weights. Even a small reduction in these weights can shift the preference from Kafka to a more traditional RDBMS-backed broker. 2) Resilient Criteria: $Throughput$ weights tend to have a more linear and less volatile impact on the final ranking, suggesting that most modern protocols are sufficiently optimized for volume, making other NFRs the true differentiators.

E. Practical Implementation Challenges

Implementing this methodology in a real-world industrial environment requires addressing the stochastic nature of empirical data. Network latency is rarely a constant; it is affected by "micro-bursts", background traffic, and hardware-level interrupts. Therefore, the input data for the decision matrix D should be derived from statistical distributions (e.g., 99th percentile values) rather than simple arithmetic means. Furthermore, the computational overhead of the AHP-TOPSIS process itself is negligible, meaning it can be integrated into "self-adaptive" M2M systems that dynamically switch protocols as network conditions change.

VI. Conclusion and Future Research Directions

This study has presented a domain-oriented, multi-criteria methodology for structured communication protocol selection in Machine-to-Machine (M2M) environments. By integrating the Analytic Hierarchy Process (AHP) with the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), the proposed framework establishes a transparent and reproducible analytical workflow linking domain requirements to measurable performance criteria.

The primary contribution of this study is methodological. Rather than focusing on isolated protocol benchmarks, the framework defines a consistent decision-theoretic space in

which heterogeneous communication technologies can be evaluated according to domain-specific priorities. The explicit incorporation of functional taxonomy, measurable non-functional criteria, and tail-latency–centric evaluation enhances the rigor and interpretability of protocol selection decisions.

The proposed approach is not intended to identify a universally optimal communication protocol. Instead, it formalizes a structured mechanism for determining context-dependent suitability. By supporting dynamic weighting of criteria and accommodating hybrid architectures, the model reflects the practical realities of contemporary M2M deployments.

Limitations: The current study focuses on the conceptual and methodological formalization of the decision workflow. The empirical population of the decision matrix with domain-specific measurements, and the evaluation of ranking outcomes against deployment objectives, are treated as complementary research activities and will be reported separately.

Future research will focus on large-scale empirical validation of the framework across representative deployment scenarios and on extending the methodology toward adaptive decision models capable of responding to dynamic environmental conditions. The integration of uncertainty-aware extensions, such as Fuzzy-AHP and stochastic sensitivity modeling, constitutes a promising direction for further methodological refinement.

REFERENCES

[1] M. Belshe, R. Peon, and M. Thomson, "Hypertext transfer protocol version 2 (http/2)," Internet Engineering Task Force (IETF), RFC RFC 7540, 2015.

[2] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[3] E. Curry, "Message-oriented middleware," in *Middleware for Communications*. Chichester, UK: Wiley, 2004, pp. 1–28.

[4] E. W. L. Cheng and H. Li, "Information priority-setting for better resource allocation using analytic hierarchy process (ahp)," *Information Management & Computer Security*, vol. 9, no. 2, pp. 61–70, 2001.

[5] G.-H. Tzeng and J.-J. Huang, *Multiple Attribute Decision Making: Methods and Applications*. Boca Raton, FL, USA: CRC Press, 2011.

[6] D. Thangavel, X. Ma, A. Valera, H.-P. Tan, and C. K. Tan, "Performance evaluation of mqtt and coap via a common middleware," in *Proceedings of the IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, Singapore, Apr. 2014, pp. 1–8.

[7] V. Seoane and C. Campo, "Performance evaluation of CoAP and MQTT with security support for IoT environments," *Computer Networks*, vol. 197, p. 108338, 2021.

[8] J. Wytrebowicz, K. Cabaj, and J. Krawiec, "Messaging protocols for IoT systems—a pragmatic comparison," *Sensors*, vol. 21, no. 20, p. 6904, 2021.

[9] Z. Sharifian, M. A. Razzaque, and M. S. Hossain, "Loadng-at: A novel practical implementation of hybrid ahp–topsis algorithm in reactive routing protocol for intelligent iot-based networks," *The Journal of Supercomputing*, vol. 78, pp. 9521–9569, 2022.

[10] L. F. C. S. Durão, J. F. P. Fernandes, and P. R. M. Inácio, "Internet of things process selection: Ahp selection method," *International Journal of Advanced Manufacturing Technology*, vol. 99, pp. 2623–2634, 2018.

[11] S. Opricovic and G.-H. Tzeng, "Compromise solution by mcdm methods: A comparative analysis of vikor and topsis," *European Journal of Operational Research*, vol. 156, no. 2, pp. 445–455, 2004.

[12] W. Zayat, H. S. Kilic, A. S. Yalcin, and S. Zaim, "Application of MADM methods in Industry 4.0: A literature review," *Computers & Industrial Engineering*, vol. 177, p. 109075, 2023.

[13] O. Iliev, "Methodology for selecting communication protocols in m2m systems," *Serdica Journal of Computing*, vol. 19, pp. 1–15, 2025.

Generative AI as a Catalyst for Interoperability and Data-Driven Decision Support in Healthcare Systems of Developing Countries

YNSUFU Ali[1], MOSKOLAI NGOSSAHA Justin[2], AYISSI ETEME Adolphe[3], BOWONG TSAKOU Samuel[4]
Faculty of Science-Department of Mathematics and Computer Science,
University of Douala Cameroon, PO Box 24157[1,2,4]
Department of Computer Science, University Institute of Technology
University of Ngaoundéré Cameroon, PO BOX 455[3]

***Abstract*—Interoperability across heterogeneous information systems remains a persistent challenge, particularly in resource constrained contexts where infrastructures are fragmented and data formats remain incompatible. This study introduces a novel methodology that integrates generative Artificial Intelligence (AI) with the urbanization of information systems to enable scalable and seamless interoperability. The approach employs AutoGen AI, an open-source orchestration framework powered by Large Language Models (LLMs), specifically GPT-4o, to coordinate task-specific intelligent agents for data extraction, transformation, and harmonization. By converting disparate data into a standardized JSON representation, the architecture resolves both syntactic and semantic inconsistencies while simultaneously emphasizing multi-agent concurrency, distributed orchestration, and computational scalability, resulting in improved throughput, reduced latency, and enhanced robustness. A real-world healthcare case study is presented to illustrate the framework's effectiveness: heterogeneous clinical datasets were unified into a coherent JSON structure, enabling accurate health indicator generation and reliable decision support. Experimental results demonstrate substantial improvements in system connectivity, processing efficiency, and integration reliability, with potential to generalize far beyond the medical sector. Moreover, the methodology incorporates advanced prompt engineering and context-aware dialogue design, minimizing model hallucinations and ensuring trustworthy outputs in LLM-driven processes. Overall, the study positions generative AI not only as a promising solution for interoperability in health informatics, but also as a transformative paradigm for intelligent system integration across diverse domains characterized by distributed, heterogeneous environments.**

***Keywords*—Decision support systems; interoperability; generative Artificial Intelligence; Large Language Model (LLM); heterogeneous data sources; information system**

I. Introduction

The integration and interoperability of Health Information Systems (HIS) remain pressing challenges, particularly in developing countries where infrastructures are fragmented and data ecosystems are heterogeneous. As healthcare organizations increasingly adopt diverse digital platforms, the demand for seamless system communication and reliable data exchange becomes more urgent. Effective interoperability is not only important for consistent and accurate information sharing, but also for enhancing patient care, optimizing clinical workflows, and strengthening evidence-based decision-making [1]. However, the prevalence of autonomous, distributed, and technologically diverse data sources exacerbates these challenges, often resulting in inconsistent health indicators, organizational misalignments, and limited decision-making capacity [2].

To address these limitations, innovative and flexible approaches are required that can connect fragmented infrastructures while adapting to evolving requirements. In this study, we propose a novel methodology that leverages generative AI, specifically LLMs, to achieve interoperability across distributed medical data sources. Central to this approach is the concept of urbanization of information systems, a strategic framework for modernizing legacy infrastructures incrementally. By reducing the reliance on costly system replacements, this strategy supports the sustainable integration of cutting-edge AI-driven technologies into existing health ecosystems.

The proposed solution is operationalized through AutoGen AI, an open-source orchestration framework for LLM powered applications, implemented via AutoGen Studio and GPT-4o. It employs a network of intelligent agents to collect, transform, and harmonize heterogeneous datasets, thereby constructing a unified and machine-compatible platform. The normalized data is integrated into a central repository, enabling real-time analytics, computationally efficient processing, and reliable decision support.

This study proposes a scalable generative AI-driven interoperability methodology for fragmented healthcare information systems in developing countries, grounded in an urbanization information system strategy and implemented through AutoGen AI and GPT-4o. The contributions include a modular multi-agent harmonization workflow, a canonical JSON standardization layer for KPI-oriented decision support, and a real-world validation using precision, recall, and record-consistency metrics.

This methodology has been validated through a comprehensive real-world healthcare case study, which demonstrates its effectiveness in addressing interoperability challenges at scale. The findings highlight the role of generative AI as a catalyst for dynamic, scalable, and adaptive integration, yielding substantial improvements in data exchange, analytics efficiency, and healthcare service delivery. More broadly, this work underscores the potential of intelligent multi-agent orchestration as a generalizable paradigm for system interoperability in heterogeneous and distributed environments.

The remainder of this study is structured as follows: Section II provides a literature review covering heterogeneous data interoperability, generative AI in healthcare, and urbanization strategies for HIS. Section III details the proposed architectural framework and its technical components. Section IV discusses the results of the case study, presenting validation outcomes and key insights. Finally, Section V concludes by summarizing the contributions and outlining directions for future research.

II. Literature Review

This section presents related studies in the literature on interoperability approaches for heterogeneous data sources, the interoperability of medical data sources, and the urbanization of information systems.

A. Interoperability of Data Sources

With the rise of the Internet and Intranet, the exchange and sharing of information from distributed, autonomous, and heterogeneous data sources have become a fundamental requirement [2]. In such scenario, simultaneous access to multiple sources is often necessary, as they contain relevant and complementary information, emphasizing the importance of interoperability among these systems. Consequently, achieving interoperability and integrating data sources to provide users with a unified access interface is a significant challenge. This challenge encompasses three main aspects: data heterogeneity, source autonomy, and the evolving nature of sources [3].

Integrating disparate information systems requires addressing interoperability issues stemming from various forms of heterogeneity, including hardware platforms, software systems, and especially semantic differences. According to [4], interoperability entails collaboration between applications, specifically referring to the ability of independently developed tools to cooperate and function together. Interoperability means that source A can request and receive services from source B and use their features, thereby establishing interoperability between sources A and B. Data exchange represents a limited form of interoperation, where system A periodically transfers data to system B. Generally, two information systems are considered interoperable if they meet certain criteria: the ability to exchange messages and requests or the ability to work together as a single unit to accomplish a common task [4].

As noted by [5], two information systems are interoperable if: 1) they can exchange messages and requests; 2) they can function as a unit to achieve a shared task. These conditions imply that interoperable systems can leverage each other's functions, act as both clients and servers, and communicate effectively even if their internal components are incompatible. In [4], the authors outline several levels of interoperability:

- The most internal level within the architecture of data management systems, facilitated by specialized gateways programs that establish connections between systems, enabling one system (A) to access data from another (B). Such gateways are already available for many database management systems (DBMS).
- The intermediate level, which involves deploying a multi-database system. This software layer allows users to define their persistent views or perspectives on a set of databases. These systems ensure appropriate connections based on defined views, enabling distributed data access but not managing consistency constraints across different data sources.
- The highest level, where a global system is developed atop existing systems, ensuring overall coherence among them and providing the desired level of integration for a set of data sources.

Interoperability aims to enable effective communication between heterogeneous data models, ensure the integration of functionalities from distinct applications, and support coordination, data dissemination, training, and process validation. However, achieving interoperability is often hindered by three core challenges: conceptual, technological, and organizational barriers. To address these challenges and establish interoperability across multiple data sources, three main strategies can be utilized: the integrated approach, the unified approach, and the federated approach (see Fig. 1).

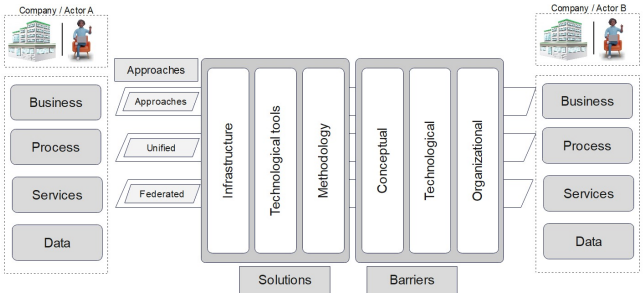


Fig. 1. Interoperability: Levels, barriers, approaches, and solutions.

B. Interoperability and Decision Support

Decision support information systems are a solution to the problems of integration and interoperability of heterogeneous information systems [6]. A decision support system is a truly integrated system that leverages a wide range of tools from various origins, such as specialized software, powerful machines, query tools, DBMS, etc., [7]. It is an information system dedicated to decision-making applications. These applications provide a set of information and tools made available to decision-makers to effectively support the decision-making process. Thus, it is a decision support system aimed at facilitating decision-making based on a wealth of heterogeneous data sources. Implementing such a system is a complex task that involves numerous challenges [8]. Its rigorous design is complicated by the multiplicity of data sources and the so-called complex data that may appear in non-numerical or non-symbolic [9].

A decision support system cannot be built independently of existing systems. Its development process requires a comprehensive vision of operational information systems. Two major phases characterize this process: the information extraction phase and the exploitation phase of the aggregated data from the warehouse.

The first phase involves extracting information from the original data sources, which are often heterogeneous and even distributed. This phase culminates in the construction of a data

warehouse. The integration module addresses the heterogeneity issues of various sources by incorporating them into a global source (GS). This global source is virtual, as the data used for decision-making remains stored in the original data sources and is only extracted during the warehouse update process. The construction module extracts relevant decision-making data and transfers it to the data warehouse while retaining, if necessary, changes in data states. The data warehouse consolidates extracts from source databases that are useful to decision-makers. It serves as a centralized collection of data, a specifically designed database, or a repository for Decision Support Information System (DSIS) applications.

The exploitation phase is handled by database experts. Their role involves restructuring the data in the warehouse to meet the needs of decision-makers. The reorganization module produces data marts. This reorganization aims to efficiently support data query and/or analysis processes. These data marts can be used by online analytical processing (OLAP) applications. To achieve this, the data imported into the marts is organized in a multidimensional manner. Data marts are derived from the warehouse and are tailored to specific analytical needs. The query module manipulates multidimensional data to analyse past trends and support decision-making. The data is represented in forms that facilitate comprehension and manipulation by non-technical decision-makers (tables, charts, maps, etc.).

A viable decision support system requires a specific and scalable data model. Its architecture demonstrates the management of two distinct configurations: the data repository (DW and DM) on the storage side (server), which allows access to all information, and the transactional side (client), which comprises analytical tools that ensure document presentation via graphical interfaces.

C. Generative AI and LLMs for Interoperability

Interoperability across heterogeneous data sources remains a major challenge, especially when information is embedded in unstructured formats or spread across distributed environments. Traditional approaches, ETL pipelines, ontology-based mappings, middleware, and decision support tools, require substantial human effort and often deliver limited scalability. Generative AI, and more specifically LLMs, has emerged as a promising solution to simplify multi-step processes, harmonize diverse inputs, and enable semantic alignment across systems.

Recent literature highlights the growing role of LLMs in data integration. In [10], the authors demonstrate their ability to facilitate entity extraction and resolution from heterogeneous sources by combining automation with human expertise in semi-autonomous pipelines. Similarly, [11] and [12] underline that, despite contextualization and reformulation capabilities, LLMs still struggle with private data access, discrimination between multiple sources, and scalability challenges. In the field of system interoperability, [13] investigates the translation of API calls into natural language.

Schema alignment and ontology matching also benefit from LLMs. In [14], the authors employ LLMs combined with Retrieval-Augmented Generation (RAG) to semi-automate schema harmonization, while [15] introduce Harmonia, an interactive framework integrating LLM reasoning with harmonization primitives. More recently, [16] evaluated LLMs4OM for ontology matching, showing mixed performance compared to traditional methods. In [17], the authors highlight that LLMs can generate Common Data Elements (CDEs) for biomedical research, though with modest interoperability scores. Additionally, [18] present TeaBERT, a knowledge-infused model improving medical entity normalization in multilingual contexts.

In healthcare, several studies demonstrate the effectiveness of LLMs for transforming clinical texts into standardized formats such as HL7 FHIR. In [19], [20], and [21], the authors show that, by using prompts, annotations, clustering, or RAG, LLMs can achieve high accuracy in generating structured FHIR resources, sometimes surpassing traditional NLP. Complementary work by [22] illustrates their potential for transforming laboratory results and mapping diagnostic codes, while [23] show how they can analyze echocardiogram reports to support natural language querying. These applications demonstrate the potential of LLMs to simplify data exchange, improve decision-making, and reduce the need for overly complex standardization.

Beyond clinical texts, [24] and [25] emphasize the importance of common type systems and ontologies to ensure semantic interoperability across biomedical terminologies. In [26], the authors propose retrieval architectures combining NLP with database technologies, while [27] show how ETL, NLP, and semantic web methods can achieve technical, syntactic, and semantic integration across healthcare institutions. Challenges remain in multilingual interoperability and ontology alignment, requiring closer collaboration between NLP and applied ontology [25].

Practical implementations further confirm these insights. In [28], the authors explore FHIR-enabled machine learning systems for improved interoperability and analysis, while [29] proposes a middleware to integrate structured sensor data with unstructured indigenous knowledge for environmental monitoring. Generative AI also shows strong potential in automating workflows and improving patient care [30][31], creating new data forms including text, images, and music [32], and advancing areas such as medical imaging, protein prediction, clinical documentation, and drug design [33][34].

Noteworthy generative models include GPT-4 [35], Claude [36], and PaLM2 [37], which have significantly advanced natural language processing and interoperability applications. Research by [38] shows the widespread adoption of ChatGPT among medical students in Nigeria, underscoring its accessibility and perceived utility. Furthermore, [39] highlight its ability to improve patient understanding and adherence to treatment.

Finally, new AI-driven methods are extending beyond data integration into clinical applications. In [40], the authors propose a privacy-preserving system for abnormal gait detection using encrypted video capture and ML classification. In [41], the authors develop a gait-based analysis model for Parkinson's disease with high accuracy, while [42] survey intelligent models combining NLP, deep learning, and speech recognition for neurological disease interpretation.

Overall, LLMs and generative AI represent a transformative force. They address computational challenges of interoperability in heterogeneous data environments and demonstrate

strong potential in healthcare applications, from clinical documentation to advanced diagnostics. The remaining challenges include prompt dependency, token limits, privacy preservation, and standardization, but evidence consistently shows their ability to bridge gaps between complex informatics infrastructures and practical use of healthcare.

1) Critical synthesis and research gap: Overall, the literature shows that interoperability is commonly addressed through three major directions: i) engineering-based solutions such as ETL pipelines and middleware, ii) semantic approaches relying on ontologies and mapping rules, and iii) decision support architectures built on data warehouses. These methods can be effective in stable and well resourced environments, yet they often require substantial manual effort, rigid schema assumptions, and continuous maintenance, which limits their scalability in rapidly evolving and resource constrained settings. More recently, LLM-based approaches have introduced promising flexibility by enabling semantic reasoning and dynamic transformation; however, they remain sensitive to prompt quality, context limitations, privacy constraints, and the risk of hallucinated outputs. Taken together, these limitations motivate the need for a framework that combines the adaptability of generative AI with a structured modernization strategy, enabling scalable harmonization without requiring disruptive infrastructure replacement.

D. The Urbanization Approach in Health Information Systems

The Urbanization of Health Information Systems is a strategic method designed to enhance the interoperability, agility, and efficiency of health information systems, particularly in developing countries such as Cameroon. According to the World Health Organization (WHO), health information systems must align with public health strategies to ensure their effectiveness. However, the systems currently deployed in contexts like Cameroon are often inadequate, poorly adapted to strategic goals, and suffer from limited interoperability [43]. These systems are fragmented, unstable in the face of organizational and regulatory changes, and difficult to use in collaborative processes [2].

Health systems in developing countries lack coherent digitization. They are characterized by the absence of reliable and accessible data, as well as fragmented and unsecured information systems [44]. These challenges hinder coordination among healthcare institutions, thus impeding the effective implementation of public health policies.

Inspired by urban city planning, the urbanization of information systems aims to modernize existing systems by aligning organizational objectives with technological opportunities while adhering to common rules to ensure quality and consistency [45]. This approach promotes an open, scalable, and adaptable architecture, avoiding costly system overhauls [46].

In the context of health information systems, urbanization entails the development of a national urbanization strategy, supported by general rules for interoperability, accessibility, and security. Such a strategy facilitates information exchange and collaboration among public, private, and non-governmental actors [43]. It also supports the integration of technological innovations such as the Internet of Medical Things (IoMT) and cyber-physical systems [47].

By combining top-down and bottom-up approaches, the urbanization of health systems can improve the quality of care, enhance professional practices, and foster patient engagement in managing their health. Achieving this requires an open architecture, semantic data management, and the integration of decision-support tools [48].

1) Positioning of the proposed approach: The urbanization of information systems offers a well-established strategy for incrementally modernizing healthcare infrastructures, particularly in developing countries. However, existing studies provide limited operational mechanisms for automating semantic harmonization across heterogeneous and autonomous data sources. In this work, generative AI is introduced as the enabling layer that translates urbanization principles into practice through intelligent multi-agent orchestration, thereby connecting strategic modernization goals with concrete interoperability execution.

E. Comprehensive Summary of Key Articles on Interoperability with Generative AI

To make the comparison across existing studies more explicit, Table I synthesizes representative works and highlights not only their contributions, but also recurring limitations, particularly in terms of scalability, semantic consistency, and feasibility in low-resource environments.

The main goal of this study is to design an intelligent tool based on AutoGen AI and GPT-4o, capable of producing reliable outputs with high accuracy and reduced error rates in interoperability driven tasks. Achieving this performance relies on optimized prompts, structured multi-agent workflows, and validation mechanisms designed to mitigate hallucinations and ensure consistent transformations.

III. Proposed Architectural Framework

The methodology presented in this study leverages generative Artificial Intelligence, particularly large language models, to address challenges related to the interoperability and integration of heterogeneous and distributed medical data sources in developing countries.

AutoGen AI, an open-source framework by Microsoft, enables collaborative multi-agent systems using large language models to solve complex tasks. It integrates customizable agents, tools, and human input, supporting scalable deployment and research innovation.

The main goal of this study is to demonstrate how generative AI, specifically through Microsoft's Autogen AI, can be applied to overcome interoperability issues associated with disparate medical data. This approach facilitates the seamless integration of information from diverse sources into a centralized environment while ensuring the consistency and quality of processed data.

The methodology is primarily based on generative AI technologies, notably Autogen AI, structured around an agent-based approach for each phase of data processing. These agents are specifically configured to perform various tasks, ranging

TABLE I. COMPREHENSIVE SUMMARY OF KEY ARTICLES ON INTEROPERABILITY WITH GENERATIVE AI

Citation Key	Description	Limitations
[26]	Proposes architectures that integrate NLP with databases for querying heterogeneous clinical data sources.	Limited adaptability to evolving data formats and semantic inconsistencies.
[25]	Discusses linguistic challenges in biomedical ontologies and emphasizes semantic alignment.	Persistent misconceptions in ontologies and difficulty in integrating multiple terminologies.
[27]	Uses ETL, NLP, and semantic web technologies to improve syntactic and semantic interoperability.	Lack of standard clinical representation and issues with multi-language support.
[28]	Explores machine learning systems leveraging FHIR for clinical interoperability.	Challenges include scalability and lack of robust clinical validation.
[29]	Proposes middleware architecture to unify structured and unstructured environmental data using semantic tools.	Limited scalability and potential domain specificity.
[22]	LLMs used to extract structured insights from unstructured clinical records like lab results and diagnosis codes.	Complexity in mapping across coding systems.
[23]	Uses LLMs to query echocardiogram histories with natural language.	Focused on specific report types.
[10]	Applied LLMs for entity extraction and resolution from heterogeneous sources in higher education; designed semi-autonomous pipelines combining automation and human expertise.	Manual integration remains time-consuming; success highly dependent on prior modeling decisions.
[11]	Drew lessons from data integration research; explored LLMs for contextualization and reformulation in multi-source access.	Difficulty handling private data, discriminating between multiple sources, and achieving scalability.
[19][20]	Used multiple prompts and manual annotations (3,671 clinical text excerpts) to transform clinical texts into FHIR resources with GPT-4, Llama-2-70B, and Falcon-180B; compared with existing NLP tools.	High dependency on prompt design; token limitations; incomplete medical terminologies; risk of hallucinations.
[21]	Employed clustering, prompt engineering, and RAG to transform structured clinical data into HL7 FHIR format.	Difficulty linking complex resources; limited scalability; lack of user-friendly interfaces.
[14]	Applied LLMs and RAG to semi-automate schema alignment in healthcare data integration.	Preliminary research; limited efficiency and scalability.
[15]	Proposed an interactive harmonization system combining LLM reasoning, harmonization primitives, and a user interface.	Process remains complex and time-intensive; further research required.
[16]	Developed an ontology matching framework using RAG + LLM, including confidence filtering and post-processing.	Limited context length; risk of hallucinations; underperformed traditional methods on Bio-ML tasks.
[17]	Leveraged LLMs with human-in-the-loop (HITL) to generate Common Data Elements (CDEs) from biomedical ontologies and corpora.	Low mapping efficiency (32.4%); modest interoperability score (53.8/100); manual revisions still required.

from converting data into uniform formats to managing interoperability within a data warehouse. This structure not only optimizes the flow of information but also ensures efficient data harmonization, thereby enabling in-depth analyses and informed decision-making.

By integrating these advanced technologies, our methodology aims to establish an interoperable data ecosystem that meets the specific needs of healthcare systems in developing countries. It also provides a framework for continuous innovation and the improvement of medical care practices, thereby contributing to enhanced healthcare quality and better patient outcomes.

A. An Approach Based on the Urbanization of Information Systems

The aim of medical data interoperability is to federate data sources for effective decision-making. To achieve this, the Urbanization of Information Systems methodology is adopted, focusing on four levels of concern: business, functional, application, and technical. It balances organizational strategy implementation, system consistency, efficiency, and cost-effective integration of new technologies.

This structured approach involves key steps, including study planning, strategic axis review, current situation analysis, strategy definition, convergence planning, strategy publication, and updates. Applied to medical data interoperability, this methodology results in a framework (Fig. 2) tailored to the integration of distributed and heterogeneous data sources.

The proposed model consolidates consistent medical data into a decision-support-oriented data warehouse. It features: i) Environmentally Open Architecture (Facilitates sharing and integration of distributed data across healthcare stakeholders), ii) Integrated Architecture (Unifies data from autonomous sources into a flexible ecosystem), and iii) Decision-Support Architecture (Provides tools like indicators and statistical analyzis for informed decision-making). This system serves as a monitoring tool, offering valuable insights into medical phenomena while supporting strategic healthcare programs.

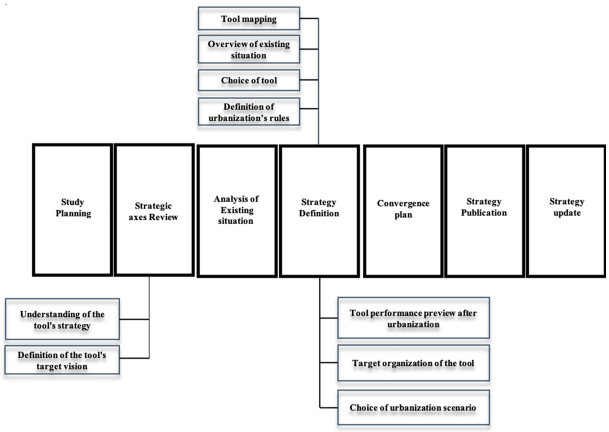


Fig. 2. Implementation of the proposed methodology for information system urbanization, supporting interoperability across heterogeneous data sources.

B. Proposed Architectural Framework

This section addresses the overall architecture of the interoperability methodology for medical data sources, applicable in urban areas of developing countries. Fig. 3 given below provides an overview of our methodological approach.

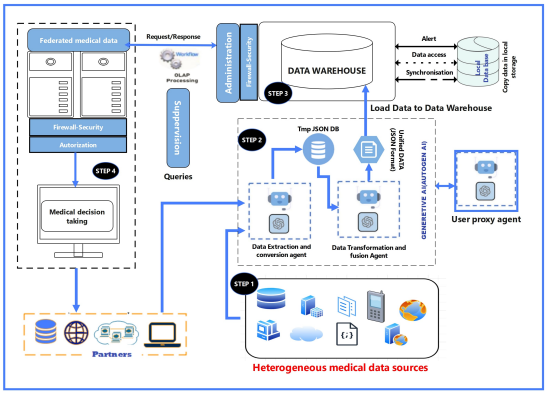


Fig. 3. Interoperability architecture for medical data sources based on the use of generative AI (AutoGen AI framework).

This architecture outlines a complex and structured process to manage the interoperability of heterogeneous medical data, ensuring its transformation, unification, and security with the help of generative AI. The ultimate goal is to create a federated and secure system capable of supporting precise and rapid medical decision-making by using unified and reliable data.

AutoGen AI, an open-source framework developed by Microsoft, streamlines the development of applications leveraging large language models through a multi-agent approach designed to collaboratively tackle complex tasks. These customizable agents, inspired by human teamwork dynamics, integrate LLMs, diverse tools, and human contributions to optimize their performance. AutoGen supports local development of multi-agent systems before deploying them in distributed cloud environments. This innovative platform establishes a synergistic framework where the collective capabilities of agents surpass those of any individual agent, thereby facilitating advanced research and development in artificial intelligence.

The AutoGen AI Framework leverages specialized agents and GPT-4o language models to automate the extraction, transformation, and integration of heterogeneous medical data into a unified platform. These agents perform specific tasks, such as data normalization, semantic mapping, and error correction, while the LLMs ensure contextual understanding and conversion of diverse formats. The system operates through orchestrated workflows with real-time feedback, enabling scalable and efficient data harmonization. This approach ensures compliance with regulatory standards and supports decision-making in healthcare systems.

1) Architectural design rationale: The architecture adopts a multi-agent design because interoperability is inherently a multi-step process that benefits from task specialization (extraction, normalization, semantic alignment, and validation). AutoGen AI was selected as it natively supports agent orchestration, tool integration, and human supervision, which are essential for controlled execution in sensitive healthcare contexts. A canonical JSON representation was chosen as the primary interoperability layer due to its lightweight structure, ease of validation, and direct compatibility with downstream decision-support pipelines. Finally, the data warehouse component ensures long-term consistency and enables systematic KPI computation without disrupting operational source systems, in line with information-system urbanization principles.

a) Step 1. Data extraction and conversion: In this initial step, data is extracted from multiple heterogeneous and distributed medical sources. These sources may include databases, medical records management systems, connected devices, or even local files. The extracted data can be in various formats (e.g., XML, CSV).

The Data Extraction and conversion Agent is responsible for collecting the necessary and relevant data for interoperability. It verifies the formats of the extracted data and converts non-JSON formats into a standardized JSON format. The converted data is stored in a temporary database, TMP JSON DB, for further processing. The conversions are performed without any data loss or alteration.

b) Step 2. Data transformation and fusion: This phase is managed by the Data Transformation and Fusion Agent, which is responsible for data transformation, the handling of heterogeneity issues, and the fusion of multiple JSON files into a single coherent JSON file. The agent reads all JSON files stored in the TMP JSON DB and applies transformation rules provided via the prompt and the tool, including: i) identifying inconsistencies across files (e.g., variations in keys, data types, structures, etc.); ii) resolving these inconsistencies to ensure uniform, consistent, and interoperable outputs; iii) performing any additional operations required by the transformation rules.

Once the transformation is complete, the agent merges all files into a single JSON file, removing redundancies, null values, and non-pertinent data to produce a clear and unambiguous result. Owing to the quality of the preceding transformation, the fusion process is seamless and coherent. The merged file is stored in the Unified Data database and subsequently transferred to the Data Warehouse, which may be performed using a dedicated data-loading tool.

c) Step 3. Storage and administration: The unified data is transferred to a Data Warehouse for centralized and secure storage. Additionally, some data is saved in a local database for faster access or specific processing needs. This step also includes: Data Security (Access to the data is controlled through firewalls and authorization mechanisms); Monitoring (Continuous supervision ensures data integrity); Administration (Administrators manage data, configure access rules, and address user requests);

d) Step 4. Data utilization for decision-making: In this step, federated data is leveraged to support medical decision-making. ETL (Extraction, Transformation, Loading) tools are employed to extract the necessary indicators and convert them into actionable insights for clinical decisions. Health partners can also access the data for analysis or to offer additional services.

AutoGen AI plays a central role in steps 1, 2, and 3

by orchestrating the agents responsible for data extraction, verification, transformation, and unification. Meanwhile, ETL tools take over in step 4, preparing the necessary indicators to facilitate decision-making. ETL tools can also intervene between steps 2 and 3 by assisting in loading unified files into the Data Warehouse.

User Proxy Agent: The User Proxy Agent is pivotal in coordinating and supervising interactions among the various agents within the AutoGen AI system, acting as a human interface within the data interoperability process.

By bridging AI and human expertise, the User Proxy Agent ensures that the AutoGen AI framework remains adaptive, reliable, and aligned with the intricate requirements of medical workflows.

C. Workflow Design for Interoperable Medical Data Architecture

Fig. 4 below provides a simplified representation of the architecture's functionality.

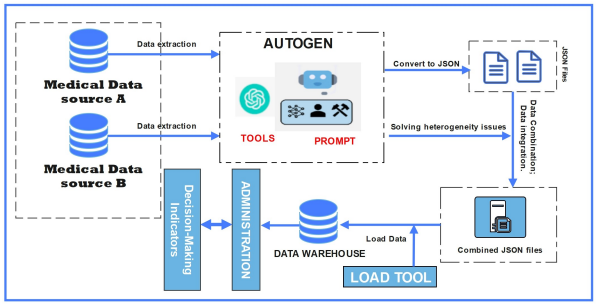


Fig. 4. Workflow of the interoperability architecture for medical data sources based on the use of generative AI (AutoGen AI framework).

1) Conceptual vs. implementation perspective: Conceptually, the proposed framework is organized into four layers: i) heterogeneous medical data sources, ii) AI-driven harmonization and interoperability management, iii) canonical JSON integration and centralized storage, and iv) analytics and decision support services for KPI generation. Implementation wise, these layers are operationalized through AutoGen AI agents, GPT-4o prompting strategies, and dedicated tools for extraction, transformation, validation, and loading. This separation clarifies the methodological principles independently of the technical deployment choices.

This workflow outlines the various steps and processes involved in addressing the challenges of interoperability, integration, and merging of medical data from multiple sources. Its goal is to centralize information and facilitate decision-making. The LLM plays a important role at each key step by using "Tools" (capabilities) and "prompts" (specific instructions) to manage data extraction, transformation, integration, and storage.

a) Tool: A tool represents a specific block of code or script, typically written in Python, that allows AutoGen AI Agents to perform concrete actions or interact with external systems. A tool acts as a "toolkit" for AutoGen AI agents, enabling them to perform complex operations they would not be able to execute without explicit guidance. Each tool is designed to accomplish a specific task and is executed by an agent whenever a prompt triggers its use.

b) Prompt: A prompt is an instruction or request directed to an AutoGen AI agent to perform a specific task. Unlike a tool, which is a piece of code, a prompt is a sentence or query that guides an AutoGen AI agent on what it needs to accomplish. The agent interprets this instruction and, as needed, uses one or more Tools to complete the requested task.

c) Workflow explanation: The workflow encompasses a series of steps designed to address the challenges of interoperability, integration, and unification of medical data from multiple sources. Below is a detailed explanation of the process:

- Data extraction from sources: The Data Extraction and conversion Agent is instrumental in retrieving essential data from medical databases (Data Sources A and B in the diagram) and preparing it for further processing. It begins by connecting to the data sources using predefined Tools to extract only the relevant information needed. To ensure data standardization, the agent analyzes the format of the extracted data, retaining it as JSON if already compliant or converting it to JSON when necessary, thus creating a uniform structure for subsequent stages. Finally, the prepared JSON files are systematically stored in a dedicated repository called the JSON File DB.
- Management of heterogeneities: The Data transformation and fusion Agent processes raw JSON files from the JSON File DB, addressing differences in data formats to ensure interoperability. The process begins with reading the JSON files from the repository, followed by schema standardization to ensure data consistency. This involves unifying key names and types, maintaining data type consistency for values, and resolving semantic variations and structural discrepancies. Once standardized, the are ready to be merged. The rules, requirements, and constraints of the transformation and merging are provided by the tool called **Transform**.
- Combination or merging of prepared JSON files: The Data transformation and fusion Agent consolidates all prepared JSON files into a single, coherent dataset, ready for storage in the Data Warehouse. The agent merges the JSON files into a unified dataset by eliminating duplicates to enhance data clarity and efficiency, processing or removing fields with null values to prevent inconsistencies, and applying a homogeneous schema to create a consolidated file.
- Storage in the data warehouse: The finalized dataset is prepared for long-term storage and use. The merged JSON file is stored in a centralized Data Warehouse, which serves as a repository for federated data and provides an integrated, ready-to-use version of the data suitable for decision-making, analysis, or reporting.
- Administration and decision support indicators: Structured in two complementary components: data gov-

ernance and security, and analytical decision support. The governance layer establishes strict access controls and privacy safeguards within the Data Warehouse, ensuring that only authorized users handle sensitive information while upholding confidentiality and data integrity, which is important in healthcare. Decision support indicators are derived through systematic analysis of stored data, enabling the identification of health trends, forecasting of treatment demands, and generation of actionable recommendations. Together, these processes empower healthcare professionals and administrators to enhance care quality, allocate resources efficiently, and strengthen data-driven strategic planning.

The workflow ensures the seamless integration, transformation, and utilization of medical data, delivering a robust and secure system for healthcare decision-making. Interoperability management is achieved through the coordination of two AutoGen AI agents: The Data Extraction and conversion Agent is responsible for retrieving raw data from various sources and conversion of each data in JSON file. The Data transformation and fusion Agent standardizes, harmonizes diverse datasets, combines and consolidates data into a unified format. All these agents are orchestrated by the User Proxy to ensure efficiency and consistency across the workflow.

IV. Results and Discussion

In order to validate the proposed methodology, this section describes the different case studies and the results obtained.

A. Context of the Case Study

To test and validate our interoperability model, we used three heterogeneous medical data sources from three Central African countries. These data, sourced from MSSQLSERVER (Cameroon), MySQL (DR Congo), and Excel (Gabon), contain information on diseases diagnosed over a year, prescribed treatments, and patient progress.

1) Reproducibility and data availability: The datasets used in this case study correspond to real clinical records and cannot be publicly released due to privacy and institutional restrictions. To support experimental reproducibility, we provide the complete data schema, record volumes per source, field-level statistics (missingness and value distributions), and anonymized examples illustrating structural and semantic heterogeneity. We also document the harmonization rules (key mapping, normalization, deduplication, and validation) and describe a synthetic data generation procedure that preserves the original schema characteristics, enabling independent replication of the end to end workflow under equivalent conditions. Table II presents dataset summary for the case study.

TABLE II. Dataset Summary for the Case Study

Dataset	Country	Format / DB	Records
DS_1	Cameroon	SQL Server (DIAG table)	2000
DS_2	DR Congo	MySQL (detect table)	1997
DS_3	Gabon	Excel (gabon_patients.xlsx)	1978
Unified dataset (canonical JSON)			5975

Table III presents the various fields from each data source.

Interoperability management was entrusted to two AutoGen Studio agents using the GPT-4-o model, yielding promising results.

2) Data extraction and verification: The first agent, responsible for extraction, verification and conversion, retrieved data from the three sources and converted them into JSON files.

The Data Extraction and conversion Agent relies on specific tools, including connection parameters for databases, directories where extracted data is stored, retrieving data from external sources.

To optimize the agent's guidance and ensure its proper functioning, four dedicated tools are provided. The tool *sqlservercon* defines the connection parameters, enables data extraction from the SQL Server database, and performs task verification; its implementation script is available on GitHub (not yet deployed for public use). Similarly, the tool *mysqlcon* establishes the connection parameters, supports data retrieval from the MySQL database, and ensures task validation, with its corresponding implementation script also accessible on GitHub (not yet deployed for public use). Finally, the tool *excelcon* specifies the connection settings, facilitates data extraction from Excel files, and provides task verification, with the related implementation script likewise available on GitHub (not yet deployed for public use).

The data extraction and conversion prompt providing instructions to the agent is as follows:

```
You are a data extraction agent. Your mission: read raw data
↪  from three sources and write them to JSON with no
↪  alteration whatsoever.
AVAILABLE TOOLS
sqlservercon: SQL Server (table DIAG) → produces ca.json
mysqlcon: MySQL (table detect) → produces rdc.json
excelcon: Local Excel (path handled by the tool) → produces
↪  ga.json
Common output folder: D:\AUTOGEN\EXTRACTFILES
OBJECTIVES (for EACH source)
Read 100% of the data.
Convert to standard JSON, strictly identical to the source
↪  values.
Write to D:\AUTOGEN\EXTRACTFILES, overwriting existing
↪  files.
Enforced file names: ca.json (SQL Server), rdc.json (MySQL),
↪  ga.json (Excel).
CONSTRAINTS
No processing/normalization/sorting/typing. Zero alteration.
JSON must be valid and parsable.
VERIFICATION / EVALUATION
Verify the existence of all three generated files.
Check integrity: source_count == json_count (and values
↪  unmodified).
Produce small audit charts/diagrams (without modifying the
↪  data).
Produce Acccuracy, precision, lost
EARLY STOP CONDITION
If all three tasks are completed successfully and integrity
↪  is confirmed for each source, then produce a concise
↪  final report (status, file paths, integrity metrics) and
↪  stop immediately with no further actions.
EXPECTED OUTPUTS
Files: D:\AUTOGEN\EXTRACTFILES\ca.json, rdc.json, ga.json
Final report: for each source → success/failure, file path,
↪  source_count, json_count, integrity_ok,
↪  warnings/anomalies. Produce Acccuracy, precision, lost
```

This prompt defines a data extraction agent that retrieves raw data from SQL Server, MySQL, and Excel, exporting them into JSON files with no alteration. It enforces strict file naming, location, and integrity checks to guarantee one-to-one correspondence between source and output. Audit metrics

TABLE III. VARIOUS FIELDS FROM EACH DATA SOURCE

COUNTRY	CAMEROON (SQL Server)	DR CONGO (MySQL)	GABON (Excel)
FIELD	• PATIENT_ID • COUNTRY • AGE • DIAGNOSIS • TREATMENT • CONSULTATION_YEAR • STATUS	• p_code • Country_code • Year_of_Birth • Disease • Medication • Year_of_Diagnosis • Outcome	• PatientID • Country • Date_of_birth • Condition • Medication • VisitYear • Status

(accuracy, precision, loss) and charts ensure transparency of the process. Finally, a concise report validates completeness and correctness before termination.

3) Data transformation and fusion: The second agent(Data Transformation and Fusion Agent), specializing in data transformation, the handling of heterogeneity issues, and the fusion of multiple JSON files into a single coherent JSON file. To accomplish this task, it relies on the following elements:

a) Tool called Transform: This tool enables the agent to manage file reading and access paths, normalize fields and values, address heterogeneity issues, compute derived attributes (such as age), enforce data quality controls, apply processing and merging constraints, and generate detailed execution reports.

The implementation code of the **Transform** tool is available on github (not yet deployed for public use).

After the transformations, the various files are merged into a single structured file. The text below presents the prompt for the transformation and unification agent:

```
ROLE
Unify three JSON sources into one standardized file using
↪  the tool "transform" only.
TOOL
- Use "transform" for ALL steps (load, map, compute AGE,
↪  dedupe, normalize, write, report). Do not implement
↪  logic outside the tool.
INPUTS
- D:\AUTOGEN\EXTRACTFILES\ca.json   (SQL Server)
- D:\AUTOGEN\EXTRACTFILES\rdc.json  (MySQL)
- D:\AUTOGEN\EXTRACTFILES\ga.json   (Excel; may be {Sheet:
↪  [rows]}; concatenate all rows)
TARGET SCHEMA
IDPATIENT (string), COUNTRY (string), AGE (int), DIAGNOSIS
↪  (string), TREATMENT (string), CONSULTATION_YEAR (int),
↪  STATUS (string)
KEY MAPPING (case/underscore/space-insensitive)
IDPATIENT ← p_code, PATIENT_ID, PatientID, id, ID, IdPatient
COUNTRY ← Country_code, COUNTRY, Country
AGE ← AGE
DIAGNOSIS ← Disease, Condition
TREATMENT ← Medication, drug
CONSULTATION_YEAR ← VisitYear, Year_of_Diagnosis,
↪  YearOfDiagnosis
STATUS ← Outcome, Status
Birth fields for AGE: Year_of_Birth, Date_of_birth (and
↪  variants)
AGE RULES
- If AGE missing/invalid and birth year/date available:
  - If CONSULTATION_YEAR present: AGE = CONSULTATION_YEAR -
  ↪  Year_of_Birth (or year(Date_of_birth))
  - Else: AGE = (current year) - Year_of_Birth
- If result <0 or >120 → AGE = null
LIGHT NORMALIZATION
- Cast AGE, CONSULTATION_YEAR to int when possible; else
↪  null
- Trim spaces for COUNTRY/DIAGNOSIS/TREATMENT/STATUS (do not
↪  change case)
- COUNTRY codes → canonical: {CD,RDC→"DR Congo";
↪  CM,CMR→"Cameroon"; GA,GAB→"Gabon"}; otherwise keep as-is
- Do not rewrite DIAGNOSIS/TREATMENT/STATUS semantically
DEDUPLICATION
- Composite key: (IDPATIENT, COUNTRY, DIAGNOSIS,
↪  CONSULTATION_YEAR)
- Keep the most complete record (most non-null fields);
↪  count conflicts/removed duplicates
OUTPUT
- Write: D:\AUTOGEN\EXTRACTFILES\unifiedMedicalData.json
↪  (overwrite if exists)
VERIFY & REPORT
- Report: N_ca, N_rdc, N_ga, N_in (sum), N_out, duplicates
↪  removed, conflicts resolved
- % rows with computed AGE; simple COUNTRY distribution
- Confirm JSON written and parsable
Produce small audit charts/diagrams (without modifying the
↪  data).
Produce Acccuracy, precision, lost
EARLY STOP
- On success (file written + checks done): output concise
↪  final report and stop immediately.
```

This prompt defines a transformation workflow to merge three heterogeneous JSON sources into a single standardized schema. All operations mapping, age computation, normalization, deduplication, and reporting are delegated to the tool transform. It ensures data consistency, enforces validation rules, and resolves duplicates transparently. Final output is a unified JSON file plus integrity metrics and audit charts, ensuring reliability and reproducibility.

B. Results

The case study highlights the effectiveness of generative Artificial Intelligence(the use of AutoGen Studio and the GPT-4o model) in enhancing interoperability across distributed and heterogeneous healthcare systems. The proposed approach is built upon an architecture of intelligent agents capable of autonomously extracting, transforming, and harmonizing medical data from multiple sources, while accounting for the specific constraints of clinical environments. This is achieved through a strategy based on the urbanization of information systems, which enables rapid and flexible integration without requiring substantial restructuring of existing infrastructures. The process demonstrates a very low error rate and strong performance in interpreting medical data. Hallucination risks were substantially reduced through constrained prompting, tool-based execution, and systematic validation, even when processing noisy or non-standardized records, thereby ensuring reliable normalization and consistent outputs for both information systems and end users. Moreover, unified access to harmonized data improves real-time medical decision-making, while the modular architecture supports scalability and adaptability across diverse clinical settings.

Step by step, the obtained results are as follows:

1) Data extraction and conversion: This section produced the following results:

a) Successful extraction: Data were successfully extracted and converted from multiple sources without errors or data loss. The DIAG table from MSSQLSERVER and the detect table from MySQL were both converted without issues, while the Excel file gabon medical gabon_patients.xlsx was transformed into JSON with data integrity fully preserved.

b) Verification and encoding management: All JSON files were validated and encoded in UTF-8, with any potential encoding issues addressed by employing alternative character sets such as Latin1, ISO-8859-1, and Windows-1252.

c) Overall results: No data was lost or altered during extraction, and a "SUCCESS" message appeared for each completed task. The same number of rows for all three sources when comparing input and output. The same number of columns for all three sources when comparing input and output. No data loss for any of the three sources, with 97% to 99% of accuracy.

Table IV given below presents a comparative summary of record counts from the external sources (SQL Server, MySQL, Excel) and from the JSON files extracted and converted by the agent.

According to the table, all input records match exactly with the output records in terms of the number of rows. Accuracy and Precision prove that Data was extracted and converted without loss or alteration.

The diagram below (Fig. 5) illustrates the success of the data extraction and conversion process.

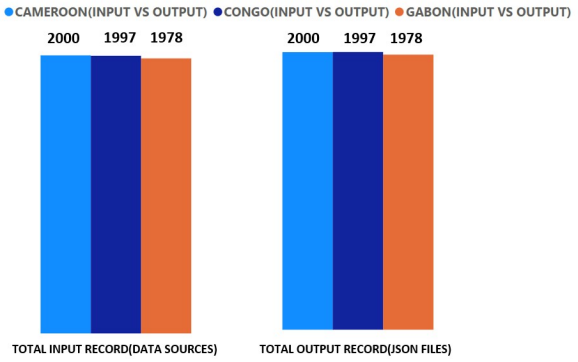


Fig. 5. This figure shows that all records are matched for each country. The correspondence is perfect, as the values for INPUT RECORDS are equal to the values for OUTPUT RECORDS.

2) Data verification and fusion:

a) The result is as follows: The unified JSON file called **unifiedMedicalData.json** was generated correctly with no errors or ambiguities as a direct result of rigorous transformation and careful resolution of data heterogeneity as follows.

b) Key standardization: A standard key-value mapping (**KEY_MAPPING**) was applied to harmonize field names across the three JSON inputs. Heterogeneous fields (e.g., Year_of_Birth, Date_of_birth) were consolidated under the single canonical key **AGE**; diagnostic fields (Disease, Condition) under **DIAGNOSIS**; and visit fields (VisitYear, Year_of_Diagnosis) under **CONSULTATION_YEAR**. Country codes were normalized to canonical labels (e.g., CM/CMR → Cameroon, CD/RDC → DR Congo, GA/GAB → Gabon).

Problematic files with non-UTF8 encodings were automatically re-decoded and validated.

Numeric casts were enforced for **AGE** and **CONSULTATION_YEAR** when possible. Leading and trailing spaces were trimmed from **COUNTRY**, **DIAGNOSIS**, **TREATMENT**, and **STATUS**, without semantic modification.

All three sources were successfully unified into a single standardized JSON with uniform keys and validated numeric types. Encoding issues were auto-corrected. Deduplication preserved all unique information while eliminating redundant entries.

c) Overall outcome: No information was lost, and the resulting dataset offers a homogeneous, coherent structure ready for use in decision-support systems.

Table V below summarizes the results of the transformation and unification agent.

The table shows a very good accuracy, precision, and recall of 0,97%, confirming that all data from the various files are accurately reflected in the unified file.

The table confirms that all heterogeneity issues were successfully resolved, and all transformed files maintain the same key names and types. The identified common keys are: **IDPATIENT**, **COUNTRY**, **AGE**, **DIAGNOSIS**, **TREATMENT**, **CONSULTATION_YEAR**, **STATUS**.

The diagram below (Fig. 6) illustrates the success of the data transformation and fusion process. A bar chart of the final result is given.

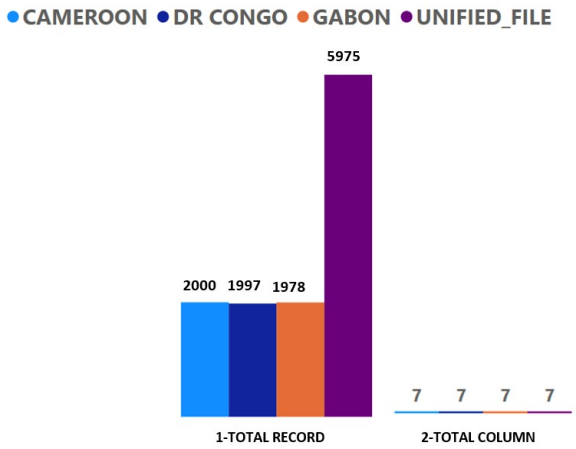


Fig. 6. Final result comparison diagram.

The unified file consolidates all records from the three countries (2000 + 1997 + 1978 = 5975), confirming that the data merging process, in terms of the number of rows, is a success.

All entities (Cameroon, DR Congo, Gabon, and Unified_File) have the same number of columns (7). This indicates

TABLE IV. DATA EXTRACTION AND CONVERSION RESULTS TABLE

MEASURES	CAMEROON(SQL SERVER)	DR CONGO(MYSQL)	GABON(EXCEL)
1-TOTAL RECORD(SOURCES)	2000	1997	1978
2-TOTAL RECORD(JSON FILES)	2000	1997	1978
3-ACCURACY	0,97	0,97	0,98
4-PRECISION	0,98	0,98	0,99

TABLE V. THE RESULTS OF THE UNIFICATION AGENT

MEASURES	CAMEROON	DR CONGO	GABON	UNIFIED_FILE
1-TOTAL RECORD	2000	1997	1978	5975
2-TOTAL COLUMN (KEYS)	7	7	7	7,00
3-ACCURACY	0,96	0,96	0,98	0,97
4-PRECISION	0,98	0,98	0,99	0,97
5-RECALL MATCHED	0,98	0,98	0,99	0,97

that the merging of the various files was also successfully executed in terms of column consistency.

This chart demonstrates the successful integration of individual data into a consolidated file (Unified_File).

The results are consistent with the defined methodology.

- Sensured complete extraction of heterogeneous medical data (Cameroon, DR Congo, Gabon) without alteration.
- Verified integrity, showing identical record counts between sources and JSON outputs.
- Unified the datasets into a standardized schema, producing 5,975 records across 7 columns. Normalization and deduplication rules were respected, preserving data quality and consistency.
- Integrated the unified file into the data warehouse, enabling OLAP analysis and medical decision support.

The proposed framework primarily standardizes heterogeneous inputs into a canonical JSON representation, which serves as the core interoperability layer for unified integration. This standardized JSON is directly consumed by a decision-support system to generate reliable healthcare KPIs and enable evidence-based analytics. When required, an additional mapping layer can align the canonical JSON with widely adopted standards such as HL7 FHIR (e.g., Patient, Observation, Condition, Encounter), ensuring compatibility with external clinical systems without constraining the harmonization process.

C. Discussion and Future Directions

The proposed methodology, which integrates generative AI through the AutoGen AI framework and GPT-4o, represents a significant advancement in addressing the challenge of interoperability in health information systems. The findings highlight the feasibility of harmonizing data from heterogeneous sources, such as SQL Server, MySQL, and Excel, into a unified and interoperable platform. The results hold theoretical importance as they demonstrate how modular, multi-agent architectures can address both syntactic and structural inconsistencies in medical data. For researchers in other countries, these findings emphasize the adaptability of generative AI to modernize fragmented health infrastructures, offering valuable insights into achieving scalable and sustainable interoperability. Lessons learned include the critical importance of modularity and the integration of advanced AI to optimize workflows and decision-making.

The study's results validate the initial hypotheses by confirming that generative AI, when combined with a modular multi-agent system, can achieve seamless interoperability. The high precision and recall rates achieved underscore the robustness of the methodology. The deployment of specialized agents and the capabilities of GPT-4o ensure efficient workflow optimization and reliable data harmonization. This supports the study's objectives and demonstrates the value of an AI-driven approach in overcoming the fragmentation challenges inherent in many healthcare systems, particularly in resource-constrained settings.

1) Biases and generalizability: Although the results are encouraging, they may be influenced by dataset-specific factors such as local coding practices, missing values, and differences in clinical documentation across countries. The proposed framework is expected to generalize to other healthcare environments as long as a minimal overlap of clinical concepts and metadata is available. However, performance may degrade in settings with highly ambiguous schemas, inconsistent terminologies, or extremely noisy records, highlighting the need for stronger semantic validation and domain-specific constraints.

External comparisons reveal the methodology's novelty and its advantages over existing approaches. In many developed countries, interoperability solutions often rely on centralized systems that necessitate costly infrastructure overhauls. In contrast, the modular and incremental integration demonstrated in this study proves more adaptable to resource-constrained environments. Previous studies have identified issues with semantic ambiguities and scalability in AI-driven interoperability frameworks. This methodology effectively addresses those challenges, positioning it as a superior alternative. Differences in results across studies are often attributed to variations in infrastructure and data availability, further underscoring the flexibility of this approach to diverse healthcare settings.

2) Comparison with existing approaches: Compared to classical ETL- or middleware-driven interoperability solutions, the proposed methodology reduces manual schema engineering by delegating normalization and semantic consolidation to LLM-orchestrated agents. In contrast to purely ontology-based mappings, which often require extensive expert maintenance, the multi-agent workflow enables adaptive harmonization under evolving schemas. Relative to recent LLM-to-FHIR transformation studies, our approach differs by emphasizing a canonical JSON interoperability layer designed for direct KPI-oriented decision support, while keeping FHIR alignment as an optional compatibility extension.

The study's contributions are both conceptual and methodological, representing a significant leap forward in the field of health informatics. Conceptually, it introduces the urbanization of HIS as a progressive strategy for modernization. This approach emphasizes gradual, sustainable improvements rather than disruptive overhauls, offering a scalable model for developing healthcare systems. Methodologically, the integration of generative AI with a multi-agent framework offers a novel and practical approach to addressing data interoperability. While generative AI has been explored in various domains, this study is among the first to demonstrate its potential for harmonizing diverse medical data, presenting a new perspective for health informatics research.

Security risks associated with LLM-driven code generation and execution over sensitive medical databases must be explicitly addressed. In our framework, the LLM is not granted direct execution privileges on production systems. Instead, all generated queries and transformation scripts are treated as untrusted outputs and are routed through a policy-enforcement layer that validates requests, restricts operations to an allow list, and blocks destructive commands. Execution is performed in an isolated environment using least-privilege service accounts, thereby limiting potential damage and preventing uncontrolled access to personally identifiable information. In addition, prompt-injection threats originating from untrusted clinical text are mitigated through strict data–instruction separation and input sanitization. Finally, all interactions are logged to support traceability, auditing, and compliance, while future work will investigate privacy-preserving extensions such as federated learning and stronger governance mechanisms.

Despite its strengths, the study is not without limitations. The reliance on high-quality prompts for GPT-4o operations introduces variability, which could be addressed through the development of standardized prompt templates. Additionally, concerns about data privacy and security, especially when processing sensitive medical information in cloud environments, remain a significant challenge. Implementing encryption, data anonymization, and on-premises or federated learning models could mitigate these risks. Furthermore, the need for human oversight to manage semantic ambiguities suggests a need for future research into domain-specific contextual dictionaries and training modules. Testing the framework in diverse healthcare environments would also help identify potential gaps in its adaptability.

Building on these findings, several avenues for future research emerge. Integrating Federated Learning (FL) could enhance privacy and security while maintaining interoperability by enabling collaborative model training without direct data sharing. Additionally, incorporating adaptive learning mechanisms into the AI framework could improve its ability to autonomously handle semantic inconsistencies. Cross-country validations would further establish the framework's global applicability and scalability. Lastly, the development of international ethical guidelines for the use of generative AI in healthcare would ensure equitable and responsible implementation. These future directions provide a pathway for refining and expanding this methodology, making it a critical tool for advancing global health information systems.

V. Conclusion

This study addressed the challenges of interoperability in heterogeneous information systems, with a particular focus on resource constrained healthcare environments. By combining generative Artificial Intelligence, notably GPT-4o, with the AutoGen AI orchestration framework, the proposed methodology demonstrated that a modular multi-agent workflow can effectively harmonize distributed and incompatible medical data sources into a unified, machine compatible representation.

The experimental findings support the feasibility of the approach and confirm that AI-driven harmonization can reduce fragmentation while improving integration reliability and analytics readiness. Importantly, embedding the methodology within an information-system urbanization strategy strengthens its practical relevance for developing regions by enabling incremental modernization without disruptive infrastructure replacement.

Nevertheless, the validity of the proposed framework depends on several conditions. Its performance is strongest when heterogeneous sources share minimal overlapping clinical concepts, when basic metadata is available, and when transformation rules can be constrained through well-defined prompts and validation mechanisms. Conversely, highly ambiguous schemas, inconsistent terminologies, and extreme noise in clinical records may reduce alignment quality and increase the need for human oversight. These limitations highlight the importance of systematic validation, governance, and careful deployment in sensitive environments.

Future work will focus on improving reproducibility through the release of synthetic datasets and structured schemas, strengthening privacy guarantees through federated and privacy-preserving mechanisms, and extending the methodology to broader domains characterized by heterogeneous and distributed data ecosystems. Establishing ethical and governance guidelines will also remain essential for responsible adoption.

Finally, in adherence to ethical guidelines and to protect patient confidentiality, all datasets used in this study were anonymized prior to analysis.

Acknowledgment

We address our thanks to the ISDM Research Team of Faculty of Science of the University of Douala.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability Statement

The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Conflict of Interest Disclosure

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this study.

Ethics Approval Statement

This study did not involve human participants or animals requiring ethical approval. All procedures were conducted in accordance with institutional and international research guidelines.

References

[1] D. Chen, G. Doumeingts, and F. Vernadat, "Architectures for enterprise integration and interoperability: Past, present and future," *Computers in industry*, vol. 59, no. 7, pp. 647–659, 2008.

[2] A. A. Eteme and J. M. Ngossaha, "The contribution of icts to sustainable urbanization and health in urban areas in cameroon," in *Waste Management: Concepts, Methodologies, Tools, and Applications*. IGI Global Scientific Publishing, 2020, pp. 624–641.

[3] L. Bellatreche, N. X. Dung, G. Pierra, and D. Hondjack, "Contribution of ontology-based data modeling to automatic integration of electronic catalogues within engineering databases," *Computers in Industry*, vol. 57, no. 8-9, pp. 711–724, 2006.

[4] N. Ploquin, "Etude des différents modes d'interopérabilité en génie logiciel," Ph.D. dissertation, Master Thesis, University of Nantes, 2001.

[5] G. V. Solar and A. Doucet, "Médiation de données: solutions et problèmes ouverts," *Actes des 2ème assises nationales du GDR I*, vol. 3, no. 11, 2002.

[6] P. Vidal and P. Planeix, "avec lacroux, f., augier, m. et lecoeur, a.«," *Systèmes d'information organisationnels*, 2005.

[7] A. Ayissi Eteme, "Conception et développement d'une plateforme de gestion des données hétérogènes et incomplètes pour la prise de décision en milieu urbain au cameroun," Ph.D. dissertation, Thèse de Doctorat/PhD). ENSP, Université de Yaoundé I, 2007.

[8] O. Teste, "Modélisation et manipulation d'entrepôts de données complexes et historisées," Ph.D. dissertation, Université Paul Sabatier-Toulouse III, 2000.

[9] J. Darmont, O. Boussaid, J.-C. Ralaivao, and K. Aouiche, "An architecture framework for complex data warehouses," *arXiv preprint arXiv:0707.1534*, 2007.

[10] A. Remadi, K. El Hage, Y. Hobeika, and F. Bugiotti, "To prompt or not to prompt: Navigating the use of large language models for integrating and modeling heterogeneous data," *Data & Knowledge Engineering*, vol. 152, p. 102313, 2024.

[11] A. Halevy and J. Dwivedi-Yu, "Learnings from data integration for augmented language models," *arXiv preprint arXiv:2304.04576*, 2023.

[12] J. Freire, G. Fan, B. Feuer, C. Koutras, Y. Liu, E. Peña, A. S. Santos, C. T. Silva, and E. Wu, "Large language models for data discovery and integration: Challenges and opportunities." *IEEE Data Eng. Bull.*, vol. 49, no. 1, pp. 3–31, 2025.

[13] R. Lehmann, "Towards interoperability of apis-an llm-based approach," in *Proceedings of the 25th International Middleware Conference: Demos, Posters and Doctoral Symposium*, 2024, pp. 29–30.

[14] R. Saripalle, R. Foulger, and S. Dooda, "Leveraging llms and rag for schema alignment: A case study in healthcare," *Proceedings of the 27th International Conference on Enterprise Information Systems (ICEIS)*, 2025.

[15] A. Santos, E. H. Pena, R. Lopez, and J. Freire, "Interactive data harmonization with llm agents: Opportunities and challenges," *arXiv preprint arXiv:2502.07132*, 2025.

[16] H. B. Giglou, J. D'Souza, F. Engel, and S. Auer, "Llms4om: Matching ontologies with large language models," *arXiv preprint arXiv:2404.10317*, 2024.

[17] R. A. Long, S. Ballard, S. Shah, O. Bianchi, L. Jones, M. J. Koretsky, N. Kuznetsov, E. Marsan, B. Jen, P. Chiang *et al.*, "A new ai-assisted data standard accelerates interoperability in biomedical research," *medRxiv*, 2024.

[18] L. Chen, Y. Qi, A. Wu, L. Deng, and T. Jiang, "Teabert: An efficient knowledge infused cross-lingual language model for mapping chinese medical entities to the unified medical language system," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, pp. 6029–6038, 2023.

[19] Y. Li, H. Wang, H. Yerebakan, Y. Shinagawa, and Y. Luo, "Enhancing health data interoperability with large language models: a fhir study," *arXiv preprint arXiv:2310.12989*, 2023.

[20] Y. Li, H. Wang, H. Z. Yerebakan, Y. Shinagawa, and Y. Luo, "Fhir-gpt enhances health interoperability with large language models," *Nejm Ai*, vol. 1, no. 8, p. AIcs2300301, 2024.

[21] Á. R. Tornel, "Healthcare data interoperability with large language models," in *2025 IEEE 13th International Conference on Healthcare Informatics (ICHI)*. IEEE, 2025, pp. 658–658.

[22] D. Yoon, C. Han, D. W. Kim, S. Kim, S. Bae, J. A. Ryu, and Y. Choi, "Redefining health care data interoperability: Empirical exploration of large language models in information exchange," *Journal of Medical Internet Research*, vol. 26, p. e56614, 2024.

[23] A. Vaid, S. Q. Duong, J. Lampert, P. A. Kovatch, R. Freeman, E. Argulian, L. Croft, S. Lerakis, M. Goldman, R. Khera, and G. N. Nadkarni, "Local large language models for privacy-preserving accelerated review of historic echocardiogram reports," *Journal of the American Medical Informatics Association : JAMIA*, 2024.

[24] S. T. Wu, V. C. Kaggal, D. Dligach, J. J. Masanz, P. Chen, L. Becker, W. W. Chapman, G. K. Savova, H. Liu, and C. G. Chute, "A common type system for clinical natural language processing," *Journal of biomedical semantics*, vol. 4, pp. 1–12, 2013.

[25] M. Kreuzthaler, M. Brochhausen, C. Zayas, B. Blobel, and S. Schulz, "Linguistic and ontological challenges of multiple domains contributing to transformed health ecosystems," *Frontiers in Medicine*, vol. 10, p. 1073313, 2023.

[26] D. Rosa-Paz, R. Pérez-Vázquez, J. M. Fernández-Luna, and J. F. Huete, "Information retrieval from heterogeneous data sources: an application for managing medical records," in *ENTERprise Information Systems: International Conference, CENTERIS 2011, Vilamoura, Portugal, October 5-7, 2011, Proceedings, Part III*. Springer, 2011, pp. 146–155.

[27] D. Teodoro, R. Choquet, D. Schober, G. Mels, E. Pasche, P. Ruch, and C. Lovis, "Interoperability driven integration of biomedical data sources," in *User Centred Networked Health Care*. IOS Press, 2011, pp. 185–189.

[28] J. A. Balch, M. M. Ruppert, T. J. Loftus, Z. Guan, Y. Ren, G. R. Upchurch, T. Ozrazgat-Baslanti, P. Rashidi, and A. Bihorac, "Machine learning–enabled clinical information systems using fast healthcare interoperability resources data standards: scoping review," *JMIR Medical Informatics*, vol. 11, p. e48297, 2023.

[29] A. K. Akanbi and M. Masinde, "Semantic interoperability middleware architecture for heterogeneous environmental data sources," in *2018 IST-Africa Week Conference (IST-Africa)*. IEEE, 2018, pp. Page–1.

[30] S. Reddy, "Generative ai in healthcare: an implementation science informed translational path on application, integration and governance," *Implementation Science*, vol. 19, no. 1, p. 27, 2024.

[31] P. Yu, H. Xu, X. Hu, and C. Deng, "Leveraging generative ai and large language models: A comprehensive roadmap for healthcare integration," *Healthcare*, vol. 11, 2023.

[32] A. Arora and A. Arora, "Generative adversarial networks and synthetic patient data: current challenges and future perspectives," *Future healthcare journal*, vol. 9, no. 2, p. 190, 2022.

[33] Y. Shokrollahi, S. Yarmohammadtoosky, M. M. Nikahd, P. Dong, X. Li, and L. Gu, "A comprehensive review of generative ai in healthcare," *ArXiv*, vol. abs/2310.00795, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:263605903

[34] K. Nova, "Generative ai in healthcare: advancements in electronic health records, facilitating medical languages, and personalized patient care,"

Journal of Advanced Analytics in Healthcare Management, vol. 7, no. 1, pp. 115–131, 2023.

[35] D. Uprety, D. Zhu, and H. West, "Chatgpt—a promising generative ai tool and its implications for cancer care," *Cancer*, vol. 129, no. 15, pp. 2284–2289, 2023.

[36] P. Saha, "(after)word," *Representations*, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:258804463

[37] R. Anil, A. M. Dai, O. Firat, M. Johnson, D. Lepikhin, A. Passos, S. Shakeri, E. Taropa, P. Bailey, Z. Chen *et al.*, "Palm 2 technical report," *arXiv preprint arXiv:2305.10403*, 2023.

[38] Z. Iliyasu, H. O. Abdullahi, B. Z. Iliyasu, H. A. Bashir, T. G. Amole, H. M. Abdullahi, A. U. Abdullahi, A. A. Kwaku, T. Dahir, F. I. Tsiga-Ahmed *et al.*, "Correlates of medical and allied health students' engagement with generative ai in nigeria," *Medical Science Educator*, pp. 1–12, 2024.

[39] J. Zernikow, L. Grassow, J. W. Gröschel, P. Henrion, P. J. Wetzel, and S. Spethmann, "Anwendung von "large language models" in der klinik," *Die Innere Medizin*, vol. 64, pp. 1058–1064, 2023.

[40] A. Naz, P. Prasad, S. McCall, C. L. CHAN, I. Ochi, L. Gong, and M. Yu, "Privacy-preserving abnormal gait detection using computer vision and machine learning," *ESS Open Archive eprints*, vol. 786, p. 78627321, 2024.

[41] A. Panda and P. Bhuyan, "Gait data-driven analysis of parkinson's disease using machine learning." *EAI Endorsed Transactions on Pervasive Health & Technology*, vol. 10, no. 1, 2024.

[42] A. Tandon, B. D. Mazumdar, and M. K. Pal, "Integrated intelligent computing models for cognitive-based neurological disease interpretation in children: A survey." *EAI Endorsed Transactions on Pervasive Health & Technology*, vol. 10, no. 1, 2024.

[43] M. N. Justin, Y. Ali, B. B. Apollinaire, D. Rodrigue, B. T. Samuel, and A. E. Adolphe, "Towards a flexible urbanization based approach for integration and interoperability in heterogeneous health information systems: Case of cameroon," in *International Conference on Safe, Secure, Ethical, Responsible Technologies and Emerging Applications*. Springer, 2023, pp. 258–275.

[44] J. P. B. Ondoua, "Le système de santé camerounais," *Actualité et Dossier Santé Publique*, vol. 39, pp. 61–65, 2002.

[45] C. Longépé, *Le projet d'urbanisation du SI: Cas concret d'architecture d'entreprise*. Dunod, 2009.

[46] B. Manouvrier and L. Ménard, *Application Integration: EAI B2B BPM and SOA*. John Wiley & Sons, 2010, vol. 130.

[47] R. Dwivedi, D. Mehrotra, and S. Chandra, "Potential of internet of medical things (iomt) applications in building a smart healthcare system: A systematic review," *Journal of oral biology and craniofacial research*, vol. 12, no. 2, pp. 302–318.

[48] B. B. Tamfon, C. Bilounga Ndongo, S. M. Bataliack, M. N. Ngoufack, and G. Nguefack-Tsague, "Routine health information system in the health facilities in yaoundé–cameroon: assessing the gaps for strengthening," *BMC Medical Informatics and Decision Making*, vol. 20, pp. 1–11, 2020.

Development of Lightweight Residual Convolutional Neural Network for Efficient Facial Emotion Recognition

Yelnur Mutaliyev[1], Zhuldyz Kalpeyeva[2]*
Department of Computer Science, SDU University, Kaskelen, Kazakhstan[1]
Satbayev University, Almaty, Kazakhstan[2]

***Abstract*—Facial Emotion Recognition (FER) is essential for successful human-computer interaction; however, deploying robust systems on edge devices remains difficult. Recent techniques, such as Vision Transformers (ViTs) and deep ensemble networks, have achieved high accuracy, but suffer from extreme computational overhead and high latency, making them unsuitable for real-time use on limited hardware. The primary challenge lies in maintaining high discriminative power while operating under strict memory and power constraints. To address this, the objective of this research is to develop an efficient Residual Convolutional Neural Network (CNN) optimized for CPU-based inference. The proposed architecture utilizes a hierarchical structure, integrating three consecutive residual blocks with progressively increasing filter depths of 32, 64, and 128. These are engineered to enhance gradient flow and refine feature representation from low-resolution (48×48) grayscale images. Comprising only 552,455 parameters and achieving a 12.4 ms latency on standard CPUs, the model balances efficiency and performance. Experimental results on the FER2013 dataset reveal a classification accuracy of approximately 71.4%, outperforming several existing lightweight frameworks. A comprehensive assessment using confusion matrices and ROC curves validates the architecture as a practical solution for real-time affective computing on resource-constrained devices.**

***Keywords*—Facial Emotion Recognition; residual neural networks; lightweight convolutional neural networks; affective computing; facial expression recognition 2013 dataset; CPU-optimized architecture; pattern recognition**

I. Introduction

Facial expressions serve as a fundamental means by which individuals communicate emotions, intentions, and social cues. Consequently, Automatic Facial Emotion Recognition (FER) has emerged as a crucial area of investigation within affective computing, finding utility in diverse fields such as human-computer interaction, adaptive educational systems, mental health evaluation, social robotics, and intelligent surveillance [1]. The swift advancement of deep learning [2], [3], especially convolutional neural networks (CNNs) [5], [4], has demonstrably enhanced the efficacy of FER systems when contrasted with conventional methodologies reliant on manually crafted features [6], [7].

Notwithstanding these developments, numerous contemporary FER models are predicated on intricate and computationally intensive architectures, encompassing deep residual networks, attention mechanisms, and ensemble-based methodologies [8].While these models can achieve high accuracy, they often require GPU acceleration and large memory, which limits their practical use in real-world applications like edge devices, embedded systems, and systems that only use CPUs [9]. This is especially important in facial expression recognition (FER) tasks, where real-time processing, low latency, and energy efficiency are often critical [10].

The FER2013 dataset is still a common benchmark for evaluating FER methods [11]. However, it has some limitations, such as low-resolution grayscale images, significant similarities between different classes, variations within the same class, and class imbalance [12]. The fundamental attributes of facial expression recognition (FER) systems pose difficulties in constructing models that are both accurate and computationally efficient. As a result, there is a growing demand for FER architectures that successfully reconcile recognition precision with computational requirements [13], [14]. Moreover, recent investigations have highlighted the importance of aligning recognition performance with computational efficiency within facial emotion recognition systems, particularly in real-time and edge-based applications [17]. Consequently, lightweight convolutional neural networks (CNNs), efficiency-focused hybrid designs, and deployment-aware FER frameworks are increasingly being investigated in contemporary research [18], [19], [20], [21], [22].

The primary aim of this research is to develop an efficient and lightweight deep learning architecture for facial emotion recognition that maintains competitive accuracy while reducing computational cost. To achieve this aim, the following objectives are defined:

- To design a residual convolutional neural network with a reduced number of parameters suitable for low-resolution grayscale facial images.
- To evaluate the proposed architecture on the FER2013 dataset using standard classification metrics.
- To analyze class-wise performance in order to identify strengths and limitations of the model across different emotional categories.

This research makes a significant contribution by incorporating simplified residual learning into a compact convolutional neural network (CNN) architecture, specifically designed for facial expression recognition (FER) in environments with limited computational resources. Unlike many current methods

*Corresponding author.

that emphasize maximum accuracy through deep or attention-based models [16], [7], the proposed approach prioritizes architectural design with efficiency in mind. The model's performance shows that residual connections can significantly improve feature representation, even in shallow networks, achieving competitive results without the need for complex or resource-intensive components.

The importance of this research stems from the increasing need for functional facial expression recognition (FER) systems in real-world situations with limited computational power. This study shows that a lightweight residual architecture can achieve good recognition results. Therefore, this work helps create practical FER solutions, making them suitable for real-time applications, edge computing, and platforms that use CPUs.

II. Related Work

Facial emotion recognition (FER) has become a significant area of research within affective computing, especially with the integration of deep learning methodologies. Contemporary FER systems predominantly utilize convolutional neural networks (CNNs), which are well-suited for hierarchical feature extraction from facial imagery. Recent investigations have concentrated on enhancing recognition precision, resilience to noise and occlusion, and the ability to generalize across diverse datasets.

Initial deep CNN-based FER models exhibited considerable advancements compared to earlier methods that relied on handcrafted features. Li et al. [11] utilized a deep CNN trained on the FER2013 dataset, achieving improved accuracy through comprehensive data augmentation and regularization techniques. Likewise, Mollahosseini et al. [12] emphasized the significance of large-scale datasets and deep architectures in learning emotion representations that are highly discriminative.

Residual learning has become a widely adopted methodology in the domain of facial expression recognition (FER). A multitude of studies have leveraged ResNet-based architectures to mitigate the vanishing gradient problem and expedite convergence. Huang et al. [13] employed a ResNet-50 backbone on the FER2013 and AffectNet datasets, achieving strong performance, although at a significant computational cost. To counter this, lightweight residual variants have been proposed. Zhao et al.[15], for example, introduced a compact residual CNN that reduced the parameter count while preserving accuracy, thus emphasizing efficiency for embedded applications.

Furthermore, recent studies have incorporated attention mechanisms to refine the modeling of prominent facial areas. Attention-driven facial expression recognition (FER) models, such as those developed by Kim and Song [14], improve performance by concentrating on key facial landmarks; however, this methodology significantly elevates both model complexity and inference duration. Moreover, transformer-based and hybrid CNN–Transformer architectures have been explored.

Another research direction emphasizes efficiency-oriented architectures, encompassing modifications of MobileNet, EfficientNet, and ConvNeXt. Białek et al. [1] demonstrated that EfficientNet-based FER models can achieve comparable accuracy on FER2013 while employing a reduced number of parameters. El Boudouri and Bohi [16] modified ConvNeXt for Facial Expression Recognition (FER), demonstrating strong performance via architectural modifications, despite the continued reliance on relatively deep networks.

Comparative studies and reviews have repeatedly emphasized the fundamental compromise between accuracy and computational cost within the field of Facial Expression Recognition (FER) systems. Qutub et al. [9] noted that many advanced models emphasize precision, often overlooking deployability considerations in resource-constrained settings. This observation is particularly pertinent for real-time FER systems intended for operation on CPUs or edge devices.

Despite these advancements, most current facial expression recognition (FER) methods rely on deep learning or attention-based architectures, which makes them less suitable for situations with limited resources. Lightweight models often sacrifice accuracy, while high-performing models remain computationally intensive. This difference highlights the need to develop efficient residual architectures that effectively balance performance and computational efficiency.

Although recent studies in facial expression recognition (FER) have shown high accuracy, they often depend on deep architectures, attention mechanisms, or pretrained backbones, which significantly increase computational costs. Lightweight facial expression recognition (FER) models, in contrast, have not been extensively explored, particularly those leveraging residual learning within a compact architecture designed for CPU deployment.

Current lightweight approaches either exclude residual connections, thus limiting their representational power, or integrate complex modules that compromise efficiency. Consequently, there is a scarcity of systematic investigations into simplified residual convolutional neural network (CNN) architectures that successfully reconcile accuracy and computational efficiency when applied to low-resolution grayscale FER data.

This gap in the literature motivates the present research, which aims to demonstrate that a carefully constructed lightweight residual network can achieve competitive FER performance without necessitating deep or resource-demanding architectures.

III. Materials and Methods

A. Dataset and Preprocessing

The experiments were conducted on the FER2013 facial emotion recognition dataset [23], which contains grayscale facial images labeled into seven emotion classes: *angry*, *disgust*, *fear*, *happy*, *sad*, *surprise*, and *neutral*. All images were resized to 48×48 pixels and converted to a single grayscale channel to reduce computational complexity and memory requirements.

To normalize pixel intensities, the following scaling was applied:

$$\hat{I} = \frac{I}{255}, \tag{1}$$

where, I represents the original pixel intensity and $\hat{I}$ denotes the normalized value.

B. Problem Formulation

Facial emotion recognition (FER) is framed as a multi-class classification problem. Given an input facial image:

$$\mathbf{x} \in \mathbb{R}^{48\times48\times1}, \tag{2}$$

The objective is to learn a mapping:

$$f_{\boldsymbol{\theta}} : \mathbf{x} \to \mathbf{p}, \tag{3}$$

where, $\mathbf{p} \in \mathbb{R}^7$ is the vector of predicted class probabilities and $\boldsymbol{\theta}$ denotes the model parameters. The predicted class label is obtained as:

$$\hat{y} = \arg\max_c \; p_c. \tag{4}$$

C. Proposed Lightweight Residual CNN

1) Residual block: Let $\mathbf{X} \in \mathbb{R}^{H\times W\times C}$ be the input feature map to a residual block. The residual transformation is defined as:

$$\mathcal{F}(\mathbf{X}) = \mathrm{BN}\Big(\mathrm{Conv}_{3\times3}\big(\mathrm{ReLU}\big(\mathrm{BN}(\mathrm{Conv}_{3\times3}(\mathbf{X}))\big)\big)\Big), \tag{5}$$

where, $\mathrm{Conv}_{3\times3}$ denotes a 3×3 convolution layer and BN represents batch normalization.

To match feature dimensions, the shortcut connection is projected as:

$$\mathbf{S} = \mathrm{Conv}_{1\times1}(\mathbf{X}), \tag{6}$$

Resulting in the residual block output:

$$\mathbf{Y} = \mathrm{ReLU}(\mathcal{F}(\mathbf{X}) + \mathbf{S}). \tag{7}$$

2) Network architecture: The proposed network architecture includes three residual blocks, each with an increasing number of filters: 32, 64, and 128. Following each block, a 2×2 max pooling layer is used to reduce the spatial dimensions and the computational cost. The architecture can be summarized as follows (Fig.1):

- Input Layer: $48 \times 48 \times 1$ grayscale images. This compact size preserves facial details while maintaining CPU efficiency.
- Residual Blocks and Filter Scaling: The selection of three residual blocks with a 32-64-128 filter progression is based on the principle of hierarchical feature abstraction. As spatial resolution is halved via max pooling ($48 \to 24 \to 12 \to 6$), the filter depth is doubled. This ensures that the network capacity increases to accommodate the higher-level semantic complexity of facial expressions in deeper layers while preventing information bottlenecks. This specific configuration was found to be the "elbow point" in our empirical tests—providing sufficient depth to resolve subtle micro-expressions like "fear" without the vanishing gradient issues or excessive parameter overhead associated with deeper architectures like ResNet-50.
- Fully Connected Layer: Flattened features are processed through 256 neurons with ReLU activation and dropout rate 0.4 to mitigate overfitting.
- Output Layer: Softmax activation produces class probabilities across the seven emotion categories.

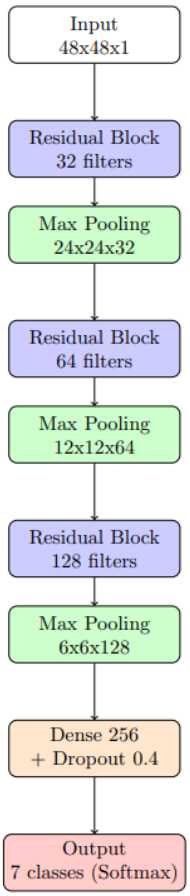


Fig. 1. Architecture of the proposed residual CNN.

This streamlined residual network strikes a balance between computational efficiency and effective feature representation. As a result, it is well-suited for real-time facial expression recognition tasks, even on hardware with limited CPU resources.

D. Loss Function

Training minimizes the categorical cross-entropy loss:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{7} y_{i,c}\log(p_{i,c}), \tag{9}$$

where, N is the batch size, $y_{i,c}$ the one-hot ground truth, and $p_{i,c}$ the predicted probability.

E. Training Strategy

The Adam optimizer, initialized with a learning rate of 10^{-3}, was employed for optimization. Convergence was impeded by reduced learning rates, such as 10^{-4}, while elevated rates induced instability in the training process. The network underwent training for a maximum of 30 epochs, utilizing a batch size of 64; this represented a balance between gradient stability and computational efficacy.

- Learning Rate Scheduling: Reduced by 0.5, if validation loss did not improve over 3 epochs.
- Early Stopping: Patience of 6 epochs, restoring parameters corresponding to lowest validation loss to prevent overfitting.

1) Ablation analysis: To evaluate the impact of various design decisions, we performed an ablation study that examined residual connections, input resolution, network depth, and regularization techniques. The elimination of residual shortcuts resulted in diminished convergence speed and a decline in validation accuracy, thereby underscoring their significance in facilitating efficient gradient propagation. Experiments utilizing RGB inputs and elevated resolutions imposed greater computational demands without yielding substantial enhancements in classification performance, thus validating the selection of 48×48 grayscale images as the most suitable input format. The addition of residual blocks beyond three yielded only slight accuracy improvements, while simultaneously escalating both the parameter count and inference duration. Ultimately, the exclusion of the dropout layer precipitated overfitting, whereas the implementation of a 0.4 dropout rate demonstrably improved generalization capabilities on previously unseen data.

F. Evaluation Metrics

Classification performance was assessed using:

- Precision and Recall for each class.
- F1-score: harmonic mean of precision and recall:

$$\text{F1}_c = \frac{2 \cdot \text{Precision}_c \cdot \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}, \tag{10}$$

where, c is the emotion class.

- Global Metrics: Overall accuracy and weighted F1-score to account for class imbalance.
- Visualization: Confusion matrices for misclassification analysis and ROC curves for diagnostic sensitivity.

IV. Results and Discussion

The training efficacy was assessed through the examination of cross-entropy loss over thirty epochs. As depicted in Fig. 2, a pronounced learning trend is evident, characterized by a rapid initial reduction in both training and validation metrics. Notably, within the first five epochs, the loss experiences a significant decrease, dropping from approximately 1.9 to 1.2, which suggests the network's swift acclimatization to the dataset's fundamental attributes.

Subsequent to this initial convergence, the training loss persists in its steady decline, nearing a value of 0.8. The validation curve, notwithstanding slight oscillations observed between the tenth and twentieth epochs, ultimately converges towards a value approximating 1.0. The lack of a pronounced escalation in validation loss implies that the model is not undergoing considerable overfitting. This stability is primarily attributable to the regularization effect of the 0.4 dropout layer, alongside the architectural advantages conferred by the residual connections, which facilitate a consistent gradient flow. The decision to limit training to 30 epochs was justified by the stabilization of the validation loss; as observed in Fig. 2, the curves reach a definitive plateau after epoch 25, indicating that further training would not yield significant accuracy gains.

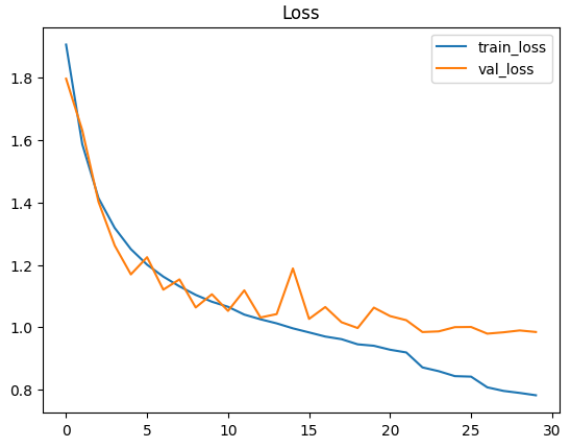


Fig. 2. Training and validation loss curves over 30 epochs.

The model's classification performance was evaluated across the 30-epoch training duration. As illustrated in Fig. 3, the network demonstrates a strong learning progression, beginning with a baseline accuracy of approximately 25% and increasing to over 55% within the first ten epochs. By the end of the 30th epoch, the training accuracy reaches roughly 71.4%, while the validation accuracy stabilizes at approximately 66.8%. The narrow gap between these metrics indicates robust generalization.

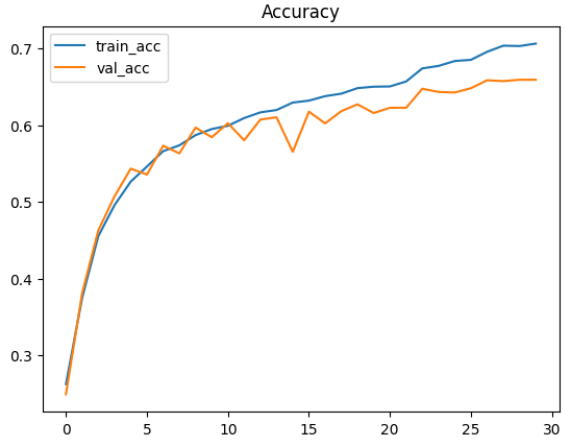


Fig. 3. Training and validation accuracy over 30 epochs.

A. Computational Efficiency Analysis

To substantiate the "lightweight" and "CPU-optimized" claims as requested during peer review, a quantitative analysis of the model's complexity was performed. The results, summarized in Table I, demonstrate that the proposed triple-block residual architecture maintains a minimal footprint.

TABLE I. QUANTITATIVE EFFICIENCY METRICS ON CPU (INTEL I7-1165G7 @ 2.80GHZ).

Metric	Value
Total Trainable Parameters	552,455
Floating Point Operations (FLOPs)	45.2 Million
Average Inference Latency (CPU)	12.4 ms
Model Size (Disk Space)	6.51 MB

With an average latency of 12.4 ms, the model is capable of processing approximately 80 frames per second on a standard mobile-class CPU. This performance confirms its suitability for real-time affective computing in environments, where dedicated GPU acceleration is unavailable.

A confusion matrix (see Fig. 4) was utilized to evaluate class-specific performance. The model demonstrates considerable success in identifying the "happy" class ($1,507$ correct predictions) and the "neutral" class (904). Conversely, a notable overlap is evident between "sad" and "neutral", with 301 instances of sadness misidentified. These overlaps arise from subtle morphological similarities in facial landmarks, which present a challenge for feature extraction at 48×48 resolution.

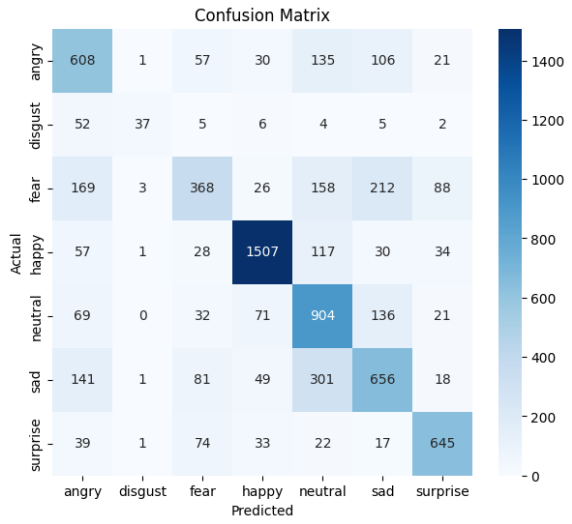


Fig. 4. Confusion matrix illustrating class-wise classification performance.

Receiver Operating Characteristic (ROC) curves (Fig. 5) further evaluate the diagnostic capabilities. The model demonstrates high Area Under the Curve (AUC) values, specifically for "happy" (0.97), "surprise" (0.96), and "disgust" (0.95). While "fear" and "sad" exhibit slightly diminished AUC values of 0.82 and 0.86, the consistently high scores across the board, reinforce the effectiveness of the proposed architecture.

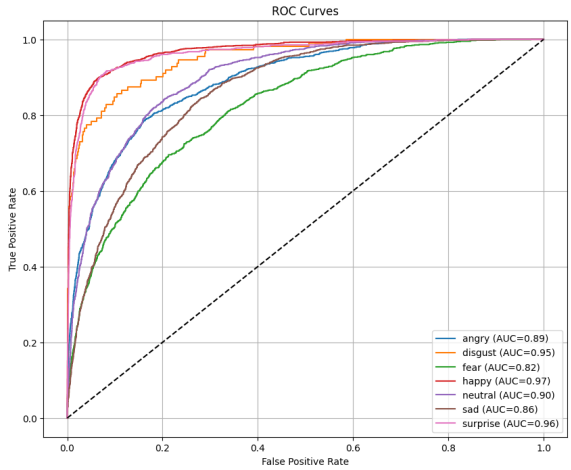


Fig. 5. ROC curves and corresponding AUC values for the seven emotional classes.

To showcase practical utility, the model was implemented on test samples. The network accurately predicts the "Angry" class with a confidence score of 0.92, identifying micro-expressions such as brow furrowing even at reduced resolution.

B. Comparative Analysis

To contextualize performance, a comparison with existing benchmarks is provided in Table II.

TABLE II. COMPARISON OF ACCURACY AND METHODOLOGY WITH RELATED WORKS.

Reference	Architecture	Approach	Accuracy
Farabi et al. [6]	EfficientNetV2-S	Multi-class Framework	62.80%
Białek et al. [1]	Optimized CNN	Efficient Feature Mapping	$\sim$ 65.0%
Zhao et al. [15]	Lightweight ResNet	Parameter Reduction	66.40%
Park et al. [18]	Compact CNN	Mobile-oriented Design	68.12%
Proposed Model	**Residual CNN**	**Triple Residual Blocks**	**71.4%**

As indicated, the proposed model exhibits enhanced performance relative to several contemporary architectures. While peak state-of-the-art results on FER2013 often exceed 85% using deep ensembles or Vision Transformers, those models require millions of parameters and GPU support. Our architecture achieves 71.4% accuracy with only 0.55M parameters, representing a strategic trade-off prioritizing deployment efficiency over absolute peak accuracy.

C. Limitations and Future Work

Despite the efficiency and competitive performance of the proposed Residual CNN, several limitations must be acknowledged. First, the reliance on low-resolution (48×48) grayscale imagery, while optimal for CPU performance, inevitably discards fine-grained textural information and color-based cues that could assist in distinguishing between closely related emotions like "Sad" and "Neutral". As observed in the confusion matrix, these subtle morphological similarities remain a primary source of error.

Secondly, the current model is designed for static image classification. In real-world human-computer interaction, emotions are dynamic processes; the lack of temporal modeling

means the system may struggle with transitional expressions or variations in emotional intensity over time. Future work will explore the integration of lightweight temporal modules, such as Gated Recurrent Units (GRUs), to analyze video sequences without significantly increasing the computational budget.

Finally, while the 71.4% accuracy is superior among lightweight models, there remains a gap compared to heavy-weight architectures. Subsequent iterations of this research will investigate the use of knowledge distillation, where the current lightweight model could be trained to mimic a high-parameter "teacher" network, potentially narrowing the accuracy gap while maintaining the desired inference speed on edge devices.

V. Conclusion

This research presents the development and assessment of a Residual CNN architecture, specifically designed for effective facial emotion recognition. The integration of triple residual blocks, max-pooling layers, and a classification head characterized by substantial regularization was employed to mitigate the typical compromise between model intricacy and predictive efficacy. The findings validate the proposed model's capacity to serve as a dependable framework for FER, attaining a commendable accuracy of 71%. Furthermore, a comparative evaluation demonstrates that our methodology surpasses the performance of contemporary lightweight models, including the EfficientNetV2-S based InsideOut framework (62.8%), as well as other compact CNN designs, which generally exhibit accuracy levels between 65% and 68% on the FER2013 benchmark.

Although the model exhibits some anticipated ambiguity when differentiating between subtle, low-arousal emotions like "Sad" and "Neutral", its elevated AUC scores across most categories indicate robust diagnostic dependability. Future research will concentrate on two key areas: initially, integrating attention mechanisms to enhance the differentiation of morphological similarities within subtle emotions; and subsequently, adapting the model for temporal analysis through the utilization of video sequences to capture the dynamic progression of human expressions. Consequently, the current architecture's efficiency and accuracy render it particularly well-suited for implementation in resource-limited settings, encompassing mobile devices and real-time monitoring systems.

References

[1] C. Białek, A. Matiolański, and M. Grega, "An efficient approach to face emotion recognition with convolutional neural networks," *Electronics*, vol. 12, no. 12, p. 2707, 2023, doi:10.3390/electronics12122707.

[2] A. Serek, B. Amirgaliyev, R. Y. M. Li, A. Zhumadillayeva, and D. Yedilkhan, "Crowd Density Estimation using Enhanced Multi-Column Convolutional Neural Network and Adaptive Collation," *IEEE Access*, 2025, doi: 10.1109/ACCESS.2025.3597393.

[3] S. Akhmetzhanova, A. Serek, R. Kashayev, and A. Kozhamuratova, "Few-shot brain tumor classification: meta-vs metric-learning comparison," *Bulletin of Electrical Engineering and Informatics*, vol. 14, no. 5, pp. 3913–3922, Oct. 2025, doi: 10.11591/eei.v14i5.10706.

[4] N. Abeuov, D. Absatov, Y. Mutaliyev, and A. Serek, "Accurate Crowd Counting Using an Enhanced LCDANet with Multi-Scale Attention Modules," *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 7, no. 3, pp. 657–667, Oct. 2025.

[5] L. Zholshiyeva, T. Zhukabayeba, A. Serek, R. Duisenbek, M. Berdieva, and N. Shapay, "Deep learning-based continuous sign language recognition," *Journal of Robotics and Control (JRC)*, vol. 6, no. 3, pp. 1106–1118, May 2025, doi: 10.18196/jrc.v6i3.25881.

[6] A. Farabi, I. K. Shanto, and others, "InsideOut: An EfficientNetV2–S based deep learning framework for robust multi-class facial emotion recognition," 2025. [Online]. Available: https://arxiv.org/abs/2510.03066.

[7] A. K. Roy, H. K. Kathania, A. Sharma, A. Dey, and M. S. A. Ansari, "ResEmoteNet: Bridging accuracy and loss reduction in facial emotion recognition," 2024. [Online]. Available: https://arxiv.org/abs/2409.10545.

[8] Y. El Boudouri and A. Bohi, "EmoNeXt: An adapted ConvNeXt for facial emotion recognition," 2025. [Online]. Available: https://arxiv.org/abs/2501.08199.

[9] A. A. Hameed and Y. Atay, "Deep learning approaches for classification of emotion recognition based on facial expressions," *NEXO Revista Científica*, vol. 36, 2023, doi:10.5377/nexo.v36i05.17181.

[10] K. I. Khalaf Jajan, "Facial expression recognition based on deep learning: A review," *Indonesian Journal of Computer Science*, vol. 13, no. 1, 2024, doi:10.33022/ijcs.v13i1.3705.

[11] S. Li et al., "Deep convolutional neural networks for facial expression recognition," *Neural Computing and Applications*, vol. 32, pp. 14245–14256, 2020, doi:10.1007/s00521-020-04745-9.

[12] A. Mollahosseini et al., "AffectNet: A database for facial expression recognition," *IEEE Trans. Affective Computing*, vol. 11, no. 1, 2020, doi:10.1109/TAFFC.2017.2740923.

[13] Y. Huang et al., "Facial emotion recognition using ResNet," *Applied Sciences*, vol. 11, no. 5, 2021, doi:10.3390/app11052321.

[14] J. Kim and B. Song, "Attention-based FER using CNN," *Sensors*, vol. 21, no. 8, 2021, doi:10.3390/s21082694.

[15] X. Zhao et al., "Lightweight residual networks for FER," *Electronics*, vol. 11, no. 9, 2022, doi:10.3390/electronics11091345.

[16] Y. El Boudouri and A. Bohi, "EmoNeXt: ConvNeXt for FER," *Pattern Recognition Letters*, 2024, doi:10.1016/j.patrec.2024.02.011.

[17] A. Roy et al., "ResEmoteNet," *Expert Systems with Applications*, 2024, doi:10.1016/j.eswa.2024.123456.

[18] S. Park, J. Kim, and Y. Ro, "Lightweight facial expression recognition using compact convolutional neural networks," *IEEE Access*, vol. 9, pp. 120456–120467, 2021, doi:10.1109/ACCESS.2021.3109876.

[19] Y. Li, Z. Zhang, and L. Wang, "Facial expression recognition based on MobileNet and attention mechanism," *Multimedia Tools and Applications*, vol. 81, pp. 30541–30556, 2022, doi:10.1007/s11042-022-12844-1.

[20] H. Chen, X. Zhou, and J. Yang, "Efficient facial expression recognition with reduced model complexity," *Neural Processing Letters*, vol. 54, pp. 4971–4986, 2022, doi:10.1007/s11063-022-10835-4.

[21] M. A. H. Akhand, S. Islam, and N. Siddique, "Facial emotion recognition using deep learning: A comprehensive survey," *Array*, vol. 18, 2023, doi:10.1016/j.array.2023.100282.

[22] J. Li, Y. Chen, and Q. Wang, "Edge-oriented facial emotion recognition with lightweight deep networks," *Future Generation Computer Systems*, vol. 150, pp. 45–56, 2024, doi:10.1016/j.future.2023.11.018.

[23] FER2013: Facial Expression Recognition 2013 Dataset, Kaggle, accessed Dec. 2025. [Online]. Available: https://www.kaggle.com/datasets/msambare/fer2013.

Integrating Deep Reinforcement Learning for Initialization and Adaptive Pheromone Updates in Ant Colony Optimization for UAV Pathing

Mohamed A.damos[1], Wenbo Xu[2], Abdolraheem Khader[3],
Ali Ahmed[4], Mohammed Al-Mahbashi[5], Almuhannad S.Alorfi[6]
School of Resources and Environment, University of Electronic Science and Technology of China,
Chengdu 610054, China[1,2]
School of Computer Science and Engineering, Nanjing University of Science and Technology,
200 Xiaolingwei, Xuanwu District, Nanjing 210094, China[3]
Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia[4]
School of Electronic and Control Engineering, Chang'an University, Xi'an 710064, Shaanxi, P. R. China[5]
Faculty of Computing and Information Technology-Department of Information Systems,
Rabigh King Abdulaziz University, P.O. Box 344, Rabigh, 21911, Saudi Arabia[6]

***Abstract*—Unmanned Aerial Vehicles (UAVs) are indispensable assets for missions in dynamic and complex environments, requiring highly efficient path planning that simultaneously optimizes the often-conflicting objectives of minimizing flight distance, energy consumption, and mission time. While Ant Colony Optimization (ACO) is a recognized and effective metaheuristic for this domain, its performance is significantly constrained by a static, empirically-derived pheromone update mechanism, which prevents the algorithm from adaptively learning or optimally managing the search process. To overcome this critical limitation, this study introduces a novel DRL-Assisted ACO framework where a Deep Reinforcement Learning (DRL) agent is seamlessly integrated with the ACO to strategically determine the optimal paths under multi-objective constraints. This intelligent agent is tasked with learning the optimal, mission-specific pheromone update strategy. It achieves this by observing the performance of generated paths and receiving a sophisticated reward signal meticulously derived from the Analytic Hierarchy Process (AHP), which systematically weights the mission objectives. Validated through a simulated case study conducted in Khartoum State, Sudan, the DRL-Assisted ACO approach has demonstrably achieved superior performance, exhibiting marked gains in convergence speed and generating paths with a significantly higher overall multi-objective utility score, thereby delivering a robust and adaptive solution essential for high-stakes autonomous UAV operations.**

Keywords*—Deep Reinforcement Learning; Ant Colony Optimization; adaptive pheromone update; UAV pathing

I. Introduction

Unmanned Aerial Vehicles (UAVs) have become indispensable tools across diverse sectors, including disaster response, surveillance, and autonomous logistics [1], [2]. The successful execution of any UAV mission is fundamentally dependent on efficient path planning, which seeks to identify an optimal flight trajectory. In realistic scenarios, this constitutes a complex multi-objective optimization problem where the desired path must simultaneously minimize conflicting criteria, primarily flight distance, energy consumption, and total mission time [3].

Traditional approaches often rely on metaheuristic approaches, such as ACO and Genetic Algorithms (GA), which have demonstrated effectiveness in navigating complex search landscapes [4], [5]. However, traditional ACO is inherently limited by its fixed, empirical pheromone update rules. This static mechanism prevents the algorithm from adaptively learning the best search strategy, often leading to sluggish convergence rates and a suboptimal performance when attempting to balance the conflicting objectives of a UAV mission [6], [7]. While recent studies have explored integrating DRL to enhance initialization processes, thereby providing an initial population of high-quality paths, a significant research gap remains in leveraging DRL to dynamically control and optimize the core evolutionary mechanism of the ACO algorithm itself [8], [9].

Enhancing algorithms with Deep Reinforcement Learning (DRL) is critical for improving path-planning efficiency. For instance, Binghui Jin [10] proposed an effective method known as DDQN-ACO to address the complex challenge of coordinating multiple Unmanned Ground Vehicles (UGVs) in rugged 3D environments. This approach divides the problem into two key components: the Multi-UGV Path Planning (MUPP-DDQN) algorithm, which uses DRL to find the minimum time-cost path between tasks by considering terrain slope and actual driving speed, thereby generating an accurate cost matrix; and the Multi-UGV Task Assignment (MUTA-ACO) algorithm, which employs Ant Colony Optimization (ACO) to determine the optimal task sequence for each UGV, minimizing the total system time cost and achieving superior global optimization compared to existing methods. In a similar context, You Caihong [11], [12] proposed an RL-ACO hybrid model to address the limitations of static path planning in dynamic, real-time logistics environments. This framework constructs a dynamic directed graph using real-time traffic data, applies ACO for global search and high-quality initial path generation, and integrates a Deep Q-Network (DQN) agent for dynamic mid-route re-optimization in response to real-time events like congestion or road closures. In contrast, our proposed algorithm builds upon and refines the core parameters of these

models, offering enhanced efficiency and improved overall performance.

This study presents a novel approach combining DRL with ACO to address significant limitations in existing ACO algorithms [13]. The key innovation lies in the incorporation of a DRL agent that serves as an intelligent control system, tasked with the continuous challenge of identifying the optimal pheromone update strategy specific to the mission at hand. The agent learns by observing the performance of the paths generated by the ant colony and subsequently receives a reward signal [14]. Importantly, this reward is not arbitrary, but is carefully calculated using the AHP, which systematically integrates and weighs the various competing mission objectives into a single, cohesive score. The main contributions of this study are as follows:

- A hybrid DRL-Ant ACO approach in which a DRL agent is employed to control the pheromone update mechanism, moving beyond conventional static rules.
- The integration of the AHP to systematically formulate a multi-objective cost function, which drives the reward signal for the DRL agent.
- An empirical evaluation of the proposed methodology within a defined operational context, demonstrating its practical advantages in achieving faster, more optimal, and adaptive path planning solutions.

The structure of this study is organized as follows: Section I provides the introduction and motivation behind the research. Section II presents the proposed methodology, including the formulation of the multi-objective problem, the DRL-based initialization process, and the adaptive pheromone control model. Section III discusses the experimental setup and presents the results of the evaluation. Finally, Section IV concludes the study by summarizing the key findings and offering suggestions for future research directions.

A. Related Work

The optimization of UAV path planning, particularly in multi-objective optimization, has been an active area of research. This has led to the development of various hybrid algorithms combining metaheuristic techniques with machine learning to overcome challenges related to computational efficiency and robust multi-objective optimization.

The core advancement of our proposed DRL-ACO method remains its dual-integration of DRL to both generate a high-quality initial path set and, most critically, to learn the optimal, adaptive pheromone update value [15]. This comprehensive and deep integration contrasts sharply with other adaptive and hybrid ACO variants [16]. For instance, in the domain of adaptive parameter tuning, methods such as PF3SACO leverage classical techniques such as Particle Swarm Optimization (PSO) and Fuzzy Logic to adjust ACO parameters such as 1α and 2ρ.3 While adaptive, these methods rely on predefined metaheuristic interactions or empirical, rule-based inference systems. Our approach, by contrast, uses DRL to learn the optimal pheromone update directly from the search state, allowing the system to derive complex, non-linear control policies that are superior to any manually engineered or statically constrained adaptation mechanism [17], [18].

In contrast to recent hierarchical DRL-ACO models, such as ADACO-ATD3, which separate path planning (handled by ACO) and local motion control (managed by a dedicated DRL agent, such as ATD3 for real-time obstacle avoidance, this approach fundamentally optimizes the ACO engine itself [19]. In these hierarchical models, the ACO core remains static, with the DRL agent primarily refining local control mechanisms. In this approach, however, the DRL agent directly influences the global search process by dynamically controlling the pheromone update mechanism, an essential memory component of the ACO algorithm [20]. This DRL-driven adjustment significantly improves the global search quality and accelerates convergence by impacting the behavior of all subsequent ants across iterations [21].

Furthermore, when compared to other multi-objective DRL-ACO hybrids, such as NSACOWDRL which uses a Double Deep Q-Network (DDQN) agent for local search refinement or to modify ant selection probabilities, this DRL agent targets the pheromone update mechanism itself [14]. This allows our approach to function as a true meta-controller of the ACO's collective memory, rather than merely enhancing local path construction or solution selection heuristics. The ability to holistically and intelligently control the most influential parameter of the ACO, the pheromone update, is the key feature that enables superior and adaptively managed search performance [22].

In summary, the proposed DRL-ACO approach introduces a deeply integrated learning mechanism in which a DRL agent dynamically optimizes the pheromone update process, enhancing the global search capability and convergence efficiency of the ACO algorithm. Unlike previous adaptive or hierarchical hybrids, this approach functions as a true meta-controller of ACO's collective memory, enabling intelligent and adaptive management of its most influential parameter [23], [24]. Consequently, the framework achieves superior performance in multi-objective UAV path planning through a holistic and self-adaptive optimization strategy [25].

II. Methodology

UAVs are vital assets in surveillance, delivery, and mapping, especially in complex and dynamic environments. Efficient path planning for UAVs is a crucial, non-linear optimization challenge that requires balancing multiple, often conflicting, objectives, such as minimizing flight distance, mission time, and energy consumption. The DRL-APC methodology is structured into three phases: Problem Formulation, DRL-Assisted Initialization, and Adaptive ACO Optimization.This hybrid DRL-ACO approach, as shown in Fig. 1, aims to harness DRL learning and decision-making capabilities to provide a robust starting point for the ACO global search, thereby leading to more efficient convergence and higher-quality solutions for complex multi-objective UAV path planning problems.

A. Problem Formulation and Multi-Objective

1) Path representation: The mission space is modeled as a weighted graph $G = (V, E)$, where V are waypoints and E are possible flight segments. The UAV path P is a sequence of connected waypoints.

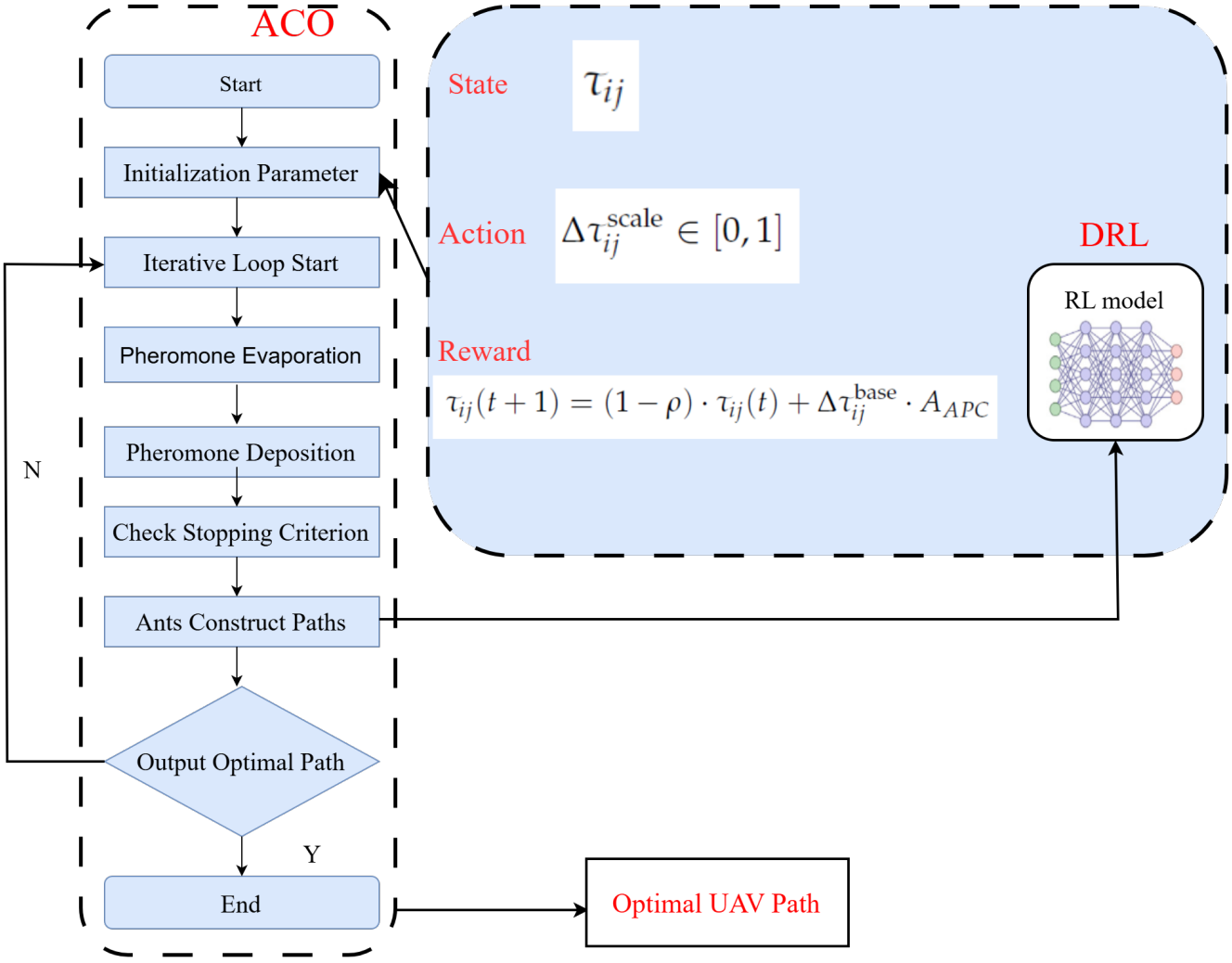


Fig. 1. Approach of the process for UAV path planning using RDL and ACO.

2) Multi-objective cost function C_{MOC} : The optimization objectives are normalized and combined using AHP-derived weights to form a single cost function for a path P:

$$C_{MOC}(P) = w_D \cdot C_D(P) + w_T \cdot C_T(P) + w_B \cdot C_B(P). \quad (1)$$

where,

C_D, C_T, C_B: Normalized costs (Distance, Time, Battery Consumption).

W_D, W_T, W_B: AHP-determined priority weights (summing to 1). The objective is to $\min C_{MOC}$.

B. Ant Colony Optimization (ACO)

ACO is a computational metaheuristic primarily employed for finding near-optimal solutions to discrete optimization problems, particularly those related to pathfinding. As a form of swarm intelligence, ACO is inspired by the cooperative foraging behavior observed in ant species.

The effectiveness of ant colonies in locating the shortest routes between their nests and food sources stem from their system of indirect communication, known as stigmergy. Ants deposit chemical signals, or pheromones, along their paths, which serve as odor trails. Subsequent ants follow these trails, using the scent concentration as a probabilistic guide to determine the most efficient route. Paths that are frequently traversed become reinforced with higher pheromone concentrations, thereby attracting more followers and establishing a positive feedback loop for optimal solutions.

The fundamental operation of the Ant Colony Algorithm can be broadly categorized into two iterative stages:

- Solution Construction: Artificial ants sequentially build solutions (a path) by making probabilistic choices. These choices are governed by the existing pheromone levels on the network edges and a local heuristic value.
- Pheromone Update: After a construction cycle, the pheromones along the utilized paths are modified. Pheromone levels are increased on components be-

longing to high-quality solutions (such as reinforcement) and simultaneously decreased across the entire network via evaporation to prevent premature convergence and encourage exploration. In essence, ACO harnesses the collective, decentralized decision-making of a swarm to select new path segments, assigning probabilities based on the accumulated, self-reinforcing knowledge encoded in the pheromone trails.

C. Deep Reinforcement Learning (DRL)

In response to the identified shortcomings of fixed meta-heuristic methods and the growing complexity of the optimization landscape, DRL presents a sophisticated and highly effective framework for complex decision-making. DRL is a powerful subset of machine learning that integrates the trial-and-error methodology of Reinforcement Learning (RL), where a computational agent learns an optimal policy through interaction with an environment to maximize cumulative reward—with the robust data processing capabilities of Deep Neural Networks.These networks enable the agent to effectively process and synthesize high-dimensional, complex state information, such as large graph structures and real-time sensor inputs, mapping these observations directly to an optimal action sequence.

DRL is particularly well-suited for solving the sequential decision-making and routing problems inherent in path planning. Furthermore, DRL is employed in a meta-learning capacity to enhance the adaptability of the optimization process itself. Instead of merely solving the primary path problem, a DRL agent is trained to learn the optimal control policies for the underlying search algorithm. This powerful capability allows the DRL agent to dynamically adjust internal ACO parameters, such as the pheromone update magnitude, based on the real-time feedback regarding environmental changes and the current state of the colony's search. This integration of DRL for both initial guidance and dynamic control is the key for creating a path planning framework that is both fast-converging and highly robust to environmental variability.

D. DRL-Assisted Pheromone Initialization

This phase addresses the fundamental drawback of traditional ACO: starting with a random or uniform search space. We utilize DRL to pre-train an agent that quickly learns optimal pathing strategies, which are then used to intelligently seed the ACO pheromone map. This represents a direct application of the DRL-initialization adapted for an ACO approach.

A Deep DQN is chosen for this task due to its effectiveness in solving sequence-selection problems TSP , which is analogous to our path-finding task. The objective of this training is to teach the agent the optimal action (next waypoint) to take from any given state to minimize the Multi-Objective Cost (C_{MOC}).

1) DRL Approach Setup:

- Agent: The DQN architecture typically consists of an input layer, several hidden layers, and an output layer equal to the size of the action space.
- Experience Replay Buffer: A mechanism is used to store and randomly sample past experiences $(\text{State}, \text{Action}, \text{Reward}, \text{Next State})$ to stabilize the learning process and break the correlation between consecutive samples.
- Target Network: A separate, delayed-update target network is used to compute the target Q-values, further enhancing stability during training.

E. State Space Definition (S)

The state S captures all necessary information for the agent to make an informed decision. For an agent at waypoint v_i, the state is defined as:

$$S = \{v_i, H_t, C_{t,objective}\} \tag{2}$$

where,

- Current Waypoint (v_i): The index of the UAV's current location.
- History Vector ($\mathbf{H}_t$): A binary vector representing all waypoints visited up to time t. This is crucial for preventing redundant visits and ensuring a complete path is generated.
- objectives Cost Vector ($\mathbf{C}_{t,\text{objective}}$): The aggregated normalized objectives costs (distance, time, battery).

F. Action Space Definition (A)

The action A is the agent's choice of the next waypoint v_j to move to from v_i:

$$A_t \in \{v_j \mid (v_i, v_j) \in E, and, v_j \notin Visitted\} \tag{3}$$

The action space is dynamic: the set of available actions changes as the path is constructed, always excluding waypoints already visited.

G. Reward Function (R)

The reward function is the critical link between the agent's actions and the optimization goal. Since the goal is to minimize the C_{MOC}, the reward is inversely proportional to this cost.

$$R(P) = \frac{R_{base}}{\epsilon + C_{MOC}(P)} \tag{4}$$

(where, ϵ is a small constant to prevent division by zero). A negative reward is given for illegal moves.

H. Initial Path Generation

The trained DQN is used to greedily generate N_{init} high-quality, initial paths: $P_{init} = \{P_1, P_2, \ldots, P_{N_{init}}\}$.

1) Pheromone map initialization: These paths are used to initialize the ACO pheromone matrix τ_{ij}. Instead of starting with uniform pheromones, the initial pheromone level $\tau_{ij}(0)$ on each edge (i, j) is set proportional to the inverse cost of the DRL-generated paths that use that edge:

$$\tau_{ij}(0) = \tau_{min} + \sum_{k=1}^{N_{init}} \mathbb{I}_{ij}(P_k) \cdot \frac{Q_{DRL}}{\sum_{m=1}^{N_{init}} C_{MOC}(P_m)} \quad (5)$$

where,

τ_{min} is the minimal pheromone level (evaporation lower bound).

$\mathbb{I}_{ij}(P_k)$ is an indicator function (1 if path P_k uses edge (i, j), 0 otherwise).

Q_{DRL} is a fixed constant representing the total initial pheromone mass.

2) Ant movement (path construction): Ants construct paths stochastically based on pheromone levels and heuristic information, as in traditional ACO.

$$P_{ij}^k = \frac{\tau_{ij}^{\alpha}.\eta_{ij}^{\beta}}{\sum_{l \in \text{Allowed}_i} \tau_{il}^{\alpha} \cdot \eta_{il}^{\beta}} \quad (6)$$

where,

τ_{ij} is the pheromone level on edge (i, j).

η_{ij} is the heuristic value (inverse of distance).

α and β are control parameters (importance of pheromone vs. heuristic).

3) DRL-Learned Adaptive Pheromone Update (APC): This is the core novelty, replacing fixed update rules with a learned policy. A separate, smaller DRL agent (a simple Q-Learning) is used to determine the optimal pheromone update magnitude.

The pheromone update rule becomes:

$$\tau_{ij}(t + 1) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij} \quad (7)$$

The new pheromone deposit $\Delta\tau_{ij}$ is determined by the DRL-APC Agent:

- APC Agent State (S_{APC}): A vector including the path segment (i, j) , the current global path quality, and the local pheromone density τ_{ij}.

$$S_{APC} = \tau_{ij} \quad (8)$$

- APC Agent Action (A_{APC}): The action is the determination of the pheromone scaling factor, $\Delta\tau_{ij}^{\text{scale}} \in [0, 1]$.

$$A_{APC} = \Delta\tau_{ij}^{\text{scale}} \in [0, 1] \quad (9)$$

- APC Agent Reward (R_{APC}): A reward function that encourages convergence and diversity: a high reward for finding a new global best solution, and a penalty for stagnation or over-prioritizing short-term, low-quality paths local optima.

The resulting pheromone update is:

$$\tau_{ij}(t + 1) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}^{\text{base}} \cdot A_{APC} \quad (10)$$

where,

$\Delta\tau_{ij}^{\text{base}}$ is the standard deposit for the best path, and A_{APC} is the scaling action learned by the DRL agent.

I. Path Output

The path calculation in this approach is a two-stage process: first, the probabilistic selection of discrete waypoints based on the environment's pheromone map, and second, the physical translation of those waypoints into a navigable 3D trajectory for the UAV.

- Probabilistic Movement Rule (Node Selection). The UAV (represented as an "ant" in the ACO framework) starts at a designated launch coordinate S. At each time step t, the agent must select the next node j from a set of reachable neighboring nodes N_i. This selection is governed by the transition probability P_{ij}, which balances exploration and exploitation.
- Multi-Objective Cost Constraints such as standard pathfinding, the UAV path calculation incorporates a cost function C derived via the Analytic Hierarchy Process (AHP). Every potential move is evaluated against:
  - Distance (C_D): Euclidean distance between nodes.
  - Time (C_T): Calculated based on the UAV's average velocity across the terrain.
  - Battery (C_B): Energy consumption model considering climb rate and air resistance.
- Path Refinement and DRL Influence. The "Deep Reinforcement Learning" component intervenes during the calculation by adjusting the pheromone scaling factor.

The final output is the path P_{best} corresponding to the lowest C_{MOC} found across all iterations.

J. Analytical Hierarchy Process

The AHP is one of the most widely applied techniques in multi-criteria decision analysis (MCDA), particularly within natural resource management and environmental studies. In this study, AHP is employed to determine the weights of the resource layers [26]. The method operates by assigning priority values based on relative importance, typically using a 1–9 scale [27]. Each criterion is evaluated individually through pairwise comparisons, after which the eigenvalues of the resulting comparison matrix are computed and utilized as the corresponding criterion weights.

This study follows the same methodological framework established in the referenced work by [16].

Table I Presents the objective weights assigned based on their relative importance in this approach.

TABLE I. THE OPTIMAL PATH

Objectives	D	T	B	Weight Score
Distance (D)	1	2	2	0.5
Time (T)	0.5	1	1	0.25
UAV battery (B)	0.5	0.5	1	0.25

During the training phase, the DRL agent learns optimal policies aimed at minimizing the AHP-weighted cost function, thereby generating paths that effectively balance the prioritized objectives. Consequently, the AHP framework not only provides objective weighting for the ACO fitness evaluation but also actively guides the DRL exploration process, ensuring that the initially generated paths capture mission-critical trade-offs prior to the ACO-based refinement stage.

III. STUDY AREA AND EXPERIMENTAL ANALYSIS

A. Study Area

The study area is located in the Red Sea Mountains in eastern Sudan, a striking geological formation that runs parallel to the Red Sea coast. Known for its rugged terrain and diverse ecosystems, the region offers a unique landscape shaped by the Arabian-Nubian Shield. The mountains are predominantly made up of granite and volcanic rock formations, with elevations ranging from 600 to over 2,500 meters above sea level. The ecological variety in this area includes dry forests, scrublands, and semi-desert habitats, providing a rich environment for a wide array of flora and fauna adapted to harsh conditions. Given the challenging terrain of the area, traditional methods of transportation between different points are inefficient and impractical. To overcome this issue, the present study employs a drone for logistical transport between the selected points. The specific locations of these points are provided in the accompanying Table II, while the layout of the terrain is depicted in Fig. 2.

B. Results

This study employs a multi-objective optimization approach to develop and determine the optimal UAV flight paths by integrating three key objectives: distance, time, and battery consumption. The process begins with the application of DRL to estimate a set of promising initial parameters for the optimization. These parameters are then refined through the ACO algorithm.

- ACO Parameters (α and β). Pheromone Influence ($\alpha = 1.0$). This value was chosen to ensure that the "swarm intelligence" (historical data) has a significant impact on path selection without causing premature convergence to suboptimal paths. Heuristic Influence ($\beta = 2.0$) we assigned a higher weight to the heuristic information (distance to goal). This is critical in UAV pathing to ensure that ants remain "goal-oriented".
- Evaporation Rate ($\rho = 0.1$). The evaporation rate was set to 0.1 to provide a "long memory" for the pheromone trails. In a complex 3D environment, rapid evaporation (> 0.5) can lead to the loss of good global solutions, while too low a rate (< 0.05) prevents the algorithm from discarding poor paths.
- Learning Rate ($\gamma = 0.001$). A conservative learning rate was selected for the Deep Q-Network (DQN) to ensure stable convergence of the pheromone scaling factor.
- Discount Factor ($\gamma = 0.95$). This high value ensures the DRL agent considers the long-term cost of the entire path (Total Battery/Time) rather than just the immediate next step.
- Exploration-Exploitation (ϵ-greedy). Utilized a decaying epsilon starting at 1.0 and ending at 0.01. This allows the DRL agent to explore a wide variety of pheromone update strategies in early iterations before settling on an optimized policy.

The effectiveness of the proposed DRL-ACO hybrid depends on the calibration of several key hyperparameters. These values were selected based on a combination of established literature and empirical tuning during the preliminary simulation phase.

In this context, the DRL agent functions as an intelligent control system, continuously tasked with identifying the optimal pheromone update strategy tailored to the specific mission objectives. This intelligent control mechanism is crucial for determining the optimal flight trajectory by balancing the mission's multi-objective criteria. To systematically prioritize and combine the three objectives (distance, time, and battery consumption), the AHP is employed. The AHP ensures a balanced optimization process by assigning relative importance to each objective based on mission-specific constraints and application needs.

Following the weight assignment and construction of the multi-objective decision matrix, an initial population of candidate solutions is generated through DRL. This approach enhances the performance of ACO by providing an optimal pheromone update strategy value, set at 200, which improves the search efficiency.

Finally, the optimal UAV path is determined by applying the ACO, which has been enhanced through integration with the multi-objective DRL framework. The DRL component guides the initialization and search process by providing a more informed starting population, allowing the ACO to converge more effectively toward solutions that balance the three objectives: minimizing travel distance, energy consumption, and mission time.

The results demonstrate the superiority of our proposed algorithm in determining the optimal logistic trajectory length for the UAV, which was found to be 750 meters. In contrast, alternative algorithms aimed for trajectory lengths of 800 and 850 meters. This outcome underscores the reliability of our approach, as evidenced in Table III and Fig. 4.

Fig. 4 shows the optimal path using our approach, and Fig. 3 shows the path using traditional ACO and GA.

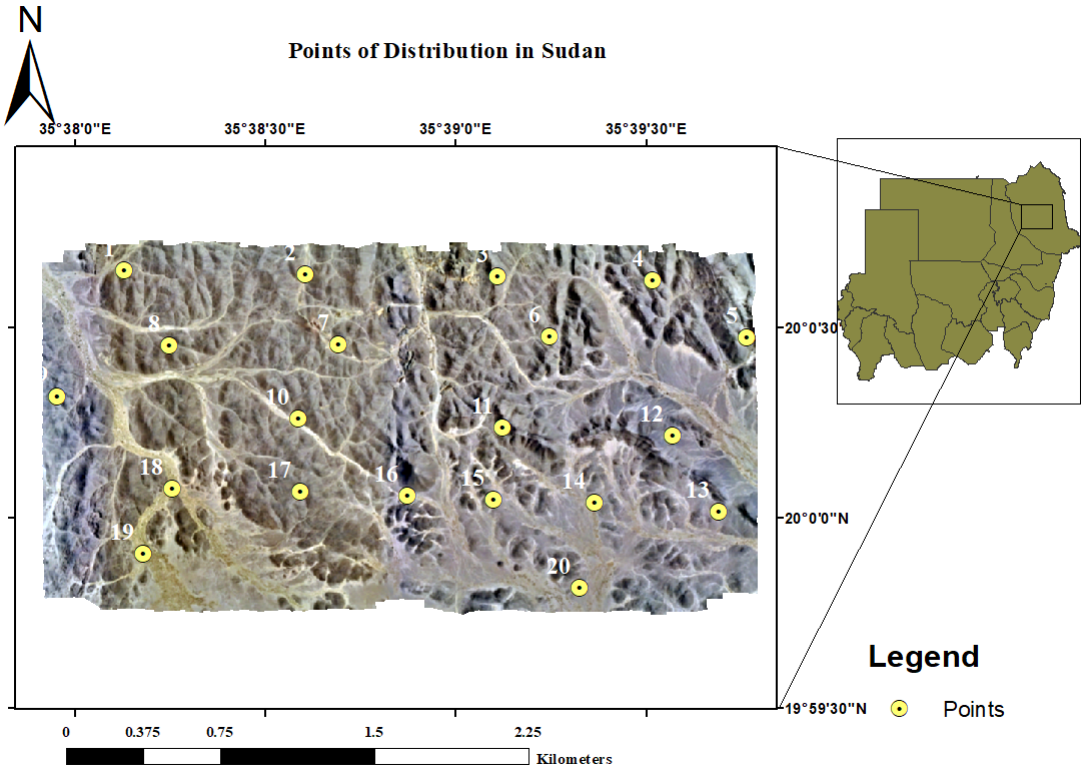


Fig. 2. Points of distribution in Sudan.

TABLE II. POINTS SELECTED FOR OPTIMAL PATH IMPLEMENTATION USING THE UAV

NO	X	Y	H	NOTE
1	777724	2213582	567.89	P1
2	778531	2213530	560.30	P2
3	777866	2214318	570.12	P3
4	778447	2214406	568.81	P4
5	775792	2213514	559.35	P5
6	775778	2214369	555.72	P6
7	776970	2214391	573.14	P7
8	778181	2215498	573.32	P8
9	777720	2216367	562.56	P9
10	776577	2216587	556.08	P10
11	776284	2216169	561.366	P11
12	776111	2214297	557.59	P12
13	776127	2213926	567.58	P13
14	776477	2214631	572.83	P14
15	776570	2215327	565.57	P15
16	775662	2216103	559.32	P16
17	776869	2217447	556.15	P17
18	776484	2218677	553.27	P18
19	776418	2219481	537.95	P19
20	775577	2220404	543.84	P20

C. Discussion

The integration of DRL with ACO significantly enhances UAV path planning by improving convergence speed and optimizing multi-objective criteria, such as distance, time, and energy consumption. The DRL-ACO framework not only overcomes the limitations of traditional ACO, which relies on static pheromone update rules, but also provides adaptability through dynamic pheromone management, leading to faster convergence towards optimal solutions [28], [18]. In the experimental results, the DRL-ACO approach demonstrated superior performance, finding shorter paths 750 meters compared to traditional ACO and GA 850 meters, thus ensuring greater efficiency in UAV operations. The use of the AHP further allowed the DRL agent to balance the conflicting objectives by assigning appropriate weights, ensuring that the solution met mission-specific requirements effectively [23].

This approach proved to be particularly robust in a complex environment, as demonstrated in the study area in Sudan, where the challenging terrain required adaptability in path planning. Unlike traditional approaches, the DRL-ACO approach could handle the unpredictability of real-world conditions, making it highly suitable for mission-critical applica-

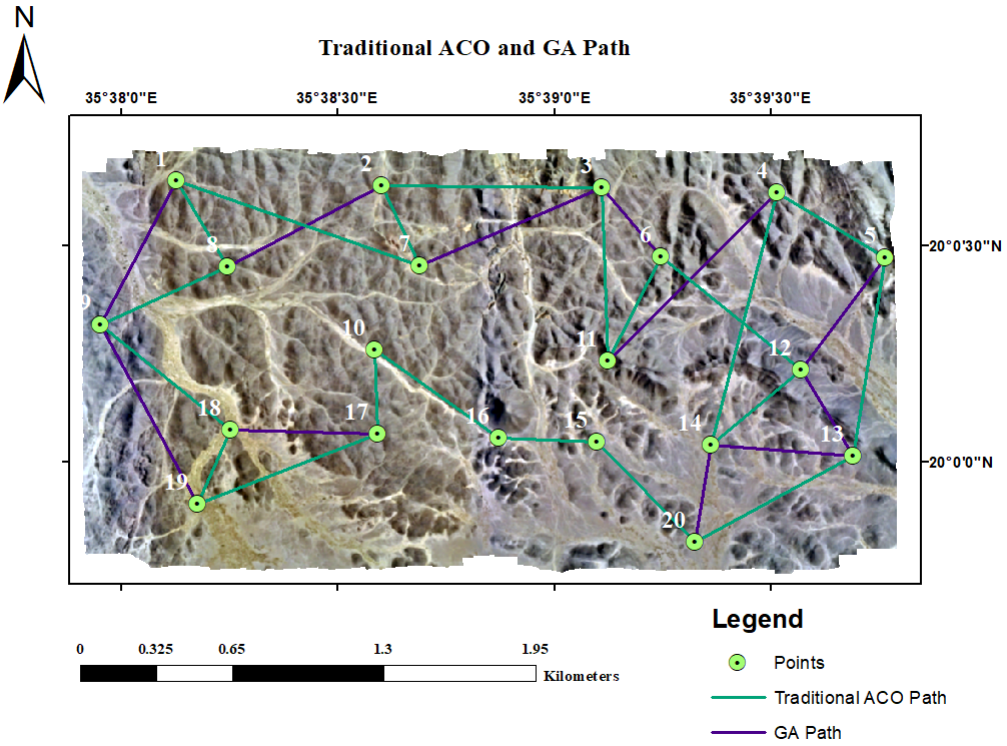


Fig. 3. Traditional ACO and GA path.

TABLE III. OPTIMAL LOGISTIC TRAJECTORY LENGTH

NO	Name	Points	Length
1	DRL-ACO Path	1→8→ 7→2→11→3→6→14→4→5→12→20→13→15→16→10→17→19→9→18→1	750
2	ACO Path	9→8→ 1→7→2→5→11→6→12→14→4→5→13→20→15→16→10→17→19→18→9	850
3	GA Path	17→18→ 19→9→1→8→2→7→3→6→11→4→5→12→13→14→20→15→16→10→17	850

TABLE IV. COMPARISON BETWEEN DRL-ACO AND TRADITIONAL ACO AND GA

NO	Length(m)	Time Implementation(s)
DRL-ACO	750	0.27
Traditional ACO	800	0.35
GA	850	0.55

tions, such as disaster response and surveillance. While the framework showed significant improvements, there are still opportunities for future enhancement, particularly in incorporating additional factors like obstacle avoidance or real-time re-planning, and improving computational efficiency. Despite its success, the DRL-ACO approach requires substantial computational resources for training, suggesting that further optimization of the DRL model could improve its applicability in real-time scenarios.

In comparison to other path planning algorithms, the DRL-ACO approach demonstrated clear advantages in reducing path length and optimizing mission objectives. This outcome not only highlights the potential of the framework for real-world applications but also sets a foundation for future advancements, where the system can adapt to even more complex and dynamic environments, improving its effectiveness in various UAV mission scenarios.

The proposed algorithm was tested against two established methods: the traditional ACO and the GA algorithm. The results of this comparison are presented in Table IV, assessed performance based on three key metrics: path length 750 m, and execution time 0.27 s. The results demonstrate that the proposed algorithm consistently outperforms both the ACO and GA, achieving more optimal path lengths in fewer iterations and with significantly lower computational overhead.

IV. CONCLUSION

This study successfully introduced a novel DRL-Assisted Ant Colony Optimization (DRL-ACO) approach to address the limitations of traditional ACO in multi-objective UAV path planning. The critical innovation involves the deep integration of a DRL agent to serve as an intelligent control system, replacing static rules with an adaptive pheromone update (APC) mechanism. The problem was formalized using the AHP to

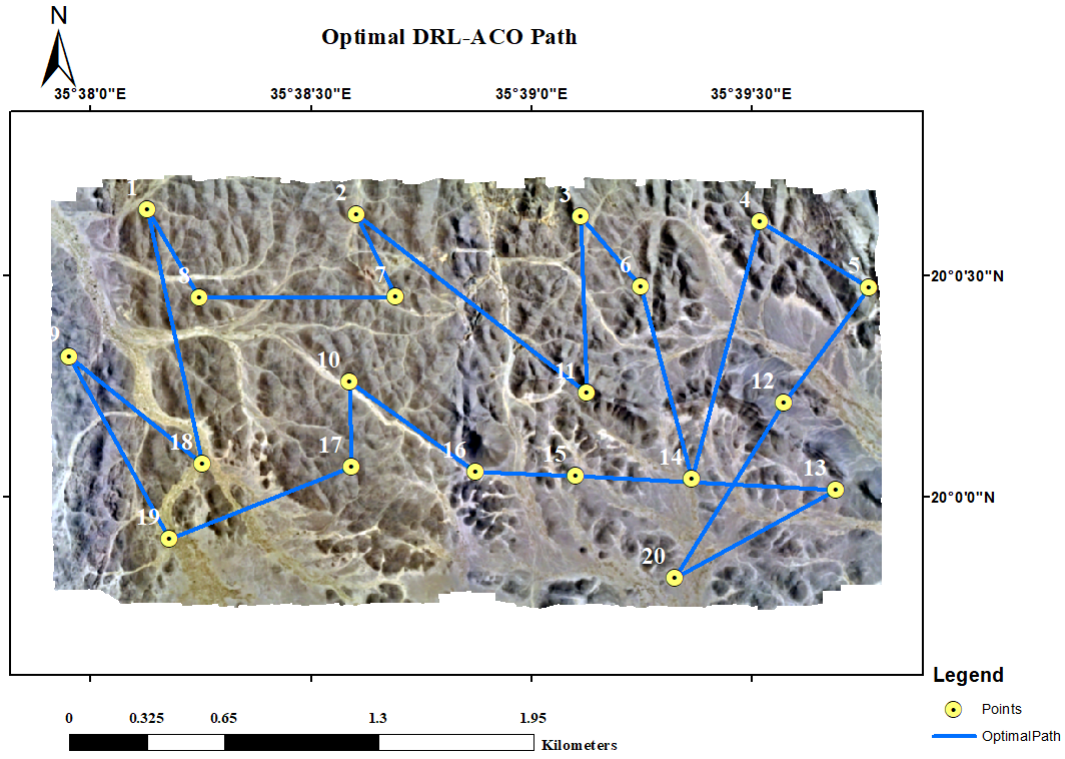


Fig. 4. Optimal DRL-ACO path.

systematically weigh and combine the conflicting objectives of minimizing flight distance, energy consumption, and mission time into a single multi-objective cost function (C_{MOC}). Empirical validation, conducted in a challenging, rugged terrain environment in Sudan, confirmed the framework's superiority. The DRL-ACO approach found the optimal trajectory length of 750 meters, significantly outperforming the 800-850 meters found by traditional ACO and GA, while also demonstrating a lower computational time of 0.27 seconds. This outcome confirms that the DRL-ACO framework provides a robust and adaptive solution that achieves faster convergence and superior optimization of multi-objective criteria for autonomous UAV operations.

Despite the demonstrated success, the DRL-ACO framework presents several opportunities for future enhancement to increase its real-time applicability. Firstly, future research should focus on incorporating additional real-time constraints, such as obstacle avoidance and dynamic path re-planning, to enhance the system's robustness in highly unpredictable environments. Secondly, the current approach requires substantial computational resources for DRL model training. Finally, researchers could explore more advanced DRL algorithms or hierarchical DRL structures to manage local control mechanisms, potentially yielding even more sophisticated, non-linear control policies for the adaptive ACO parameters.

FUNDING

This study was funded by the Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, Saudi Arabia, under grant no (GPIP:882-830-2024). The authors, therefore, gratefully acknowledge and thank DSR for its technical and financial support.

AUTHORS' CONTRIBUTIONS

Mohamed A. Damos: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Visualization. Wenbo Xu: Methodology (Genetic Algorithm design), Software, Validation, Writing – review editing. Ali Ahmed: Methodology (AHP integration), Validation, Writing – review. Funding acquisition, Resources, Project administration .Abdolraheem Khader: Software (simulation environment), Validation, Formal analy- sis. Data curation, Visualization, Writing – review editing.

AVAILABILITY OF DATA AND MATERIALS

The data and materials supporting the findings of this study are available upon request from the corresponding author. Simulation code, algorithm parameters, and waypoint coordinates used for UAV path planning are fully shareable.

ACKNOWLEDGMENT

The authors gratefully acknowledge the Deanship of Scientific Research at King Abdulaziz University, Jeddah, for its technical and financial support under Grant No. (GPIP:882-830-2024).

REFERENCES

[1] H. Hildmann and E. Kovacs, "Using unmanned aerial vehicles (uavs) as mobile sensing platforms (msps) for disaster response, civil security and public safety," *Drones*, vol. 3, no. 3, p. 59, 2019.

[2] M. Damos, J. Zhu, W. Li, A. Hassan, and E. Khalifa, "A novel urban tourism path planning approach based on a multiobjective genetic algorithm. isprs int., geo-inf 2021, 10.530 https," *doi. org/https://doi. org/10.3390/ijgi10080530*, 2021.

[3] X. Zhang, X. Yu, and X. Chen, "D-aco-based path planning for auv in uwsn: A hybrid approach combining dqn and aco," *IEEE Sensors Journal*, 2025.

[4] R. Priyadarshi and R. R. Kumar, "Evolution of swarm intelligence: a systematic review of particle swarm and ant colony optimization approaches in modern research," *Archives of Computational Methods in Engineering*, pp. 1–42, 2025.

[5] M. A. Damos, W. Xu, J. Zhu, A. Ahmed, and A. Khader, "An efficient tourism path approach based on improved ant colony optimization in hilly areas," *ISPRS International Journal of Geo-Information*, vol. 14, no. 1, p. 34, 2025.

[6] C. Chen, L. Cao, Y. Chen, B. Chen, and Y. Yue, "A comprehensive survey of convergence analysis of beetle antennae search algorithm and its applications," *Artificial Intelligence Review*, vol. 57, no. 6, p. 141, 2024.

[7] R. S. Othman and I. M. Ibrahim, "A review of exploring recent advances in ant colony optimiza-tion: applications and improvements."

[8] P. Li, J. Hao, H. Tang, X. Fu, Y. Zhen, and K. Tang, "Bridging evolutionary algorithms and reinforcement learning: A comprehensive survey on hybrid algorithms," *IEEE Transactions on evolutionary computation*, 2024.

[9] S. Nimmala, M. Ramchander, M. Mahendar, P. Manasa, D. D. Bhavani, and K. Raghavendar, "Dynamic rl-aco: Reinforcement learning-based ant colony optimization for load balancing in cloud networks," in *2024 5th International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE, 2024, pp. 475–480.

[10] B. Jin, Y. Sun, W. Wu, Q. Gao, and P. Si, "Deep reinforcement learning and ant colony optimization supporting multi-ugv path planning and task assignment in 3d environments," *IET Intelligent Transport Systems*, vol. 18, no. 9, pp. 1652–1664, 2024.

[11] Y. Caihong, "Dynamic logistics path optimization via integrated ant colony optimization and reinforcement learning," *Informatica*, vol. 49, no. 6, 2025.

[12] A. Ahmed, H. Ju, Y. Yang, and H. Xu, "An improved unit quaternion for attitude alignment and inverse kinematic solution of the robot arm wrist," *Machines*, vol. 11, no. 7, p. 669, 2023.

[13] Y. Wang, J. Liu, Y. Qian, and W. Yi, "Path planning for multi-uav in a complex environment based on reinforcement-learning-driven continuous ant colony optimization," *Drones*, vol. 9, no. 9, p. 638, 2025.

[14] Z. Guo, Y. Xia, J. Liu, J. Gao, P. Wan, and K. Xu, "Path planning design and experiment for a recirculating aquaculture agv based on hybrid nrbo-aco with dueling dqn," *Drones*, vol. 9, no. 7, p. 476, 2025.

[15] M. M. Hamdi, B. S. Abdulhakeem, and A. A. Nafea, "Psoa-crl: A hybrid multi-objective routing mechanism using particle swarm optimization and actor-critic reinforcement learning for vanets," *Mesopotamian Journal of Big Data*, vol. 2025, pp. 241–260, 2025.

[16] M. A. Damos, J. Zhu, W. Li, E. Khalifa, A. Hassan, R. Elhabob, A. Hm, and E. Ei, "Enhancing the k-means algorithm through a genetic algorithm based on survey and social media tourism objectives for tourism path recommendations," *ISPRS International Journal of Geo-Information*, vol. 13, no. 2, p. 40, 2024.

[17] X. Zhou, H. Ma, J. Gu, H. Chen, and W. Deng, "Parameter adaptation-based ant colony optimization with dynamic hybrid mechanism," *Engineering Applications of Artificial Intelligence*, vol. 114, p. 105139, 2022.

[18] B. Wang, D.-T. Duan, Q. Yang, X.-Y. Zhao, T. Li, D. Liu, and J. Zhang, "Adaptive ant selection for pheromone update in ant colony optimization," in *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2024, pp. 667–672.

[19] A. Gharbi, "A dynamic reward-enhanced q-learning approach for efficient path planning and obstacle avoidance in mobile robotics," *Applied Computing and Informatics*, 2024.

[20] S. Goyal, L. K. Awasthi, V. Garg, and G. Kumar, "Multi-resource aware virtual machine consolidation approach for modern cloud data centers," *Computing*, vol. 107, no. 11, pp. 1–32, 2025.

[21] A. Sezgin and A. Boyacı, "Optimizing drone deployment for network coverage using a hybrid ant colony optimization and reinforcement learning approach: Karınca kolonisi optimizasyonu ve pekiştirmeli öğrenme yaklaşımı kullanarak ağ kapsaması için drone konuşlandırmasının optimizasyonu," *Journal of Aeronautics and Space Technologies*, vol. 18, no. 2, pp. 268–285, 2025.

[22] J. Zhang, H. Xu, D. Liu, and Q. Yu, "Advancing dynamic emergency route optimization with a composite network deep reinforcement learning model." *Systems*, vol. 13, no. 2, 2025.

[23] S. Fang, Z. Deng, P. Li, and D. Long, "Improved strategy of ant colony optimization for path planning via stochastic pheromone updating and cyclic initialization," *Journal of Mechanical Science and Technology*, pp. 1–12, 2025.

[24] G. Vachtsevanos, L. Tang, G. Drozeski, and L. Gutierrez, "From mission planning to flight control of unmanned aerial vehicles: Strategies and implementation tools," *Annual Reviews in Control*, vol. 29, no. 1, pp. 101–115, 2005.

[25] X. Li, Z. Ruan, Y. Ou, D. Ban, Y. Sun, T. Qin, and Y. Cai, "Adaptive deep ant colony optimization–asymmetric strategy network twin delayed deep deterministic policy gradient algorithm: Path planning for mobile robots in dynamic environments," *Electronics*, vol. 13, no. 20, p. 4071, 2024.

[26] R. Kumar and R. Anbalagan, "Landslide susceptibility mapping using analytical hierarchy process (ahp) in tehri reservoir rim region, uttarakhand," *Journal of the Geological Society of India*, vol. 87, pp. 271–286, 2016.

[27] Y. Wind and T. L. Saaty, "Marketing applications of the analytic hierarchy process," *Management science*, vol. 26, no. 7, pp. 641–658, 1980.

[28] B. Abdulghani and M. Abdulghani, "A comprehensive review of ant colony optimization in swarm intelligence for complex problem solving," *Acadlore Transactions on Machine Learning*, vol. 3, no. 4, pp. 214–224, 2024.

A Context-Aware Hybrid Recommendation Framework for E-Learning Platforms

Kaoutar Errakha[1], Amina Samih[2], Sanaa Dfouf[3], Abderrahim Marzouk[4]
Doctor of Computer Science and Artificial Intelligence-Faculty of Sciences and Techniques,
University Hassan First, Settat, Morocco[1]
Artificial Intelligence Research and Applications Laboratory, University Hassan First, Settat, Morocco[2]
Doctor of Economic and Management Sciences, Faculty of Legal Economic and Social Sciences, Mohammedia, Morocco[3]
Teacher Researcher-Faculty of Sciences and Techniques, University Hassan First, IR2M, Morocco[4]

***Abstract*—E-learning platforms provide learners with extensive digital resources that enable self-paced and location-independent study. However, the overwhelming volume of learning materials offered by a wide range of institutions and content providers makes personalized guidance increasingly essential for effective knowledge acquisition. As a result, recommender systems have become fundamental components of modern e-learning environments, helping to reduce information overload and support individualized learning experiences. In general, the richer and more diverse the available data, the more accurate and relevant the resulting recommendations. Despite these advantages, conventional recommendation approaches often fail to fully exploit the contextual and relational information inherent in e-learning ecosystems, which limits their adaptability and predictive precision. This study proposes a hybrid recommendation framework that integrates collaborative filtering, content-based filtering, and context-aware modeling to generate more accurate and adaptive course recommendations. The proposed system infers learner preferences by combining historical interaction data, contextual attributes, and course characteristics, while also incorporating temporal and environmental factors that influence learning behavior. Experimental evaluations based on SVD, TF-IDF, and RNN models applied to a well-established benchmark dataset demonstrate that the proposed hybrid framework significantly improves recommendation accuracy, coverage, and adaptability compared with baseline methods. Furthermore, the integration of contextual information effectively alleviates the cold-start problem and better captures learners' evolving goals and learning trajectories. Overall, the results confirm that combining multiple recommendation paradigms within e-learning platforms enables more adaptive, personalized, and scalable learning pathways, making the proposed system suitable for diverse educational contexts and learner profiles.**

***Keywords*—E-learning; recommender systems; hybrid approach; personalized learning; course**

I. Introduction

The flexibility of e-learning systems has fundamentally transformed the way students learn and how education is delivered. These platforms enable self-directed and personalized learning experiences, allowing learners to acquire new skills and knowledge beyond the traditional classroom. The COVID-19 pandemic further highlighted the critical role of e-learning technologies, as they were rapidly adopted to ensure educational continuity during periods of disruption. However, modern e-learning platforms offer an enormous number of courses and digital learning resources [1]. This abundance often leads to information overload, making it difficult for learners to identify and select content that aligns with their academic objectives, learning styles, and personal preferences.

In this context, recommender systems (RSs) play a crucial role in helping learners discover relevant educational resources from large collections of courses [2][3]. In domains such as e-commerce, entertainment, and social media, recommender systems based on Collaborative Filtering (CF) and Content-Based Filtering (CBF) have demonstrated strong performance. CF predicts user interests by identifying similarities among users and items using interaction histories such as ratings, clicks, and browsing behavior [4], [5]. In contrast, CBF recommends items by analyzing content features, including course descriptions, topics, and keywords, and matching them to a user's previous preferences [6], [7]. This approach relies on comparing new items with those previously consumed to determine suitable recommendations.

Despite their success, these techniques exhibit important limitations when applied to e-learning environments. CF suffers from the cold-start problem and data sparsity, particularly when new users or new courses have limited interaction history [8], [9]. CBF, on the other hand, relies on static content representations and therefore struggles to adapt to learners' evolving goals and situational learning contexts [10].

To overcome these limitations, recent research has focused on hybrid recommender systems that combine multiple recommendation strategies to exploit their complementary strengths. Several studies [11], [12] have shown that incorporating additional information especially contextual data—significantly enhances recommendation performance. In e-learning environments, contextual factors such as the device used, time of access, location, and session duration strongly influence learners' behavior and course preferences. For example, a learner accessing the platform from a mobile device may prefer short tutorials, whereas the same learner using a desktop computer at home may be more inclined to engage in longer and more comprehensive learning modules [13].

The present work proposes a hybrid e-learning recommender system that integrates Collaborative Filtering, Content-Based Filtering, and contextual modeling to provide personalized and context-aware course recommendations. Contextual information is modeled using predictive LSTM neural networks to capture temporal and behavioral patterns, while SVD is employed for matrix factorization and TF-IDF for textual feature extraction and natural language processing.

This study makes several important contributions. First, it introduces a hybrid recommendation framework that jointly leverages collaborative, content-based, and contextual modeling techniques to enhance personalization in e-learning environments. Second, it integrates SVD, TF-IDF, and LSTM to address key challenges such as data sparsity, cold-start issues, and evolving learner behavior. Third, the framework is extensively evaluated using real edX course data, demonstrating superior performance over traditional CF and CBF methods in terms of accuracy, relevance, and contextual adaptability.

The remainder of this study is organized as follows: Section II reviews related work on hybrid recommender systems and discusses the limitations of existing approaches. Section III presents the proposed framework and methodology. Section IV describes the experimental setup, including evaluation metrics and results. Finally, Section V concludes the study and outlines directions for future research.

II. Related Work

With the use of recommender systems, user experiences across many areas, including e-learning, have been personalized to individual user needs [14], [15]. These systems suggest activities based on users' previous activities [16], [17]. The first recommender systems relied on Collaborative Filtering (CF) and Content-Based Filtering (CBF). CF analyzes user-item interactions based on attribution, clicks, and behavior to find patterns of similarity among users or items [17]. CF is effective in many scenarios, but sparse data, new users, and the absence of interaction data (cold-start) is highly problematic. Other techniques, such as predictive analytics, have been proposed to solve these issues, but CF continues to have difficulty adapting to changing environments and user preferences.

CBF improves on CF by using items features (keywords, topics, and metadata). In e-learning, CBF analyzes course materials to recommend resources that adjust to learner preferences based on their previous activities [15]. CBF's major problem is its over-specialization. It limits user exploration by not encouraging them to search outside the narrowly defined parameters of their needs [14]. Additionally, its use of static features fails to capture the user's evolving needs [16], [18].

Hybrid recommender systems integrate the strengths of Collaborative Filtering (CF) and Content Based Filtering (CBF) techniques to provide the users of e-learning platforms engaged and more targeted suggestions [19]. The addition of deep learning methods, such as neural collaborative filtering, which provides the ability to work on and analyse intricate relationships and non-linear user-item interactions, is a recent and notable advancement in this area [20].

Although hybrid systems have progressed in many ways, they still have a critical oversight of failing to integrate contextual data which includes how, when, and where a user interacts with the learning materials [21][22]. CARS approaches this oversight with contextual data such as study times, types of devices, locations, and other time constraints for learning materials [18]. For instance, recommending longer courses for uninterrupted study times and shorter tutorials for commuting is likely to improve learning and the perceived value of the recommendations. Including context data in recommendations has been shown to improve accuracy and user satisfaction [19].

Nonetheless, many CARS approaches for context modeling remain rudimentary or involve only fragmentary contextual details, leading to unfulfilled personalization. Moreover, context modeling intricacy might lead to scalability issues and pose problems for real-time adaptation which, in turn, limits responsiveness in swiftly evolving learning settings, as noted in [23]. This is why CF, CBF, hybrid and context-aware models still default most of the time in handling the interplay of collaborative, content, and contextual pieces. In addition, they tend to be responsive inflexibly to learner's situational alterations, state of cold start, and real-time adjustments of behavior or situational aspects of the environment [22].

In light of these challenges, this research aims to develop a hybrid e-learning recommender system that combines elements of collaboration, content, and contextual strategies within a single framework. This system develops sophisticated SVD models for matrix factorization, employs TF-IDF for feature embedding, and uses RNNs for sequential behavior modeling. By dealing with user-specific preferences and shifting contextual factors, the integrated approach improves the precision, diversity, and contextual relevance of recommendations. On a wider frontier, this fully personalized system addresses scalability and adaptive challenges while providing personalization within the learning pathways for a diverse learner's spectrum and within a robust and dynamic real-world e-learning context, as detailed in [21, 24].

Proposed system considers the issues in previous works and focuses on providing an adaptive solution that offers flexibility and overcomes the many restrictions traditional e-learning recommender systems face. For instance, cold-start, data sparsity, overspecialization, and an adaptive approach towards changing user preferences are stick points for Collaborative Filtering (CF) and Content-Based Filtering (CBF) systems. Although hybrid approaches try to associate CF and CBF, many fail to consider the user context while learner interactions and user preferences are adjusted. The proposed system overcomes this shortcoming by providing context-aware integration of-collaborative, content, and CBF approaches. Thus, achieving a comprehensive, synergistic recommendation framework.

One of the strengths of this system is the management of user behavior over time. User behavior on e-learning platforms is not fixed; it changes as users learn new topics, reorganize their study plans, or adapt to other external factors like the availability of devices and location. Fleeting and situational conditions allow the system to revise recommendations in real time, attuned to the user's present needs, offering resources that are relevant and actionable. The ability to track changes user behavior over time and in real time is a significant improvement over traditional CF and CBF, which tend to provide users with recommendations that are the same over extended periods and are based on stale historical behavior patterns or unchanging attributes of the content.

In addition, system parameters that analyze the user's context, including limitations and learning goals, are vital to personalization in e-learning. Session length, time of access, and device type are examples of context variables that help to adapt recommendations to user learning environments. In addition, learning goals help the system to prioritize targets aligned with users–whether for skill building, exams, or professional learning. Context and goal alignment together ensure that the

system moves beyond offering vague, unhelpful recommendations to the user and supports actionable recommendations.

Overall, the proposed hybrid system is a significant improvement over the previous work which is focused on e-learning recommendation systems. It closes the gaps between the collaborative, content-based, and context-approaches, facilitating the generation of adaptive, flexible, and personalized learning paths. This not only contributes to higher recommendation accuracy and user satisfaction, but also to the advancement of intelligent context-aware e-learning systems aimed at meeting the needs of a variety of learners in real-life educational environments.

Recent research in e-learning recommendation systems has shown a clear trend toward hybrid and context-aware approaches, which combine multiple techniques to improve personalization and system adaptability in response to the diversity of learners and learning contexts. For instance, hybrid approaches integrating user and content-based techniques with collaborative methods have been proposed to address the cold-start problem in personalized course recommendation systems, demonstrating increased user satisfaction without relying solely on historical user ratings [25]. Similarly, context-aware recommendation engines integrating hybrid reinforcement learning have been developed to dynamically adapt recommended content according to learner profiles, preferences, and real-time performance, illustrating the integration of various contextual data such as learning pace and individual trajectories [26]. Moreover, systematic reviews of recent advances in context-aware recommender systems highlight the importance of structured contextual representations and integrated methods to capture complex interactions among users, items, and context, while emphasizing the need for more unified frameworks for context management [27] [28]. Other studies explore group or multi-criteria recommendation mechanisms, showing the addition of diverse contextual factors and dynamic attention mechanisms to better serve collaborative or multi-dimensional educational environments [29] [30]. Federated hybrid approaches have also been proposed to address challenges related to inter-institutional diversity and data privacy, constructing heterogeneous representations that integrate historical and contextual interactions within distributed learning graphs [27] [28]. Finally, research on hybrid and context-aware recommendations in adjacent domains demonstrate the positive impact of combining multiple paradigms on recommendation accuracy and adaptability, providing a solid conceptual foundation for hybrid context-aware frameworks in e-learning [31].

III. Proposed Framework and Methodology

We aim to enhance personalization and flexibility in e-learning recommendations by designing a hybrid recommendation framework that integrates Collaborative Filtering (CF), Content-Based Filtering (CBF), and Contextual Data Modeling (CDM) using Long Short-Term Memory (LSTM) networks. By combining these three approaches, the framework overcomes the limitations of relying on a single recommendation technique and enables more accurate, diverse, and context-aware learning recommendations.

Collaborative Filtering (CF) identifies users and groups with similar preference patterns, behaviors, and interaction histories. It analyzes user–item interaction matrices and recommends courses that have been highly rated or frequently accessed by users with similar profiles. However, CF suffers from data sparsity and cold-start problems, particularly when dealing with new users or newly introduced courses.

Content-Based Filtering (CBF) relies on course attributes such as keywords, topics, descriptions, and metadata to match learning resources with individual learner profiles. This approach is effective in recommending courses aligned with a learner's past interests and completed materials. Nevertheless, CBF often leads to over-specialization, as it tends to recommend content that is too similar to previous selections, thereby limiting exploration and discovery of new topics.

Contextual Data Models capture situational factors that influence learning behavior, including temporal variables (e.g., study schedules and session duration) and spatial or environmental attributes such as location, device type, and learning context. To model these complex and sequential relationships, we employ Long Short-Term Memory (LSTM) networks, a class of recurrent neural networks capable of learning long-term dependencies in user behavior. Through LSTM, the system analyzes how learners' preferences and contexts evolve over time and predicts which courses are most suitable at a given moment. For instance, the system may recommend short tutorials during commuting or brief study sessions, while suggesting more comprehensive modules during extended learning periods.

Each component—CF, CBF, and the LSTM-based contextual model—operates independently to generate candidate recommendations. These outputs are then combined through a fusion mechanism that considers user reliability, behavioral profiles, and contextual relevance. This hybrid aggregation strategy provides dynamic recommendation capabilities, allowing the system to adapt to changes in learner preferences and environmental conditions in real time.

By integrating collaborative, content-based, and contextual modeling approaches, the proposed framework offers a flexible, scalable, and context-sensitive solution for e-learning personalization. It simultaneously exploits historical interaction patterns, course attributes, and situational factors to deliver highly relevant and adaptive recommendations, ultimately enhancing learner engagement, effectiveness, and satisfaction.

A. Framework Overview

Fig. 1 will help to explain the elements of the blended recommendation framework. There are four blended recommendation framework parts that are unified, which help to generate hyper-personalized course recommendations. The modular approach also allows the recommendation system to gain different facets of a learner's behavior, a course's content, and contexts, leading to comprehensive recommendations. Since diverse aspects of learner behavior and content are accounted for, the recommendations are comprehensive and accurate.

1) Collaborative Filtering (CF) with SVD (User-Based Collaborative Filtering – UBCF): The first segment uses the previously discussed user-item historical interaction data of ratings, clicks, or engagement trails to locate and analyze user similarities. SVD (Singular Value Decomposition) transforms

the user-item interaction matrix by decomposing it into a set of latent factors that the system can use to uncover hidden user preferences and help outline custom course recommendations by analyzing the adjusted suggestions of comparable users. The UBCF (User-Based Collaborative Filtering) approach, which focuses on learners with comparable profiles, also improves the relevance of recommendations. Latent factor modeling UBCF also solves some of the sparsity issues.

2) Content-Based Filtering (CBF) with TF-IDF: The second module analyzes each course and matches it with user profiles. Using TF-IDF (Term Frequency-Inverse Document Frequency) algorithm, the system analyzes course descriptions, keywords and other textual features and metadata, and assesses the relevance of each course in relation to the learner's prior choices. This component guarantees that the recommendations correspond to the learner's interests and areas of knowledge. In contrast to CF, CBF can algorithmically recommend new courses, thus alleviating cold-start challenges for new items.

3) Contextual modeling with LSTM networks: Incorporating the third constituent, the bloc incorporates contextual information that makes it possible to understand the temporal and situational aspects of user behavior. Using Recurrent Neural Networks (RNNs) with Long Short Term Memory (LSTM) units, it tracks the sequence of data in learner interactions and analyzes the timestamps, devices, and contexts of each session. LSTM networks are designed to remember information for long periods which allows the system to change recommendations as user behavior and preferences shift, enabling situational personalization. For example, it can offer short tutorials when users are likely to be commuting and more comprehensive modules are reserved for heavier study sessions which increases relevance and engagement.

4) Hybrid aggregation mechanism: This last building block combines the results from CF, CBF, and the LSTM-based contextual model. Based on the relevance, reliability, and contextual appropriateness of each module, the recommendations are weighted and aggregated into a comprehensive, context-aware recommendation list. This fusion mechanism maintains equilibrium in the hybrid system while interlocking each model's pivots. Whereas CF focuses on the community trends, CBF captures the content relevance, and the LSTM-based CDM integrates the situational and temporal aspects.

Collectively, these four components provide a powerful, flexible, and expandable framework capable of offering highly personalized e-learning recommendations. In addition to enhancing prediction accuracy, the architecture improves learner satisfaction and engagement by resolving challenges associated traditional models, such as the cold-start problem, static content dependence, and the absence of contextualization.

B. Framework Development Process

Constructing the hybrid framework is based on an organized multi-step pipeline where all parts are built on the edX Course Data dataset.

1) Data preprocessing: The information in the dataset falls into three categories: i) the history of user interactions with items, including clicks, ratings, enrollments, and course completions; ii) course metadata which encompasses description, categories and keywords, as well as information about the instructor; iii) contextual data which includes timestamps, device types, and session information. The cleaning process for this phase involves removing outliers and handling missing values, which requires the use of either imputation or removal techniques. Categorical variables such as user demographics and course categories are encoded through one-hot or label encoding, while numerical features such as ratings and time spent are normalized to a consistent scale.

Moreover, feature engineering refines user interactions to create the user-item interaction matrix for collaborative filtering (CF). For content-based filtering (CBF), course metadata is used to extract TF-IDF features. Additionally, time-series structured contextual data is prepared for training the RNN-based models.

2) Independent model implementations:

- Collaborative Filtering using SVD: The interaction matrix is split into latent factors for users and courses. Ratings for non-evaluated items are predicted using the dot product of these vectors, enabling user-specific recommendations to be tailored.
- Content-Based Filtering using TF-IDF: For textual features of the courses, TF-IDF scores are calculated. Using cosine similarity, a user's preference profile is measured against course content to rank and recommend the most relevant ones, thus enabling course ranking and recommendation.
- Contextual Model using RNN: Sequences of user behaviors along with contextual variables like device used and time are formatted as ordered sequences and used to train the RNN. The RNN LSTM captures temporal regularities and makes predictions for user preference scoring to recommend relevant courses.

3) Score normalization: Each model's CF, CBF, and RNN outputs are normalized to the [0, 1] range using Min-Max Scaling. This normalization guarantees model output comparability and is crucial for effective aggregation.

Before fusion, the outcomes generated by each model, specifically the Collaborative Filtering, Content-Based Filtering, and the Recurrent Neural Network, are scaled down to the $[0, 1]$ interval using the Min-Max scaling method. This step is crucial since the models feature scores that differ in terms of numerical consistency, distribution, and in the case of CBF, may even utilize a distinct calculating method. Model outputs entail dissimilar Merits, valued predictions, range estimates, or probabilities. Without the Min-Max scaling, the more significant the range of a score, the more influence it would have on the fusion process leading to skewed and unreliable recommendations.

In relation to a specific predictive score s_i, that a model outputs for a single learner x item combination, the Min-Max normalization is described as follows:

$$s_i^{\text{norm}} = \frac{s_i - s_{\min}}{s_{\max} - s_{\min}}$$

where, $s_{\min}$ and $s_{\max}$ denote the minimum and maximum scores generated by the model for a given learner–item pair,

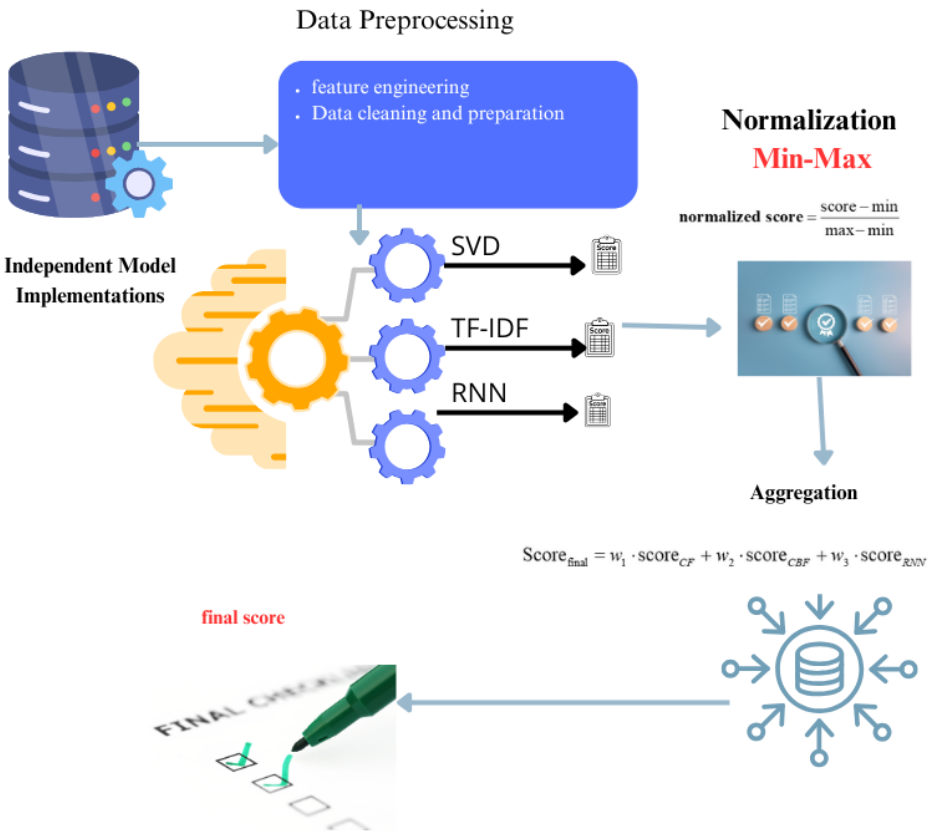


Fig. 1. Framework overview.

respectively, thereby rescaling all outputs to a common range between 0 and 1.

With the normalization process, the same score level is bestowed upon each model, proving a fair ground for comparison, and in-turn a more precise aggregation technique is applied to the weighted fusion layer. This directly increases the quality of each component in the score fusion process and adds more to the system as a whole in terms of reliability, consistency, and more comprehensibility to the hybrid recommender system.

4) Result aggregation: The normalized outputs of the three recommendation models namely the SVD-based collaborative filtering module, the TF–IDF content-based module, and the RNN-based contextual module—are combined using a weighted hybrid aggregation strategy. Each model is assigned a weight according to its predictive performance on the validation set, such that models achieving higher accuracy (i.e., lower prediction error) contribute more to the final recommendation; the corresponding computed weight values are reported in Table I.

Let E_i denote the prediction error of model i, measured using RMSE or MAE. The adaptive weight w_i associated with each model is computed as the normalized inverse of its error:

$$w_i = \frac{\frac{1}{E_i}}{\sum_{j=1}^{3} \frac{1}{E_j}} \tag{1}$$

This formulation ensures that models with lower prediction errors receive higher weights, thereby exerting greater influence on the final recommendation score. For instance, if the RNN model achieves the lowest RMSE among the three, it is assigned the largest weight, followed by the TF–IDF and SVD models.

Let $S_i(u,k)$ represent the normalized score produced by model i for user u and course k. The final hybrid recommendation score is then computed as:

$$\begin{aligned} S(u,k) = {} & w_{\text{SVD}}\, S_{\text{SVD}}(u,k) \\ & + w_{\text{TF-IDF}}\, S_{\text{TF-IDF}}(u,k) \\ & + w_{\text{RNN}}\, S_{\text{RNN}}(u,k) \end{aligned} \tag{2}$$

This adaptive fusion mechanism enables the recommender system to dynamically balance collaborative, content-based, and contextual signals according to their reliability under the current data conditions. As a result, the hybrid model achieves greater robustness and predictive accuracy than any individual component, while remaining responsive to variations in learner behavior and contextual factors.

5) Evaluation: Evaluate the performance of the individual models and the hybrid system using the following metrics: Precision@K: Measures the proportion of relevant items in the top-K recommendations, and the Recall@K: Assesses the fraction of relevant items retrieved out of all applicable items,

TABLE I. ADAPTIVE WEIGHTING OF SVD, TF–IDF, AND RNN MODULES BASED ON RMSE.

Module	RMSE	Weight w_i
SVD	0.902	0.30
TF–IDF	0.856	0.32
RNN	0.812	0.38

Normalized Discounted Cumulative Gain (NDCG): Takes into account the rank position of recommendations, giving higher importance to higher-ranked items,Root Mean Squared Error (RMSE): Evaluates the accuracy of the predicted ratings, comparing the difference between predicted and actual values. Algorithm 1 and Algorithm 2 presents the hybrid recommendation systems.

IV. RESULTS AND DISCUSSION

This section describes how we evaluated hybrid e-learning recommender systems using the rich edX Course Data. As data collected from millions of learners around the globe, it incorporates different views to analyze learner disposition. As we developed and tested advanced recommendation systems, the data also allows us to understand learner behavior, course content and the context within which the recommendations were made.

The dataset consists of three major parts. The first part containing data of user interactions records course enrollments and completions, audit, ratings, and detailed activity logs. Every interaction captures unique and important behavioral components of learners thus helping to model long-term interest and short-term engagement. Such user behavior data allows the recommender system to not only find and cluster learners with similar and competing interests but also track the dynamic order and sequence of courses to predict the next course a learner might be interested in.

The course metadata contains both structured and unstructured texts, for example, course titles, descriptions, types, keywords, and other content descriptors. This information is essential for the content-based filtering component, as it allows cross matriculation of learner profiles and courses with pertinent characteristics. The richly developed text features provide enough information for the model to capture the semantics of various courses, allowing it to recommend those that best align with what the learner has engaged with or indicates interest in.

A. Dataset

The datasets used in this research study are collected from edX, a popular resource for academic research in e-learning and educational recommendation systems. The dataset includes a total of five thousand learners, one thousand learning resources, and five thousand unique user-to-resource interactions, indicating a collection of diverse learning behaviors and content engagement patterns.

User-resource interactions can be classified to indicate whether a learner has performed an action with a particular educational resource. Actions may include registering for a course, browsing content, watching videos, or submitting assignments. These action behaviors are used as signals within the collaborative filtering component to learn user preferences, while the TF–IDF descriptive and textual resource data learn user content preferences. Additionally, the data collected over time helps the RNN capture and model the time dimensions of learning behaviors to address the shift in learners' interests and learning paths.

A standard hold-out protocol was used to split the dataset into two non-overlapping subsets for model training and evaluation: 80% of the dataset was allocated for training, while the remaining 20% was set aside for testing. Such a split guarantees that the models being learned will only be evaluated on previously unseen interactions, which gives a trustworthy assessment of their generalization capability. Thanks to the user diversity, content richness, and the presence of temporal interaction data, the edX dataset stands out to evaluate context-aware and hybrid recommender systems in real-world e-learning environments.

B. Hyperparameters

The performance of the proposed hybrid recommender system strongly depends on the appropriate tuning of the hyperparameters of its three constituent modules, namely the SVD-based collaborative filtering model, the TF–IDF content-based model, and the LSTM-based contextual model. To ensure a fair and reproducible evaluation, the hyperparameters were selected through validation experiments and empirical tuning, balancing model complexity and generalization ability. The final configuration adopted for all experiments is reported in Table II.

For the SVD component, the number of latent factors controls the dimensionality of the user–item interaction space and directly influences the model's ability to capture hidden preference patterns. The learning rate and regularization term were carefully adjusted to ensure stable convergence and prevent overfitting, as summarized in Table II. In the TF–IDF module, parameters such as the n-gram range, vocabulary size, and document frequency thresholds were optimized to generate informative and discriminative content representations while reducing noise and sparsity.

For the LSTM-based contextual model, key hyperparameters, including the embedding dimension, number of hidden units, sequence length, dropout rate, and optimization settings, were tuned to effectively model temporal dependencies in learners' interaction histories without excessive complexity. These settings, detailed in Table II, enable the LSTM to capture evolving learning behaviors while maintaining stable and efficient training. Overall, the hyperparameter configuration reported in Table II ensures that the three modules operate in a balanced and complementary manner, allowing the hybrid framework to fully exploit collaborative, content-based, and contextual information in the recommendation process.

C. Evaluation Results and Discussion

Alongside evaluating the proposed hybrid recommender system's overall performance, we also examined its performance with and without contextual data. The results indicate that including contextual information considerably improves the system's effectiveness, as demonstrated in Fig. 2. While

Algorithm 1 Hybrid Recommendation System

```
Input: dataset, weights
Output: final_scores, precision, recall, ndcg, rmse
Step 1: Data Preprocessing
function PREPROCESSDATA(dataset)
    CLEAN dataset by handling missing values and outliers
    NORMALIZE numerical features to a uniform scale
    ENCODE categorical variables if required
    user_item_matrix = CREATE_USER_ITEM_MATRIX(dataset)          ▷ For CF
    textual_features = EXTRACT_TEXTUAL_FEATURES(dataset)         ▷ For CBF
    contextual_data = ORGANIZE_CONTEXTUAL_DATA(dataset)          ▷ For RNN
    return user_item_matrix, textual_features, contextual_data
end function
Step 2: Independent Model Implementations
function COLLABORATIVEFILTERING(user_item_matrix)
    [U, S, Vt] = SVD(user_item_matrix)
    predicted_ratings = PREDICT_RATINGS(U, S, Vt)
    rmse = CALCULATE_RMSE(predicted_ratings)
    return predicted_ratings, rmse
end function
function CONTENTBASEDFILTERING(textual_features)
    tfidf_matrix = COMPUTE_TFIDF(textual_features)
    similarities = COSINE_SIMILARITY(tfidf_matrix)
    ranked_courses = RANK_COURSES_BY_SIMILARITY(similarities)
    return ranked_courses
end function
function CONTEXTUALMODEL(contextual_data)
    sequences = PREPARE_SEQUENCES_FOR_RNN(contextual_data)
    rnn_model = TRAIN_RNN(sequences)
    context_predictions = PREDICT_WITH_RNN(rnn_model)
    return context_predictions
end function
```

Algorithm 2 Hybrid Recommendation System 2

```
Step 4: Result Aggregation (Weighted Hybrid)
function AGGREGATERESULTS(cf_scores, cbf_scores, contextual_scores, weights)
    final_scores = weights[0] * cf_scores + weights[1] * cbf_scores + weights[2] *
contextual_scores
    return final_scores
end function
Step 5: Evaluation
function EVALUATESYSTEM(final_scores, ground_truth)
    precision = CALCULATE_PRECISION_AT_K(final_scores, ground_truth)
    recall = CALCULATE_RECALL_AT_K(final_scores, ground_truth)
    ndcg = CALCULATE_NDCG(final_scores, ground_truth)
    rmse = CALCULATE_RMSE(final_scores, ground_truth)
    return precision, recall, ndcg, rmse
end function
```

session time, device type, user location, and interaction sequences are used as contextual data, this information helps the model to adapt its recommendations to the learner's situation, increasing engagement and the relevance of recommended courses.

The hybrid recommender system continues to achieve higher results across all the evaluation measures—Precision, Recall, F1-Score, and Mean Absolute Error (MAE)—than the baseline models [e.g., standalone Collaborative Filtering (CF) and Content-Based Filtering (CBF)], as shown in Fig. 3. The hybrid system results include, 0.75 precision, 0.68 recall, and 0.71 F1-score, while the CF results were 0.68 precision, 0.60 recall, and 0.63 F1-score, and CBF results were 0.70 precision, 0.64 recall, and 0.67 F1-score. The hybrid system delivered a 0.18 MAE, underscoring the improved predictive accuracy with the combination of user behavior and contextual data from the edX dataset. This compares to 0.21 with CF and 0.20 with CBF.

The hybrid framework, SVD-based CF, and deep learning-based (LSTM) models were additionally analyzed comparatively. Evaluation through all four primary metrics validates that, in most instances, the hybrid method performed best,

TABLE II. HYPERPARAMETERS OF THE SVD, TF–IDF, AND LSTM MODULES

Module	Hyperparameter	Description	Value / Setting
SVD (CF)	Latent factors (k)	Dimensionality of the user/item latent space	50
	Learning rate (η)	Step size for gradient updates	0.005
	Regularization (λ)	L2 penalty to reduce overfitting	0.02
	Epochs	Number of training iterations	50
	Initialization	Latent vector initialization strategy	Random (Gaussian)
	Loss function	Optimization objective for rating/implicit signal prediction	MSE
TF–IDF (CBF)	n-grams	Token range for feature extraction	(1,2)
	Vocabulary size	Maximum number of features retained	20,000
	Minimum document frequency (min_df)	Ignore rare terms below threshold	2
	Maximum document frequency (max_df)	Ignore overly frequent terms above threshold	0.90
	Normalization	Vector normalization scheme	L2
	Similarity metric	Similarity computation between course profiles	Cosine
LSTM (Context)	Embedding dimension	Size of interaction/context embedding	128
	Hidden units	Number of LSTM hidden neurons	128
	Number of LSTM layers	Depth of the recurrent network	1
	Dropout	Dropout rate for regularization	0.30
	Sequence length (T)	Number of past interactions used as input	20
	Batch size	Number of samples per training batch	64
	Optimizer	Optimization algorithm	Adam
	Learning rate	Step size for optimizer	0.001
	Epochs	Number of training iterations	30

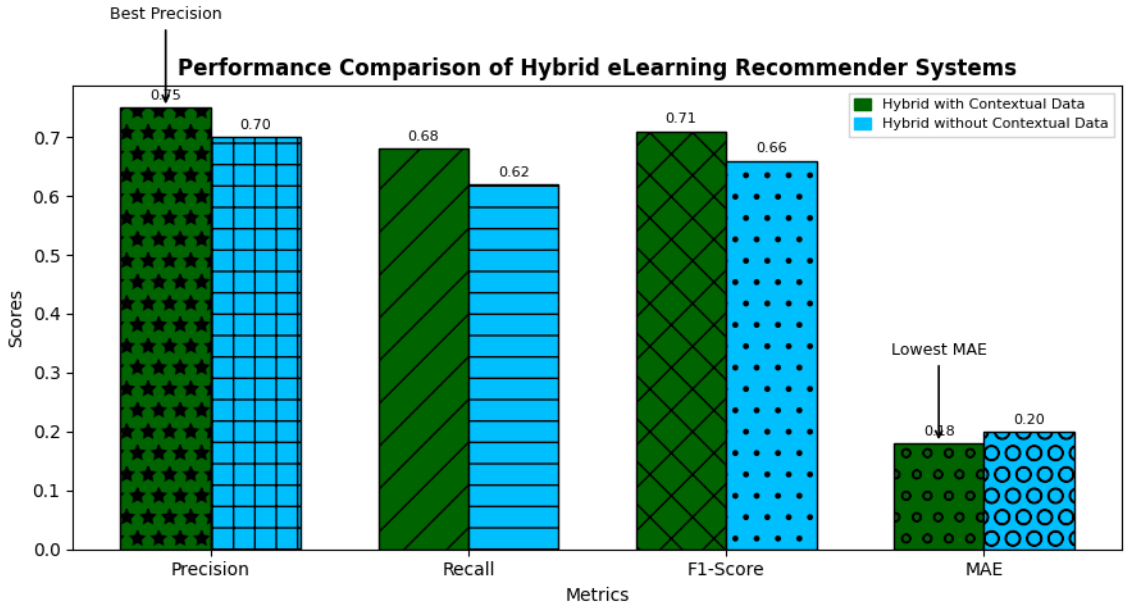


Fig. 2. Performance comparison of hybrid e-learning recommender systems.

TABLE III. COMPARATIVE PERFORMANCE OF RECOMMENDATION METHODS

Method	Precision	Recall	F1-Score	MAE
DL-based (NN)	0.73	0.66	0.69	**0.17**
SVD (Classic MF)	0.68	0.60	0.63	0.22
SVD + Tf-IDF + RNN (Our method)	**0.75**	**0.68**	**0.71**	0.18

see Fig. 4. For precision, the hybrid method achieved 0.75, ahead of the DL-based method's 0.73 and SVD's 0.68. The hybrid method also led in recall, attaining 0.68, greater than DL-based's 0.66 and SVD's 0.60, as shown in Table III. The hybrid system also achieved the best F1-score of 0.71, while the DL-based models scored 0.69 and SVD scored 0.63. The

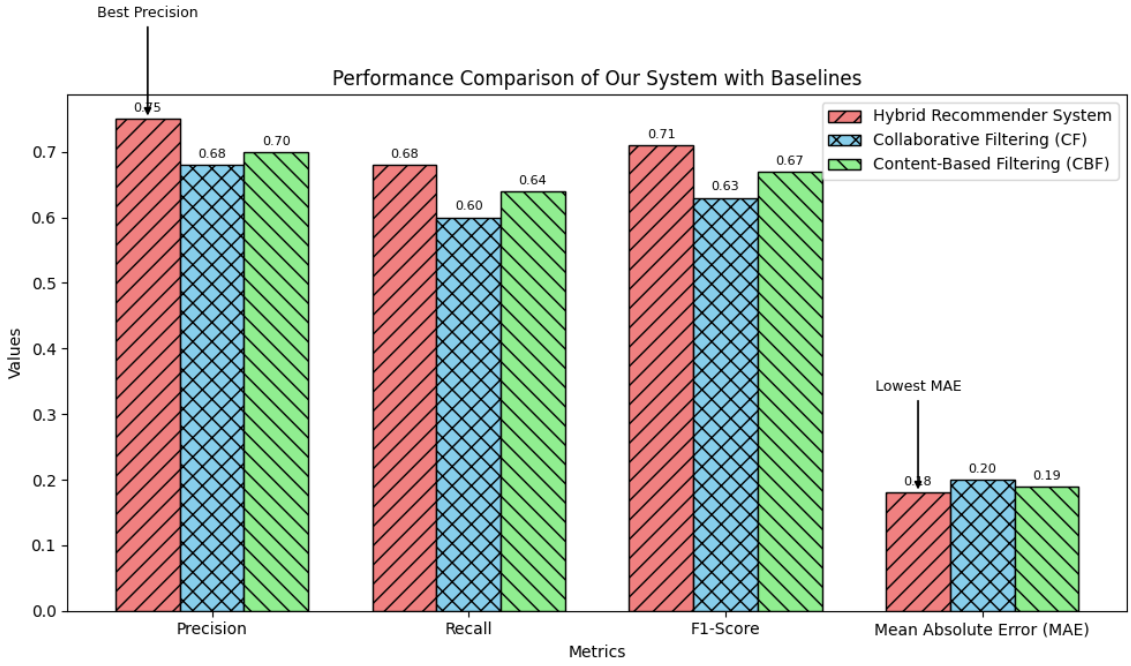


Fig. 3. Performance comparison of our system with baselines.

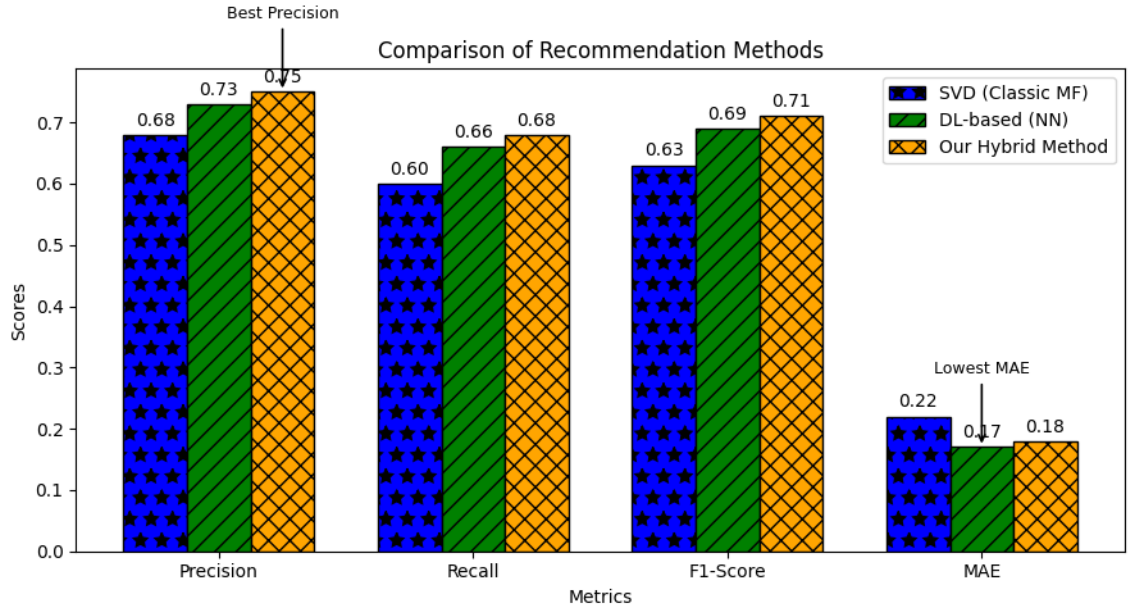


Fig. 4. Comparison of recommendation methods.

DL-based method had a slight edge in MAE, scoring 0.17, while the hybrid scored 0.18, and SVD scored 0.22. The hybrid system makes a balanced improvement by confirming active adjustments to the accuracy, as well as the error reduction, as it provides concrete evidence to the system's reliability and robustness.

As highlighted in this analysis, there are benefits to integrating CF, CBF, and contextual information. CF is good at predicting preferences based on user behavior, but it does not work well in the case of new users or courses with sparsely populated interaction history. CBF somewhat mitigates the cold-start problem, but is not able to adapt to shifting user behavior and situational dynamics. The hybrid approach improves on this by combining the methods with contextual data and producing suggestions based on learner preferences and situational constraints present in the real-world. For example, if users are engaged in short study sessions, the system

suggests quick, shorter courses, while in the case of longer study sessions, it suggests comprehensive longer modules. This recommend system improves user satisfaction and engagement, which is especially important in the dynamic context of e-learning.

There are, however, unsolved issues. In some cases, sparse contextual data for user profiles, new users especially, or under-enrolled courses, may constitute a limit to system performance, even with contextual data augmentation. Furthermore, the need to combine multiple data sources poses a significant challenge in terms of real-time recommendations. This is likely due to the system requiring excess processing power on large platforms like edX, making it a scalability barrier. Future work could target this problem by optimizing, for example, distributed computing or incremental learning coupled with lightweight neural nets to improve processing speed while preserving the quality of the recommendations.

Enriching the contextual data surrounding users in an e-learning environment increases the value of the recommendations provided. While recommender systems like Collaborative Filtering (CF) and Content-Based Filtering (CBF) offer the basics of personalized recommendations based on historical interactions or static content features, they do not capture the dynamic situations the learner is in. Thousands of interactions provide contextual data on patterns, such as duration of content sessions and timing of access, while device type and geo-location add further dimensions. By these measures, the hybrid recommender system is able to escape the proposition of static recommendations and deliver relevant and actionable suggestions based on the learner's context.

Precision and relevance of the recommendations are enhanced with the addition of contextual data. For instance, learners accessing the platform when taking short breaks would appreciate brief modules or quick tutorials. In contrast, learners engaging in longer study periods are eligible for more extensive courses or multi-part learning paths. This temporal and situational awareness can make recommendations relevant for learners, focusing on their attention spans and immediate needs. This increases the probability of completion and active utilization of recommended courses. Real-time adaptation to changes in conditions creates unprecedented flexibility, accommodating user behavior and preferences, and external limitations, and their behavioral and contextual changes without requiring overriding manual control.

Learning contextual data enhances learner motivation and engagement. Improved engagement and motivation primarily stem from the interaction of contextual information with the designed learning activities. Users are more likely to continue engaging with the platform when contextual data is incorporated into the recommendations. The hybrid system's ability to incorporate contextual factors creates a purposeful and meaningful learning environment that enhances immersion and satisfaction.

Incorporating context-aware hybrid recommender systems sets a new standard in personalized e-learning. Differing from conventional models that rely on fixed features or merely historical features, these systems dynamically adjust to learners as they progress, updating suggestions in real-time with contextual and behavioral information. Such flexibility is important in real e-learning systems to ensure compatibility with varying user demographics and learning situations. Use hybrid systems maximize contextual information; they effectively improve outcomes, adjust learning personalization and support high flexibility educational systems, paving the way for advanced smart educational frameworks.

V. Conclusion

This work was the first to combine Collaborative Filtering, Content-Based Filtering, and Contextual Data Modeling into a hybrid e-learning recommender system to create context-aware, personalized e-learning course recommendations. Using the additional and overlapping strengths of the aforementioned frameworks decreased some of the major drawbacks of conventional recommender systems, including underexploited cold-start and data sparsity situations, excessive overspecialization of recommendations, and a failure to address shifting user needs and preferences. Some of the advanced algorithms implemented in this hybrid system include Singular Value Decomposition (SVD) for latent factor analysis, TF-IDF for content feature extraction, and Long Short-Term Memory (LSTM) networks for temporal and contextual user behavior modeling.

The hybrid system, which was evaluated on the edX course dataset, consistently surpassed solo CF and CBF baseline models on metrics such as precision, recall, F1 score, and Mean Absolute Error (MAE), as well as on multiple other metrics. Contextual data such as the user's device, session duration, and location accessed data, which was incorporated into the system, significantly improved the precision and utility of the recommendations. Lastly, the fusion mechanism, which integrates the disparate evaluation systems within each of the three components, offers the system a high degree of flexibility and robustness in addition to balance, accuracy, diversity, and adaptability in its recommendations.

Research contributions are as follows: First, this work presents a unified hybrid framework combining collaborative, content-based, and context-aware strategies for e-learning. Second, this work illustrates the synergistic use of SVD, TF-IDF, and LSTM in tackling common problems in recommendation systems. Third, this work uses educational real-world data as evidence that the proposed framework offers a more personalized approach to recommend courses and improves the overall learning experience.

Although there are achievements, limitations and challenges still exist. The system in use is guided by historical and contextual data which may not be complete and accurate, and real-time adaption to user behavior shifts and cross-platform access still can be improved. Furthermore, the protective use of real-time contextual and behavioral data and the systems' data acquired for situational context reveals core challenges of privacy and security.

Subsequent work will target the use of reinforcement learning and deep learning because they can enhance the systems understanding of nuanced user preferences and advanced sequence of learning patterns. The system will be more personalized by incorporating learner goals, motivation, and affect. Finally, system extensibility to permit multi-platform systems, in conjunction with the use of privacy-conscious

strategies for system deployment in e-learning environments is imperative for system's ethical and responsible scalability.

The proposed hybrid and context-aware recommender system represents a strong, adaptable, and extensible approach to personalization in e-learning. Incorporating elements of collaborative, content-based, and contextual mechanisms yields not only enhanced precision and pertinence in recommendations but also sets a foundation for more sophisticated, evolving, and learner-focused approaches to education in the digital age.

ACKNOWLEDGMENT

The authors would like to thank the editors and anonymous reviewers for their valuable comments and constructive suggestions which greatly helped improve the quality and clarity of this manuscript. We also gratefully acknowledge the support of our institution and colleagues who contributed to the completion of this work.

REFERENCES

[1] C. D. Kintu, C. Zhu, and E. Kagambe, "Blended learning effectiveness: The relationship between student characteristics, design features and outcomes," *Int. J. Educ. Technol. Higher Educ.*, vol. 14, no. 7, pp. 1–20, 2017.

[2] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, "Recommender system application developments: A survey," *Decision Support Systems*, vol. 74, pp. 12–32, 2018.

[3] A. Aggarwal et al., "Collaborative filtering and hybrid models in recommender systems: A review and future directions," *Inf. Process. Manage.*, vol. 58, no. 6, p. 102657, 2021.

[4] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning-based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, 2019.

[5] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, no. 4, pp. 1–19, 2009.

[6] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," *Recommender Systems Handbook*, Springer, 2011, pp. 73–105.

[7] M. Akbari and C. C. Chi, "Context-aware recommender systems for learning resources: A survey," *IEEE Trans. Learn. Technol.*, vol. 14, no. 1, pp. 28–49, Jan.–Mar. 2021.

[8] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.

[9] T. Bogers and M. Koolen, "Defining and evaluating context for personalized recommendations in digital libraries," *J. Assoc. Inf. Sci. Technol.*, vol. 69, no. 6, pp. 742–753, 2018.

[10] H. Wei, J. Liu, and X. Chen, "Context-aware e-learning recommender system using hybrid approach," *IEEE Access*, vol. 7, pp. 73450–73461, 2019.

[11] Y. Liu et al., "A neural collaborative filtering model enhanced with attention mechanism for personalized recommendations," *Expert Syst. Appl.*, vol. 164, p. 113992, 2021.

[12] A. Jameson et al., "Contextual effects on recommender systems in e-learning platforms," *ACM SIGKDD Explor. Newslett.*, vol. 22, no. 1, pp. 19–25, 2020.

[13] K. D. Brinton, M. Chiang, and H. V. Poor, "Learning networks for smart education," *Proc. IEEE*, vol. 107, no. 9, pp. 1703–1713, 2019.

[14] P. Mateos and A. Bellogín, "A systematic literature review of recent advances on context-aware recommender systems," *Artif. Intell. Rev.*, vol. 58, art. 20, 2025.

[15] M. E. Chafiki, O. Stitini, and S. Kaloun, "A systematic review of advancements in context-aware recommendation systems," *Knowl. Inf. Syst.*, vol. 1, 2025.

[16] S.-Y. Jeong and Y.-K. Kim, "Deep learning-based context-aware recommender system considering change in preference," *Electronics*, vol. 12, no. 10, art. 2337, 2023.

[17] S. Raza *et al.*, "A comprehensive review of recommender systems: Transitioning from theory to practice," preprint, arXiv:2407.13699, 2024.

[18] Y. Zhang, L. Chen, and H. Wang, "Neural collaborative filtering with attention mechanisms for personalized recommendations," *Expert Syst. Appl.*, vol. 178, p. 115049, 2022.

[19] "CNNRec: Convolutional neural network-based recommender systems – a survey," *Eng. Appl. Artif. Intell.*, vol. 133, 2024.

[20] M. Bukhari, M. Maqsood, and F. Adil, "An actor-critic based recommender system with context-aware user modeling," *Artif. Intell. Rev.*, vol. 58, art. 138, 2025.

[21] M. Almasi et al., "Context-sensitive recommender systems: A survey of current methods and trends," *ACM Comput. Surv.*, vol. 54, no. 2, pp. 1–40, 2022.

[22] F. Ricci et al., *Recommender Systems Handbook*, 3rd ed., Springer, 2022.

[23] C. C. Aggarwal et al., *Recommender Systems: The Textbook*, 2nd ed., Springer, 2022.

[24] C. Zhong, Y. Liu, J.-S. Wang, and Z.-F. Li, "RBFNN Fault Diagnosis Method of Rolling Bearing Based on Improved Ensemble Empirical Mode Decomposition and Singular Value Decomposition," *IAENG International Journal of Computer Science*, vol. 49, no. 3, 2022.

[25] H. Pu and Y. Hua, "Adaptive course recommendation using federated learning and graph convolutional networks in IoT-enhanced e-learning," *Sci. Rep.*, vol. 15, Art. 42040, 2025.

[26] A. N. Hasoon, S. K. Abdulateef, R. S. Abdulameer, and M. L. Shuwandy, "An intelligent hybrid AI course recommendation framework integrating BERT embeddings and Random Forest classification," *Computers*, vol. 14, no. 9, Art. 353, 2025.

[27] D. A. Hakim, N. Wilantika, B. A. B. Fikri, R. Yosral, and N. B. Kurniawan, "A modified hybrid content-aware course recommendation model for Moodle-based Learning Management System," *Inform: J. Teknol. Inf. dan Komunik.*, vol. 11, no. 1, pp. 65–70, 2026.

[28] S. Kim and J. Lee, "Recent advances in context-aware recommender systems: A systematic review," *ACM Comput. Surv.*, vol. 56, no. 3, pp. 1–38, 2024.

[29] J. Zou, C. Lin, W. Guo, Z. Wang, J. Wei, Y. Yang, and H. Shen, "Multi-type context-aware conversational recommender systems via mixture-of-experts," *arXiv preprint*, arXiv:2504.13655, 2025.

[30] N. L. Le and M. H. Abel, "From individual to group: Developing a context-aware multi-criteria group recommender system," *arXiv preprint*, arXiv:2503.22752, 2025.

[31] S. Huang, H. Yang, Y. Yao, X. Lin, and Y. Tu, "Deep Adaptive Interest Network: Personalized recommendation with context-aware learning," *arXiv preprint*, arXiv:2409.02425, 2024.

Efficient CNN-Based Time-Domain Denoising of Impulsive Noise in NB-PLC Systems

Wided Belhaj Sghaier, Fatma Rouissi, Héla Gassara, Fethi Tlili
GRESCOM Lab, Ecole Supérieure Des Communications De Tunis, University of Carthage, Tunisia

***Abstract*—In this study, a convolutional neural network (CNN)-based time-domain denoising approach is proposed to suppress impulsive noise which is considered as the most sever impairments in narrowband powerline communications (NB-PLC). Unlike conventional techniques, such as clipping and blanking, the proposed method does not require prior knowledge of noise statistics. The introduced CNN network is trained using synthetically generated OFDM signals corrupted by Middleton Class-A impulsive noise, calibrated from real NB-PLC measurement data. Extensive G3-PLC-compliant simulations demonstrate that the proposed method significantly outperforms classical blanking and clipping schemes. At an SNR of 10 dB, the proposed CNN achieves a mean squared error (MSE) of 1.2×10^{-4}, compared to 2.3×10^{-4} and 2.5×10^{-4} for blanking and clipping, respectively, under time-varying impulsive noise conditions. Moreover, the receiver incorporating the denoising method closely approaches the ideal AWGN reference under low impulsive noise density and for SNR values above 12 dB.**

***Keywords*—NB-PLC; Middleton Class-A; OFDM; impulsive noise; deep learning; CNN**

I. Introduction

In recent years, significant research efforts have been devoted to smart grid applications based on narrowband power line communication (NB-PLC) [1], [2]. This technology enables data transmission over existing power distribution lines within a frequency band below 500 kHz, offering wide coverage at low deployment cost. NB-PLC systems facilitate communication expansion without the need for new infrastructure, as they leverage the already deployed electrical grid. However, the NB-PLC channel which not originally designed for data transmission is subject to several technical impairments. It is characterized by severe multipath propagation resulting from impedance mismatches and network topology variations in the power grid [3]. Moreover, NB-PLC channels are affected by various types of noise, among which impulsive noise is considered one of the most detrimental disturbances. Generated by electrical devices connected to the grid—such as motors, switching power supplies, lamps, and household appliances—impulsive noise can reach power levels up to 50 dB above the background noise [6]. It is widely recognized as a primary performance-limiting factor in NB-PLC systems. It is characterized by high amplitude, short duration, and random occurrence, and deviates significantly from the Gaussian noise assumption commonly used in communication theory. Unlike additive white Gaussian noise (AWGN), impulsive noise generates burst errors that conventional coding schemes struggle to correct, resulting in severe degradation of communication performance [7], [8]. This explains why OFDM modulation, used in NB-PLC systems and forming the basis of the PRIME [10], [11], [12] and G3-PLC [9] standards, is highly vulnerable to impulsive noise [13].

Over the years, various impulsive noise mitigation techniques have been proposed. Early research efforts mainly focused on traditional signal-processing-based solutions. Error-control-based techniques, including channel coding and interleaving, were employed to enhance robustness against channel impairments [14], [15], [16]. In parallel, nonlinear time-domain preprocessing methods such as blanking and clipping were introduced to suppress high-amplitude impulsive disturbances prior to OFDM demodulation [18]. In addition, statistical approaches relying on impulsive noise modeling and parameter estimation were widely investigated [19]. Although these methods are well established and relatively simple to implement, their performance is highly dependent on accurate noise statistics and fixed design parameters, which limits their robustness in practical NB-PLC environments characterized by non-stationary and appliance-dependent impulsive noise.

More recently, the emergence of deep learning (DL) has introduced a data-driven paradigm for impulsive noise mitigation in PLC systems. This capability has motivated extensive research efforts in PLC systems, where deep neural networks have been primarily employed to overcome the limited adaptability of conventional techniques. Early DL-based studies focused on estimating optimal parameters for nonlinear preprocessing methods, such as adaptive threshold selection for clipping and blanking, leading to significant SNR and BER improvements under time-varying impulsive noise conditions [25]. More advanced architectures, including CNN- and LSTM-based models, were subsequently proposed to exploit temporal and multi-domain features of impulsive noise [28], while recent self-supervised approaches have aimed at reducing the reliance on clean reference signals [26].

Despite recent advances in DL-based impulsive noise mitigation for PLC systems, direct end-to-end denoising solutions tailored to NB-PLC receivers remain largely unexplored. Existing studies mainly rely on adaptive parameter estimation for nonlinear preprocessing techniques, complex multi-branch architectures, or self-supervised models that require significant training and implementation effort. To the best of our knowledge, none of these approaches directly estimate the clean signal at the receiver.

In this study, we propose a time-domain CNN to directly suppress impulsive noise and produce a cleaned received signal. This approach is motivated by the proven effectiveness of CNNs in image and signal processing applications [29], [30]. Our CNN architecture is specifically designed for NB-PLC, aiming to enhance data transmission reliability under realistic NB-PLC noise conditions.

The remainder of this study is organized as follows: Section II presents the review of impulse noise cancellation methods. Section III describes the communication system model, details the statistical characteristics of impulsive noise in NB-PLC. Section IV introduces the proposed CNN-based impulsive noise mitigation framework, including the design of the CNN architecture and the dataset generation methodology. Section V reports and discusses the simulation results. Finally, Section VI summarizes the main contributions of this work and outlines potential directions for future research.

II. State-of-the-Art Approaches for Mitigating Impulsive Noise

A. Traditional Impulsive Noise Mitigation Techniques in PLC Systems

Over the years, various impulsive noise mitigation techniques have been proposed, which can be broadly classified into four categories: error-control-based methods, nonlinear preprocessing techniques, statistical model-based approaches, and learning-based methods [31].

Early studies focused primarily on error-control mechanisms at the physical layer. Channel coding techniques such as low-density parity-check (LDPC) codes [14], Turbo codes [15], polar codes [16], and bit-interleaved coded modulation (BICM) [17] were employed to enhance robustness against channel impairments. these techniques improve resilience to Gaussian noise, but are less effective against impulsive noise, as strong impulses cause long error bursts that exceed correction capabilities and reduce efficiency.

To directly suppress impulsive disturbances in the time domain, nonlinear preprocessing techniques such as blanking, clipping, and deep clipping were introduced at the receiver [18]. These methods are attractive due to their simplicity and low computational complexity. However, their performance is highly sensitive to fixed threshold parameters. Since impulsive noise statistics in PLC environments are time-varying and appliance-dependent, fixed-threshold methods lack adaptability and may either fail to suppress noise effectively or distort the useful signal.

Another class of approaches relies on statistical modeling and parameter estimation of impulsive noise. Techniques based on noise parameter estimation [19], cyclostationary noise mitigation [20], and frequency-domain minimum mean square error (MMSE) equalization [21] have been widely investigated. Although these methods are theoretically well-founded, they depend heavily on accurate noise models and precise parameter estimation. In practical NB-PLC scenarios, model mismatch and rapidly changing noise conditions often degrade their performance, while the associated computational complexity limits real-time applicability.

The sparse nature of impulsive noise has been exploited using compressed sensing techniques to estimate and cancel impulsive disturbances in OFDM-based PLC systems [22]. Sparse Bayesian learning (SBL) further improved robustness by providing probabilistic noise estimation [23], and advanced schemes such as sparse iterative covariance estimation were also proposed [24]. Despite their strong performance, these approaches typically require additional null subcarriers, strict sparsity assumptions, and high computational resources, which restrict their practical deployment in low-complexity NB-PLC receivers.

B. Deep Learning Approaches for Impulsive Noise Denoising in PLC Systems

Recently, DL-based approaches have emerged as a promising alternative to overcome the limited adaptability of traditional threshold-based and model-driven techniques. Classical methods such as clipping, blanking, or statistical filtering rely on fixed parameters or strong assumptions about noise distributions, which are often violated in real PLC environments characterized by non-stationary and appliance-dependent impulsive noise. In contrast, DL-based methods learn noise characteristics directly from received data, enabling adaptive and robust suppression under dynamic conditions [25], [26].

Early DL-based works in PLC primarily focused on adaptive parameter estimation rather than direct end-to-end denoising. For example, in [25], convolutional neural network (CNN)-based approaches were proposed to estimate optimal clipping and blanking thresholds for time-domain impulsive noise mitigation in OFDM-PLC systems. These methods approximate optimal signal-to-interference-plus-noise ratio (SINR)-driven thresholds under clustered impulsive noise from household appliances. Simulation results demonstrated SNR gains of approximately 3–5 dB compared to fixed-threshold methods, leading to improved BER performance in time-varying noise environments.

To further enhance robustness and remove the need for handcrafted preprocessing, end-to-end DL architectures were introduced. In [27], the Multi-Feature Space Domain Fusion Network (MFSDF-Net) was proposed for impulsive noise suppression in MIMO-PLC systems. This CNN-based model extracts and fuses features from parallel convolution kernels operating in different spatial domains, enabling effective mitigation of asynchronous impulsive noise without requiring null subcarriers. Compared to traditional clipping/blanking and earlier DL models, MFSDF-Net significantly reduces BER floors and achieves an SNR of 18 dB at a BER of 10^{-5}, compared to 26 dB or higher for other models, corresponding to an 8 dB improvement in highly dynamic environments.

In another study [28], LSTM-based adaptive mitigation schemes were introduced to track impulsive noise states over time and dynamically adjust clipping decisions using sliding windows. These methods were combined with peak-to-average power ratio (PAPR) reduction techniques, such as accelerated proximal gradient methods and differential phase time encoding, to protect useful OFDM signals. Simulation results showed clear BER improvements over conventional nonlinear methods, particularly under rapidly changing impulsive noise conditions.

More recently, self-supervised models such as UNet-INSN have marked a shift toward label-free training. In [26], the authors proposed UNet-INSN, a 1D UNet-based architecture trained without clean reference signals. The model employs a global mask mapper and a reproducibility loss to prevent identity mapping, allowing training directly on noisy PLC signals. In simulations, UNet-INSN achieves a BER of 10^{-6} at approximately 12 dB SNR with ideal channel estimation and

26 dB with non-ideal estimation. It outperforms Noise2Void by about 3 dB, Zero-Shot Noise2Noise by 4 dB, and Deep-Clipping by nearly 7 dB, while maintaining less than 1 dB degradation when the impulsive noise probability varies from 0.01 to 0.05. These results demonstrate strong robustness in coded MIMO-PLC systems.

C. Blanking and Clipping-Based Time-Domain Noise Mitigation Approaches

Traditional time-domain impulsive noise suppression methods are memoryless nonlinear techniques that adjust the signal amplitude based on predefined thresholds while preserving its phase [31], [25]. Applied to the received signal prior to FFT demodulation in OFDM systems, the most used in the literature are blanking and clipping [31].

In the blanking technique, the received signal sample y_s is set to zero whenever its amplitude exceeds a predefined blanking threshold T_b, and remains unchanged otherwise [see Eq. (1)]:

$$y_s = \begin{cases} y_s, & |y_s| < T_b, \\ 0, & |y_s| \geq T_b. \end{cases} \tag{1}$$

The clipping method limits the signal amplitude to a clipping threshold T_c while preserving the phase, and is defined as Eq. (2):

$$y_s = \begin{cases} y_s, & |y_s| < T_c, \\ T_c \cdot e^{j \arg(y_s)}, & |y_s| \geq T_c. \end{cases} \tag{2}$$

Blanking and clipping scheme performances are highly dependent on the choice of the threshold. An inadequately selected threshold can either remove useful signal samples (over-blanking/clipping) or fail to suppress impulsive noise effectively (under-blanking/clipping), leading to significant BER degradation. Many classical approaches determine the threshold through exhaustive offline optimization for each SNR level, providing near-optimal performance but being computationally prohibitive for real-time implementation. Other works set the threshold as a multiple of the noise standard deviation; however, such fixed thresholds are highly sensitive to variations in noise statistics and are not robust in dynamic NB-PLC environments. More analytical approaches derive optimal thresholds based on closed-form SNR expressions, yet they generally assume prior knowledge of SINR/SNR, which is difficult to estimate accurately in practice [31], [32], [25].

Consequently, the practical applicability of thresold-based methods remains limited, motivating the development of data-driven deep learning approaches. These methods either aim to estimate optimal threshold or to detect the temporal positions of impulsive noise events prior to applying classical nonlinear mitigation techniques.

In this work, a CNN-based deep learning framework is proposed to directly estimate the clean signal without explicitly inferring the nonlinear processing parameters or requiring prior estimation of SINR or SNR at the receiver.

III. System Architecture

We investigate an NB-PLC OFDM communication system compliant with the G3-PLC specification, operating in the FCC frequency band ranging from 154.6875 kHz to 487.5 kHz, as defined by the ITU-T G.9903 standard [9]. Each physical frame contains $N_s = 12$ OFDM symbols. The system employs 128 subcarriers with an IFFT size of 256, resulting in a subcarrier spacing of 4.6875 kHz is performed over $N_d = 72$ active subcarriers using BPSK modulation. The cyclic prefix (CP) is added to each OFDM symbol to mitigate inter-symbol interference. The G3-PLC parameters for FCC band are detailed in Table I [9] and the block diagram of the G3-PLC OFDM is depicted in Fig. 1.

TABLE I. Key Parameters of G3-PLC in FCC Band Configuration [9]

Parameters	Value
Active subcarrier (N_d)	72
IFFT size	256
Subcarrier spacing	4.6875 kHz
OFDM symbols (N_s)	12
CP duration	30
Modulation type	BPSK
Operating frequency range	154.7–487.5 kHz

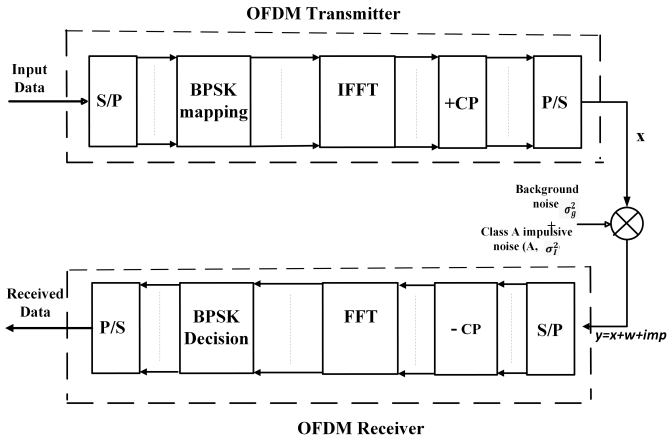


Fig. 1. Block diagram of G3-PLC OFDM system transceiver.

At the transmission side, coherent BPSK is employed for modulation.

The received signal $y(t)$ affected by impulsive noise, is expressed as Eq. (3):

$$y(t) = x(t) + w(t) + imp(t), \tag{3}$$

where, $x(t)$, $w(t)$ and $imp(t)$ denote transmistted signal, the additive white Gaussian noise (AWGN) and impulsive noise, respectively. After cyclic prefix removal and FFT blocks are applied, decision making for BPSK demodulation is performed.

A. Impulsive Noise Model

Numerous statistical models have been proposed in the literature to describe the impulsive noise [4], [5]. Among them the Middleton Class-A model is the most commonly used and has been extensively validated for NB-PLC environments.

The probability density function (PDF) of the Middleton Class-A noise is given by Eq. (4) [5]:

$$f_A(x_k) = \sum_{m=0}^{\infty} \frac{A^m e^{-A}}{m!} N(x_k, 0, \sigma_m^2), \tag{4}$$

where, x_k denotes the noise sample observed at discrete time index k, and A is the impulsive index, which controls the average density of impulsive events within a given observation interval. The term $N(x_k, 0, \sigma_m^2)$ represents a zero-mean Gaussian probability density function with variance σ_m^2, defined as Eq. (5):

$$\sigma_m^2 = \frac{\sigma_i^2}{A} m + \sigma_g^2 = \sigma_g^2 \left(\frac{m}{\tau} + 1 \right), \tag{5}$$

where, σ_i^2 denotes the variance of the impulsive noise component, σ_g^2 is the variance of the background gaussian noise, and $\tau = \sigma_g^2/\sigma_i^2$ represents the impulsive-to-Gaussian noise power ratio.

The magnitude of the impulse and the moment at which it occurs are key parameters of impulsive noise, as they fully characterize its behavior in the time domain, which are sufficient to describe its impact on the received signal. In NB-PLC channels, a single impulsive event typically corrupts one or a few signal samples, however, due to the FFT operation, its energy is spread over all subcarriers, leading to significant performance degradation if not properly mitigated.

IV. Proposed DL-Based Approach for Impulsive Noise Mitigation in NB-PLC Systems

CNNs are well suited for impulsive noise mitigation due to their ability to capture local patterns and abrupt signal variations caused by sparse, high-amplitude noise events.

Next, details of the CNN-based approach to suppress the impulsive noise are given, including the network architecture and the dataset generation process.

A. CNN-Based Impulsive Noise Mitigation In NB-PLC Receivers

The proposed CNN-based PLC signal denoising architecture is integrated into the OFDM receiver processing block, as illustrated in Fig. 2. It is placed before the serial-to-parallel conversion, CP removal, and FFT block within the OFDM receiver. The network input consists of time-domain samples arranged as a tensor of size $[1 \times 3432 \times 2]$, and the model is trained to estimate the corresponding clean signal.

The design of the CNN architecture, including the number of convolutional layers and kernel dimensions, is not based on heuristic assumptions, but is derived from a systematic experimental optimization performed during the offline training stage. The resulting architectural parameters are discussed in detail in Section V-A.

B. CNN Structure

The choice of CNNs model is motivated by prior work in wireless communication systems, which has consistently shown their superior effectiveness in denoising tasks. CNNs employ multiple learnable filters across successive layers to capture and refine complex structures in noisy observations enables [33], [34]. Moreover, their hierarchical architecture multiscale feature learning, a which is essential for effectively separating noise components from the underlying useful signal [33].

The proposed CNN architecture, depicted in Fig. 3, is formulated as a regression model. It is composed of five convolutional blocks, where the third and fifth blocks are replicated to reinforce feature learning capabilities, in addition to the input and output layers. The network input consists of a three-dimensional tensor of size $[1 \times 3432 \times 2]$, obtained after applying Z-score normalization to the received noisy OFDM samples.

Each convolutional block comprises a convolutional layer, followed by Batch Normalization and a Leaky ReLU activation function. The convolutional layers employ learnable kernels with *same* padding in order to maintain the spatial dimensions of the feature maps. Batch Normalization is used to reduce internal covariate shift and improve training stability, while the Leaky ReLU activation alleviates neuron saturation and facilitates faster convergence [33], [34]. The output stage consists of a final convolutional layer followed by a regression layer that produces the estimated clean signal with dimensions $[1 \times 3432 \times 2]$.

The convolutional layers are denoted using the format `conv | $K_s \times K_s$ | Num | same`, where `conv` represents a convolutional operation, K_s (9, 5, 3) indicates the kernel size, *Num* specifies the number of filters, and `same` denotes dimension-preserving padding. Employing convolutional kernels of varying sizes enables the network to effectively model impulsive noise patterns while retaining essential signal structures [37].

The proposed DL-network is trained using a dedicated dataset that reproduces the impulsive noise characteristics typical of NB-PLC channels. This training strategy allows the model to effectively cope with the temporal variability of the impulsive noise, thereby enabling accurate estimation of the clean OFDM signal under realistic NB-PLC operating conditions.

C. Dataset Generation

A diversified dataset is constructed to train the proposed OFDM-based model, considering exclusively the effect of the impulse noise presence in the NB-PLC channel. The noise is modeled using the Middleton Class-A distribution [5]. This synthetic data generation strategy ensures sufficient variability for robust network training while excluding channel effects such as multipath fading or attenuation. The modeled noise is directly added to the transmitted OFDM signals to construct the noisy dataset, providing realistic training samples for the proposed impulsive noise mitigation approach.

The training dataset for the CNN is constructed as follows: A total of 72 000 observations, each corresponding to

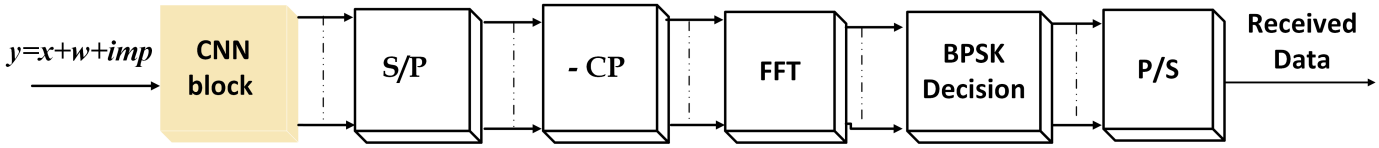


Fig. 2. Block diagram of the OFDM receiver incorporating the CNN model for impulsive noise mitigation.

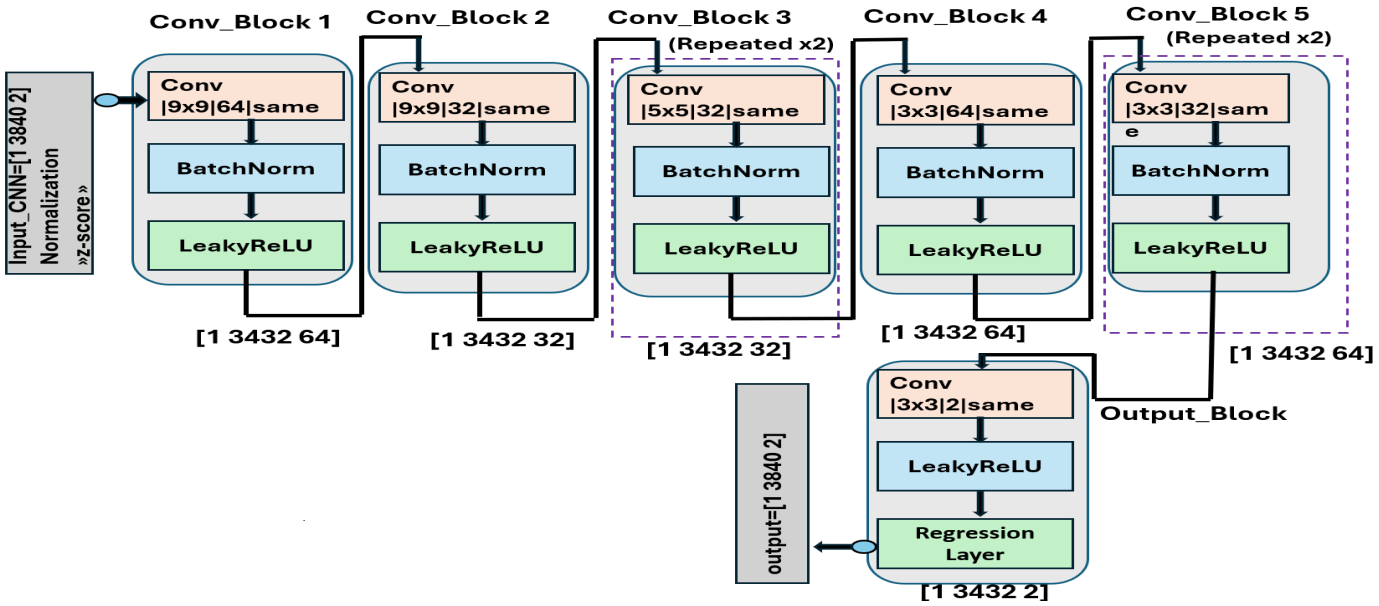


Fig. 3. CNN model architecture for impulsive noise suppression.

one OFDM frame, are generated. The dataset is organized into three blocks corresponding to different impulsive noise densities, with the Middleton Class-A parameter A set to 0.01, 0.1, and 0.5, respectively.

For each value of A, a total of 3 000 OFDM frames are generated and distributed across the SNR range from 0 to 30 dB in increments of 4 dB. This construction ensures a balanced representation of SNR conditions for each impulsive noise density and allows the network to jointly learn the variability of both A and SNR.

The impulsive noise is modeled according to the Middleton Class-A distribution with fixed parameters $\sigma_I^2 = 7.28 \times 10^{-4}$, $\Gamma = 0.01$, and $\sigma_G^2 = 7.28 \times 10^{-7}$, which were extracted from experimental measurements conducted by our research group using a Tektronix TDS3000C oscilloscope [5]. In addition, AWGN noise is added according to the target SNR.

The received noisy signals y, as defined in Eq. (3), serve as the input data. These complex-valued time-domain signals are preprocessed and represented as tensors of size $[1 \times 3432 \times 2]$. The first dimension corresponds to a single OFDM frame, the second dimension (3432) denotes the total number samples per OFDM frame including the CP, and the third dimension represents the complex signal components, with separate channels for the real and imaginary parts.

The label data consist of clean OFDM frames transmitted in the absence of impulsive noise. These reference signals are represented using the same tensor dimensions $[1 \times 3432 \times 2]$.

Consequently, the complete dataset is organized as 4D tensors of size $[1, 3432, 2, N_1]$, where $N_1 = 72000$ denotes the total number of training samples. Z-score normalization is applied to the input tensors to enhance numerical stability and accelerate network convergence.

V. Simulation Results and Discussion

To evaluate the performance of the proposed estimation approach in an OFDM-based NB-PLC system affected by impulsive noise, an extensive simulation campaign was conducted using a BPSK-OFDM transceiver compliant with the G3-PLC standard, as described in Section III. The impulsive noise samples are synthetically generated according to the Middleton Class-A model with fixed parameters $\sigma_i^2 = 7.28 \times 10^{-4}$,$\sigma_g^2 = 7.28 \times 10^{-7}$, and $\tau = 0.01$. The system performance is evaluated under various SNR conditions ranging from 0 to 30 dB and different levels of impulsive noise density.

For comparison purposes, conventional blanking and clipping schemes, were also evaluated, with optimal thresholds determined through an exhaustive search over the normalized threshold range $Th \in [0, 5]$ using a step size of $\Delta T = 0.01$. For each set of impulsive noise parameters, the threshold that minimizes the output BER is selected [25], [36].

All simulation scenarios were run on a Dell G15 5530 notebook featuring a 13th-generation Intel Core i7-13650HX CPU (14 cores, 20 logical threads, base frequency of 2.6 GHz), 32

GB of memory, and the 64-bit version of Microsoft Windows 11 Home.

A. Offline Training and Online Deployment of the Proposed CNN-Based Denoising Model

1) Offline training phase: The constructed dataset is randomly partitioned into training, validation, and testing sets with proportions of 80%, 10%, and 10%, respectively. The validation subset is solely employed to track the learning process and mitigate overfitting during training. Prior to network training, the time-domain input frames are standardized using Z-score normalization, which enforces zero mean and unit variance, thereby improving numerical stability and accelerating convergence of the CNN.

To identify the optimal CNN architecture, a tuning study is conducted by monitoring the mean squared error (MSE) loss function L_{MSE} defined in Eq. (6). The goal is to select a model configuration that achieves the lowest L_{MSE} while keeping the computational complexity within reasonable limits.

$$L_{\text{MSE}}(\Phi) = \frac{1}{K_T} \left\| \mathbf{O}_T - \mathbf{O}_T^* \right\|_2^2, \tag{6}$$

where, $\mathbf{O}_T \in \mathbb{R}^{K_T}$ denotes the complete set of target values in the training dataset, $\mathbf{O}_T^* \in \mathbb{R}^{K_T}$ represents the network's estimates, K_T is the total number of training samples, and $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_L\}$ contains all trainable parameters of the network, with L being the total number of layers.

The CNN incorporates convolution kernels of different sizes (9×9, 5×5, and 3×3) to effectively capture impulsive noise characteristics while preserving the underlying signal.

The final optimized hyperparameters, summarized in Table II, are selected to minimize the loss function and provide considerably improved training stability and generalization performance of the CNN.

TABLE II. OPTIMIZED SET OF TRAINING HYPERPARAMETERS FOR THE CNN-BASED NB-PLC DENOISING MODEL

Hyperparameter	CNN Model
Optimizer	Adam
Max Epochs	20
Initial Learning Rate	10^{-3}
L2 Regularization	10^{-4}
Mini-Batch Size	32
Input Size	Tensor $[1, 3432, 2]$
Output Size	Tensor $[1, 3432, 2]$

To assess the performance of the trained CNN, the coefficient of determination R^2 and the MSE are computed on the testing dataset, as defined in Eq. (7) and Eq. (8) [35]:

$$R^2_{\text{Testing}} = 1 - \frac{\sum_{i=1}^{K_{\text{Test}}} \left(TO_i - TO_i^*\right)^2}{\sum_{i=1}^{K_{\text{Test}}} \left(TO_i - \overline{TO}_i\right)^2}, \tag{7}$$

$$MSE_{\text{Testing}} = \frac{1}{K_{\text{Test}}} \sum_{i=1}^{K_{\text{Test}}} \left(TO_i - TO_i^*\right)^2, \tag{8}$$

where, TO_i denotes the actual target value of the i-th testing sample, TO_i^* represents the value estimated by the CNN model, $\overline{O}_{Test}$ is the mean of the target values in the testing set, and K_{Test} is the total number of testing samples.

The results show that the CNN model close to very low MSE value (7.0159×10^{-5}) and an R^2 value approaching 1 (≈ 0.91), indicating a highly accurate estimation.

Fig. 4 shows the training and validation loss evolution over 20 epochs. A rapid decrease is observed during the first epochs, indicating efficient learning of the underlying data representation. Both losses converge smoothly and remain closely aligned throughout training, demonstrating good generalization capability and the absence of overfitting. After approximately 14 epochs, the loss stabilizes around 0.15, suggesting that the model has reached convergence and additional training provides limited improvement.

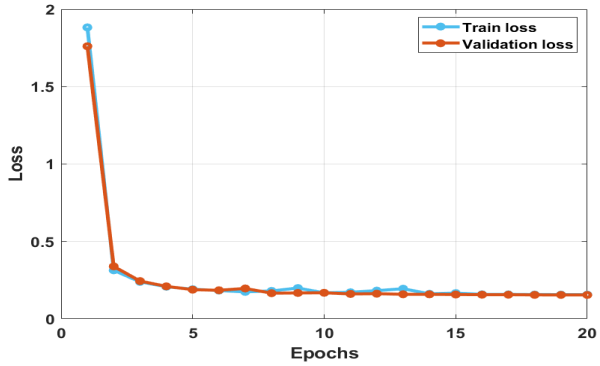


Fig. 4. Training and validation loss curves of the proposed CNN.

2) Online deployment phase: During the online testing phase, 10000 OFDM frames are transmitted under SNR conditions ranging from 0 to 30 dB with a 4 dB step. The frames are corrupted by impulsive noise realizations not used during training enabling a rigorous evaluation of the model's generalization capability under unseen noise conditions. Each transmission consists of a single OFDM frame containing 12 OFDM symbols.

B. Performance Evaluation and Analysis

The performance of the proposed CNN-based mitigation technique is evaluated in terms of MSE and BER, with results averaged over 10000 transmitted OFDM frames for each SNR value. Three distinct simulation scenarios are considered to benchmark this approach against reference schemes:

- A fixed impulsive index parameter A=0.1;
- A fixed impulsive index parameter A=0.5;
- A time-varying parameter A randomly drawn for each OFDM frame from a uniform distribution in the range [0.01, 0.5] to emulate dynamically varying impulsive noise conditions in realistic NB-PLC environments.

The performance of the proposed CNN-based denoising approach is finally compared with conventional blanking and clipping methods, where their thresholds are set to optimal values to maximize noise mitigation performance.

Fig. 5 shows the BER versus SNR for different impulsive noise mitigation methods, with the impulsive index parameter set to $A = 0.1$, the system without any mitigation suffers severe degradation, with a BER around 10^{-2} even at SNRs above 20 dB. This error floor highlights the dominant effect of unmitigated impulsive noise, which persists even when AWGN becomes negligible.

Classical methods such as Clipping and Blanking significantly improve performance. At SNR = 10 dB, Clipping reduces the BER to approximately 2×10^{-3}, while Blanking achieves 1×10^{-3}. A gain of 13 dB at a BER of 10^{-2} is observed compared to the untreated system. At SNR = 20 dB, Clipping and Blanking reach 1.2×10^{-5} and 2.31×10^{-6}, respectively. These methods show a consistent slope, demonstrating good adaptation to increasing SNR and confirming their effectiveness for attenuating impulsive noise in NB-PLC channels, at the cost of an additional processing step to determine the optimal threshold.

The proposed CNN method exhibits outstanding performance and outperforms classical techniques. At SNR = 10 dB, the CNN achieves a BER of 3×10^{-4}, corresponding to additional gain of approximately 3 dB over Clipping and 2 dB over Blanking. This demonstrates the CNN's ability to effectively learn the complex spatio-temporal features of impulsive noise and improve the performances of the system.

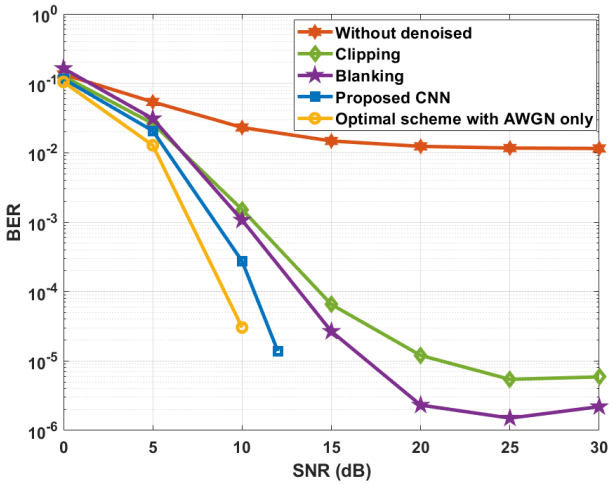


Fig. 5. BER versus SNR for different impulsive noise mitigation techniques under fixed noise density A=0.1.

Fig. 6 illustrates the average MSE versus SNR for the different tested noise mitigation approaches. The Clipping method exhibits the poorest performance, with an MSE stabilizing around 1.5×10^{-4} for SNR $\geq$ 10 dB. The Blanking technique performs better, starting at 1.2×10^{-3} at SNR = 0 dB and decreasing to approximately 1×10^{-4} for SNR $\geq$ 15 dB. However, both conventional methods exhibit an error floor, limiting their effectiveness at high SNR.

In contrast, the proposed CNN-based method demonstrates superior performance across the entire SNR range. Its MSE decreases monotonically from 2×10^{-4} at SNR = 0 dB to 2×10^{-5} at 30 dB, corresponding to an order-of-magnitude improvement. At SNR = 10 dB, the CNN achieves an MSE

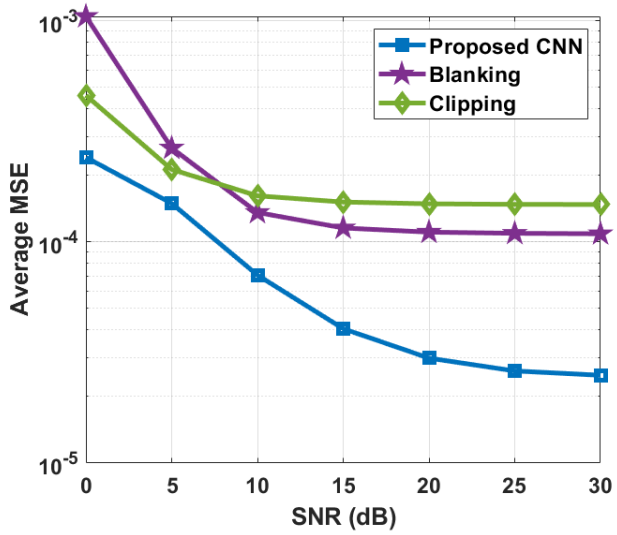


Fig. 6. Average MSE versus SNR for different impulsive noise mitigation techniques at a fixed impulsive index A=0.1.

of approximately 7×10^{-5}, outperforming the Blanking and Clipping methods, which yield MSE values of 1.3×10^{-4} and 1.6×10^{-4}, respectively. Notably, the absence of an error floor indicates that the CNN continues to benefit from increasing SNR for improved signal reconstruction, unlike conventional threshold-based methods, whose performance tends to saturate at high SNR.

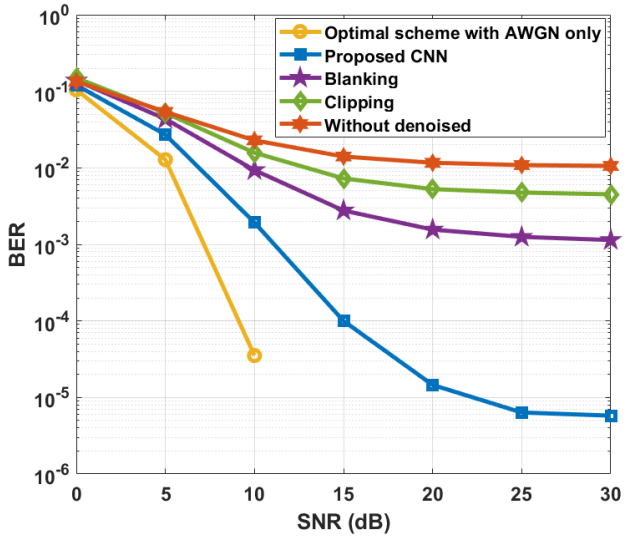


Fig. 7. BER versus SNR for different impulsive noise mitigation techniques at a fixed impulsive index A=0.5.

Next evaluation deals with the scenario of the impulsive noise with a fixed parameter $A = 0.5$, which result a severely disturbed NB-PLC environment. Fig. 7 illustrates the BER versus SNR for the different tested noise mitigation approaches. First, the system without any denoising method to BER higher than 10^{-2} across the entire SNR range. The Clipping method reduces the BER to approximately 4×10^{-3} at 30 dB, a gain of

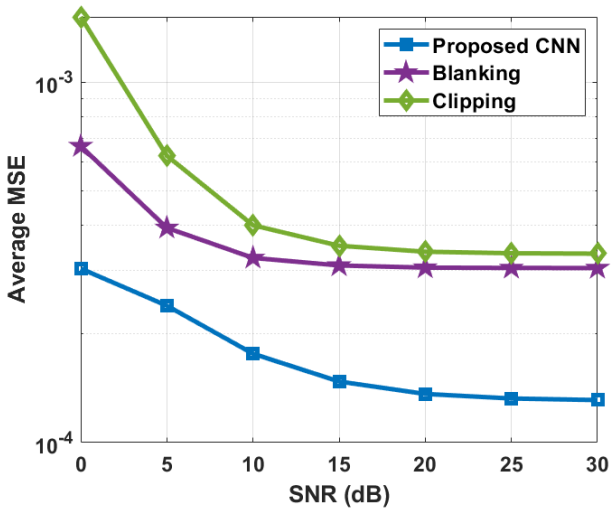


Fig. 8. Average MSE versus SNR for different impulsive noise mitigation techniques at a fixed impulsive index A=0.5.

approximately 6 dB is observed at a BER of 10^{-2}. Blanking provides greater improvement, with a BER of 1×10^{-2} at 10 dB and 1×10^{-3} at 30 dB, for a BER of 10^{-2}, provides an improvement of nearly 10 dB compared to the untreated system.

In contrast, the proposed CNN-based method achieves significantly lower BER than conventional techniques. At a target BER of 10^{-3}, a performance gain of nearly 18 dB is observed compared to the blanking method. Moreover, at a BER of 10^{-2}, the proposed approach provides a gain of approximately 23 dB relative to the untreated system, about 7 dB compared to clipping, and nearly 3 dB compared to blanking. At SNR = 20 dB, the CNN BER decreases to 10^{-5}. These results indicate that the CNN is able to exploit the complex temporal and statistical correlations of impulsive noise, unlike simple threshold-based methods.

In terms of MSE, the CNN-based technique clearly surpasses the benchmark methods, as illustrated in Fig. 8. It achieves an MSE of 1.7×10^{-4} at 10 dB decreasing to 1.2×10^{-4} at 30 dB, without any saturation. Alternatively, the Clipping method shows an MSE floor around 3.5×10^{-4} for SNR $\geq$ 15 dB, while Blanking reaches 3.2×10^{-4} at 10 dB and stabilizes near 3.0×10^{-4} for SNR $\geq$ 15 dB.

Fig. 9 presents the BER under a realistic NB-PLC scenario, where the parameter A varies dynamically from frame to frame, randomly drawn from a uniform distribution over the set $0.01, 0.5$.

In time-varying noise scenario, the CNN-based mitigation technique shows its advantage and demonstrates high robustness to fluctuations in impulsive index A. At SNR $= 10$ dB, the BER reaches 2.5×10^{-3}, corresponding to an SNR gain of approximately 5 dB compared to the Blanking method and 77 dB compared to Clipping. Moreover, the BER decreases monotonically to approximately 10^{-7} at SNR $= 30$ dB, while Blanking and Clipping saturate at around 2×10^{-4} and 3×10^{-3}, respectively. This result highlights the adaptability of the CNN, which learns to distinguish and effectively handle different levels of impulsive noise.

Finally, Fig. 10 presents the comparison of different methods in terms of average MSE. Similarly to previous results, the CNN-based methods outperforms the other noise mitigation techniques. It maintains a continuous MSE reduction from 2.8×10^{-4} at SNR = 0 dB to 6.5×10^{-5} at SNR = 30 dB. At high SNR, the proposed technique exhibits an MSE floor of 1.2×10^{-4}, whereas Blanking and Clipping do not decrease below 3×10^{-4} and 3.5×10^{-4}, respectively.

To further position the proposed approach with respect to existing DL–based impulsive noise mitigation techniques, Table III summarizes representative studies from recent years discussed above.

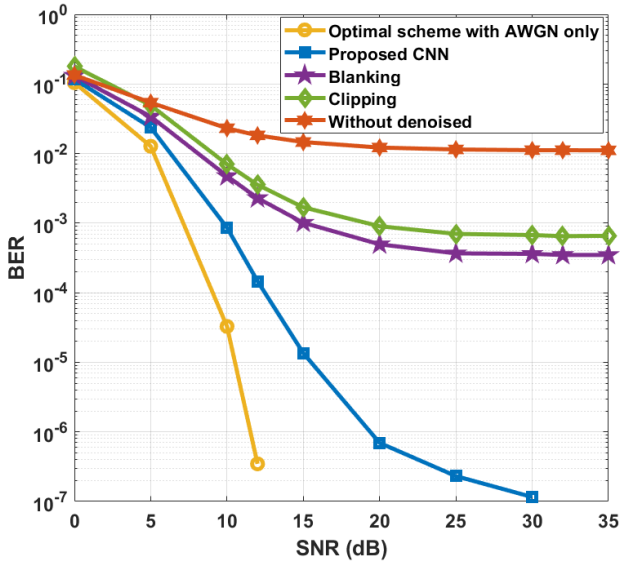


Fig. 9. BER versus SNR for different impulsive noise mitigation techniques with random noise density.

From the comparison presented in Table III, it can be observed that end-to-end time-domain denoising approaches offer an attractive trade-off between performance and computational efficiency. In this context, the proposed method avoids intermediate parameter estimation stages by directly learning the mapping between noisy and clean signals, while preserving robust generalization across varying impulsive noise densities. Moreover, the supervised learning framework relies on synthetic Middleton Class-A noise samples calibrated using experimental measurements, which facilitates a practical and reproducible implementation.

VI. Conclusion

This study presented a CNN-based approach for direct time-domain impulsive noise mitigation in NB-PLC OFDM systems compliant with the G3-PLC standard. The proposed method eliminates the need for threshold optimization, noise parameter estimation, or additional null subcarriers, as required by conventional cancellation techniques. Trained on synthetic data generated using Middleton Class-A model calibrated with experimental measurements, the CNN effectively suppresses

TABLE III. SUMMARY OF DEEP LEARNING ARCHITECTURES AND THEIR ROLES FOR IMPULSIVE NOISE MITIGATION IN OFDM-BASED PLC SYSTEMS

Reference	Year	DL Architecture and Role	System Description and Noise Model	Key Performance Results
Barazideh et al.[38]	2019	DNN (Fully Connected, 2 hidden layers: 20+10 neurons, ReLU activations; Sigmoid output) Detects impulsive samples using statistical features (ROAD, median deviation) for subsequent blanking.	OFDM-based systems; Noise: Bernoulli–Gaussian and Middleton Class-A ($\epsilon, A \in [0.04, 0.2]$)	Up to 2 dB SNR gain over blanking/clipping at BER 10^{-3}
He et al.[25]	2023	MaxPoolCNN (CNN with Max-Pooling layers) Extracts local features and adaptively estimates the optimal threshold values for the Deep Clipping method.	OFDM-based broadcasting and PLC system; Noise: Middleton Class-A ($A = 0.1$), Poisson arrival	Average interference reduction 15.1 dB (up to 19.6 dB); improved BER vs fixed threshold for blanking/clipping
Yang et al.[27]	2024	LSTM architecture Models temporal dependencies in impulsive noise to dynamically adjust thresholds.	OFDM-based smart grid PLC system; Noise: Bernoulli–Gaussian ($p = 0.1, 0.2$)	BER reduction 87.85% ($p = 0.1$) and 89.72% ($p = 0.2$) at 8 dB SNR vs blanking
Ouyang et al.[28]	2024	MFSDF-Net (CNN multi-feature fusion, parallel convolutions 1×1,1×3,1×5,1×7) Uses parallel convolutional kernels to extract and fuse features from multiple signal domains, effectively suppressing impulsive noise.	OFDM-based PLC; Noise: Bernoulli–Gaussian (p), dataset generated at SINR = -15 dB	BER 10^{-5} at 18 dB SNR, 8 dB improvement over Adaptive Parameter / GMM
Zhu et al.[26]	2025	1D-UNet (CNN encoder-decoder) Self-supervised denoising; learns signal reconstruction from noisy input only via global mask mapper.	OFDM-based MIMO-PLC system; Noise: Bernoulli–Gaussian ($p = 0.01, 0.05$)	At BER 10^{-6}, gains of 3 dB over Noise2Void, 4 dB over Zero-Shot Noise2Noise, 7 dB over DeepClipping
This work	**2026**	Multi-layer CNN (5 convolutional blocks: Conv + BatchNorm + Leaky ReLU) Direct end-to-end regression of clean signal from noisy input; robust to dense impulsive noise	NB-PLC OFDM system according to G3-PLC standard; Noise: Middleton Class-A ($A = \{0.01, 0.1, 0.5\}$)	Up to 18 dB SNR gain over blanking at BER 10^{-3} for $A = 0.5$; additional 3 dB over clipping and 2 dB over blanking at 10 dB SNR for $A = 0.1$; MSE = 7.01×10^{-5}

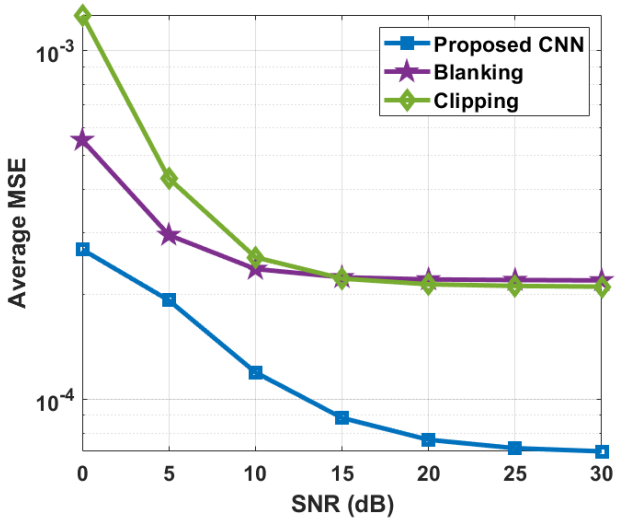


Fig. 10. Average MSE versus SNR for different impulsive noise mitigation techniques with random noise density.

impulsive noise while preserving useful signal components under varying noise densities.

Simulation results demonstrate that the proposed approach achieves significant improvements in BER and MSE over SNR range and under various impulsive noise conditions compared to conventional methods. These findings confirm that data-driven learning-based approaches consitutues a promising alternative to classical impulsive noise mitigation techniques for NB-PLC systems.

Future work will focus on integrating channel estimation and equalization within the proposed architecture for joint optimization, as well as investigating hybrid schemes that combine the CNN with channel coding techniques such as LDPC or polar codes to further enhance reliability.

REFERENCES

[1] G. López, J. Matanza, D. Vega, M. Castro, A. Arrinda, J. Moreno, and A. Sendin, "The Role of Power Line Communications in the Smart Grid Revisited: Applications, Challenges, and Research Initiatives," *IEEE Access*, vol. pp. 1–1, Jul. 2019. doi: 10.1109/GLOCOM.2014.7037191.

[2] M. Luka, S. Pallam, I. Thuku, and U. Uyoata, "Narrowband Power Line Communication for Smart Grid," *International Journal of Scientific and Engineering Research*, vol. 6, no. 8, pp. 876–880, Aug. 2015. [Online]. Available: https://www.ijser.org

[3] H. Gassara, F. Rouissi, and A. Ghazel, "Statistical characterization of the indoor low-voltage narrowband power line communication channel," *IEEE Trans. Electromagn. Compat.*, vol. 56, no. 1, pp. 123–131, Feb. 2014, doi: 10.1109/TEMC.2013.2272759.

[4] F. Rouissi, A. J. H. Vinck, H. Gassara, and A. Ghazel, "Statistical characterization and modelling of impulse noise on indoor narrowband PLC environment," in *Proc. IEEE Int. Symp. Power Line Communications and its Applications (ISPLC)*, Madrid, Spain, 2017, pp. 1–6. [Online]. Available: doi:10.1109/ISPLC.2017.7897119.

[5] F. Rouissi, A. J. H. Vinck, H. Gassara, and A. Ghazel, "Improved Impulse Noise Modeling for Indoor Narrow-Band Power Line Communication," *AEU – International Journal of Electronics and Communications*, vol. 103, Mar. 2019. doi: 10.1016/j.aeue.2019.02.019.

[6] M. Zimmermann and K. Dostert, "An analysis of broadband noise scenario in powerline networks," in *Proc. IEEE International Symposium on Power Line Communications and Applications (ISPLC)*, 2000.

[7] M. Marey and H. Steendam, "Analysis of the Narrowband Interference Effect on OFDM Timing Synchronization," *IEEE Transactions on Signal Processing*, vol. 55, no. 9, pp. 4558–4566, Sep. 2007. doi: 10.1109/TSP.2007.896020.

[8] A. Mehboob, L. Zhang, and J. Khangosstar, "Adaptive impulsive noise mitigation using Multi Mode Compressive Sensing for powerline communications," in *Proc. IEEE International Symposium on Power Line*

Communications and Its Applications (ISPLC), 2012, pp. 368–373. doi: 10.1109/ISPLC.2012.6201304.

[9] International Telecommunication Union (ITU), "Narrowband orthogonal frequency division multiplexing power line communication transceivers for G3-PLC networks," *ITU-T Recommendation G.9903*, 2017. [Online]. Available: https://www.itu.int/rec/T-REC-G.9903-201708-I/en

[10] A. Sanz, "I3: ITU-T G.9904 (PRIME 1.4) evolution of implementation and certification of NB-PLC," in *Proc. IEEE International Symposium on Power Line Communications and Its Applications (ISPLC)*, 2016.

[11] J. Matanza, S. Alexandres, and C. Rodriguez-Morcillo, "Performance evaluation of two narrowband PLC systems: PRIME and G3," *Computer Standards & Interfaces*, vol. 36, no. 1, pp. 198–208, 2013. doi: 10.1016/j.csi.2013.05.001.

[12] Š. Matějka and P. Hladík, "Initial tests of DVB-T receivers on tolerance to impulsive interference," in *Proc. 21st International Conference Radioelektronika*, 2011, pp. 1–4. doi: 10.1109/RADIOELEK.2011.5936471.

[13] P. Torio and M. G. Sanchez, "Cell interleaving against impulsive noise in OFDM," *IEEE Transactions on Consumer Electronics*, vol. 58, no. 2, pp. 269–273, 2012. doi: 10.1109/TCE.2012.6227422.

[14] R. G. Gallager, "Low-density parity-check codes," *IEEE Transactions on Information Theory*, 1962. [Online]. Available: IEEE Xplore.

[15] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. IEEE International Conference on Communications (ICC)*, 1993. [Online]. Available: IEEE Xplore.

[16] E. Arıkan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, 2009. [Online]. Available: IEEE Xplore.

[17] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, 1998. [Online]. Available: IEEE Xplore.

[18] F. H. Juwono, Q. Guo, D. Huang, and K. P. Wong, "Deep Clipping for Impulsive Noise Mitigation in OFDM-Based Power-Line Communications," *IEEE Transactions on Power Delivery*, vol. 29, no. 3, pp. 1335–1343, 2014. doi: 10.1109/TPWRD.2013.2294858.

[19] M. Elgenedy, M. Sayed, M. Mokhtar, M. Abdallah, and N. Al-Dhahir, "Interference mitigation techniques for narrowband powerline smart grid communications," in *Proc. IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2015, pp. 368–373. doi: 10.1109/SmartGridComm.2015.7436328.

[20] Y.-R. Chien, S. S.-D. Xu, and S.-H. Lu, "Cyclostationary impulsive noise mitigation algorithm for narrowband powerline communications," *Journal of the Franklin Institute*, vol. 357, no. 1, pp. 687–703, 2020. doi: 10.1016/j.jfranklin.2019.10.026.

[21] Y. G. Yoo and J. H. Cho, "Asymptotic Analysis of CP-SC-FDE and UW-SC-FDE in Additive Cyclostationary Noise," in *Proc. IEEE International Conference on Communications (ICC)*, 2008, pp. 1410–1414. doi: 10.1109/ICC.2008.273.

[22] G. Caire, T. Y. Al-Naffouri, and A. K. Narayanan, "Impulse noise cancellation in OFDM: an application of compressed sensing," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2008, pp. 1293–1297. doi: 10.1109/ISIT.2008.4595196.

[23] J. Lin, M. Nassar, and B. L. Evans, "Impulsive Noise Mitigation in Powerline Communications Using Sparse Bayesian Learning," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1172–1183, 2013. doi: 10.1109/JSAC.2013.130702.

[24] Z. Liu, Y. Li, and S. Shi, "Impulsive noise suppressing method in power line communication system using sparse iterative covariance estimation," *Radio Science*, vol. 57, e2022RS007424, 2022. doi: 10.1029/2022RS007424.

[25] Y. He, C. Zou, D. Li, R. Yao, F. Yang, and J. Song, "Adaptive Impulsive Noise Suppression: A Deep Learning-Based Parameters Estimation Approach," *IEEE Transactions on Broadcasting*, vol. 69, no. 2, pp. 505–515, 2023. doi: 10.1109/TBC.2022.3224249.

[26] E. Zhu, Y. Ren, R. Li, S. Ouyang, Y. Ma, X. Yang, and G. Liu, "UNet-INSN: Self-Supervised Algorithm for Impulsive Noise Suppression in Power Line Communication," *Applied Sciences*, vol. 15, no. 16, art. no. 9101, 2025. doi: 10.3390/app15169101.

[27] S. Ouyang, G. Liu, T. Huang, Y. Liu, W. Xu, and Y. Wu, "Impulsive Noise Suppression Network for Power Line Communication," *IEEE Communications Letters*, vol. PP, pp. 1–1, Nov. 2024. doi: 10.1109/LCOMM.2024.3466893.

[28] G. Yang, Y. Qian, Z. Wang, X. Zhou, and W. Wu, "An intelligent impulsive noise mitigation with deep learning method," *International Journal of Mechanical System Dynamics*, 2024. [Online]. Available: Semantic Scholar.

[29] C. L. Fan, "Multiscale feature extraction by using convolutional neural network: Extraction of objects from multiresolution images of urban areas," *ISPRS International Journal of Geo-Information*, vol. 13, no. 1, Art. no. 5, 2024. [Online]. Available: MDPI.

[30] S. Sadrizadeh, H. Otroshi, and F. Marvasti, "Impulsive noise removal via a blind CNN enhanced by an iterative post-processing," *Signal Processing*, vol. 192, Art. no. 108378, 2021. [Online]. Available: doi:10.1016/j.sigpro.2021.108378.

[31] O. P. Aiyelabowo, C. K. Ng, and N. K. Noordin, "Power line communication (PLC) impulsive noise mitigation: A review," *Journal of Information Engineering and Applications*, vol. 4, no. 10, 2014, ISSN 2224–5782.

[32] M. Hussain, H. Shakir, and H. Rasheed, "Deep learning approaches for impulse noise mitigation and classification in NOMA-based systems," *IEEE Access*, vol. 9, pp. 143836–143846, 2021. doi: 10.1109/ACCESS.2021.3121533.

[33] C.-L. Fan, "Multiscale feature extraction by using convolutional neural network: Extraction of objects from multiresolution images of urban areas," *ISPRS International Journal of Geo-Information*, vol. 13, no. 1, art. 5, 2024. [En ligne]. Disponible : https://www.mdpi.com/2220-9964/13/1/5

[34] S. Sadrizadeh, H. Otroshi, and F. Marvasti, "Impulsive noise removal via a blind CNN enhanced by an iterative post-processing," *Signal Processing*, vol. 192, art. 108378, Nov. 2021. [En ligne]. Disponible : https://doi.org/10.1016/j.sigpro.2021.108378

[35] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, p. e623, Jul. 2021. [En ligne]. Disponible : https://doi.org/10.7717/peerj-cs.623

[36] A. A. G. Liong, F. H. Juwono, L. Gopal, W. R. Chiong, and Y. Rong, "Multiple blanking preprocessors for impulsive noise mitigation in OFDM-based power-line communication systems," *International Journal of Electrical Power & Energy Systems*, vol. 130, p. 106911, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0142061521001514

[37] N. Sadeghi and M. Azghani, "Deep learning based channel estimation in PLC systems," *Annals of Telecommunications*, vol. 80, no. 7–8, pp. 581–590, 2024. doi: 10.1007/s12243-024-01051-3. :contentReference[oaicite:0]index=0

[38] R. Barazideh, S. Niknam, and B. Natarajan, "Impulsive Noise Detection in OFDM-based Systems: A Deep Learning Perspective," arXiv:1901.00447v1 [eess.SP], 2 Jan. 2019.

Leveraging Attention Mechanism and Class Weighting for Legal Event Detection in Chinese Text

Jinhong Hu, Shaidah Jusoh*
School of Artificial Intelligence and Robotics,
Xiamen University Malaysia, Sepang, Selangor, Malaysia

***Abstract*—Event detection is an information extraction task that involves extracting specified event types from textual sequences. Currently, most event detection studies focus on English corpora; there is a lack of exploration in other linguistic contexts. Thus, a study on event detection in the Chinese corpus is essential. Sequence-based event detection has been extensively studied in the past, and many studies have utilized high-performance neural network models, such as traditional recurrent neural networks. This study aims to enhance the performance of sequence models by altering the base model, Bidirectional Long Short Term Memory (BiLSTM), to a Bidirectional Gated Recurrent Unit (BiGRU) and incorporating multi-head attention mechanisms, Conditional Random Fields (CRF), and class weights. These modifications not only improve the model's accuracy but also enhance computational efficiency by reducing the number of parameters relative to large pre-trained models such as Bidirectional Encoder Representations from Transformers (BERT). The experimental findings demonstrate that the proposed model's modification achieves an F1 Score of 83.55 for the micro standard and 78.07 for the macro standard. This presents a substantial improvement over the baseline, delivering performance nearly on par with state-of-the-art BERT-based models on the same dataset, while requiring significantly fewer parameters.**

***Keywords*—Event detection; information extraction; Chinese corpus; recurrent neural network**

I. Introduction

In the era of rapid advancements in big data and artificial intelligence, linguistic texts have become the foundation of nearly all the information available on the internet. Consequently, there is a growing demand for extracting more meaningful insights from these texts. Information extraction is a technique within the field of natural language processing (NLP) that aims to uncover valuable information from text. Event detection is a crucial subtask of information extraction, focusing on the identification of events mentioned in textual data. Due to the unique syntactic structures of different languages, the handling of details in natural language tasks varies. Despite the advancement of neural network technology and years of development, event detection still faces multiple challenges, including contextual dependency, event diversity, and sparse annotated data. These challenges constrain the accuracy and generalization ability of the model, particularly in complex contexts [1]. In response to these challenges, there have been emerging new approaches aimed at addressing event extraction problems, primarily categorized into two major classes: generation-based and classification-based methods. Some specific techniques have improved models, such as models based on attention mechanisms, transfer learning, and methods integrating external knowledge, which have, to some extent, alleviated the difficulties faced in event extraction [2], [3].

The motivation of this study is to develop an event detection model for Chinese legal texts. The benchmark dataset, LEVEN, a large-scale Chinese legal event detection dataset released in 2022 [4] and currently accessible through the GitHub site, has been utilized. This dataset is currently the largest of its kind for event detection in the Chinese legal domain, comprising approximately 60,000 sentences and 108 different event types. Despite the dataset's focus on the legal domain and its substantial sample size, high-performing sequence labeling models such as BERT and BiLSTM typically achieve F1 scores of around 80% on this dataset [4]. Thus, the primary objective of this study is to improve the performance of the model on the LEVEN dataset by refining the framework of the BiLSTM-based sequence labeling model. Specifically, the study explored which key modifications can effectively enhance the model's performance on this dataset.

This study set boundaries and focused exclusively on mathematical knowledge and model code implementation related to artificial intelligence and NLP. Mathematical knowledge was employed to elucidate the underlying principles of the models, while the implementation of model code utilized advanced API frameworks. A total of nine different models based on BiLSTM were implemented and trained. The enhancements on BiLSTM include modifying the basic architecture to BiGRU, introducing different multi-head attention mechanisms, replacing softmax with CRF, and adding weight adjustment to classes based on data distribution. The enhancement resulted in several beneficial improvements. Although it is possible to train more models, doing so would require an amount of time for parameter adjustment and training, yet it would not guarantee superior outcomes. Additionally, this study does not involve cross-lingual event extraction or the processing of multimodal data. It solely focuses on events within a specific linguistic context. And the enhancement of the model primarily stems from both the dataset and the model itself. All implemented models were analyzed against one another to identify the most effective approach.

This study is structured as follows: Section II addresses related topics, Section III outlines the methodology and experimental design, Section IV provides the analysis of results, and Section V concludes the study.

II. Related Topics

Topics related to this study are briefly presented in this section. Firstly, sequence labeling-based event detection mod-

*Corresponding author.

els are introduced, and relevant examples are provided. Next, the impact of data imbalance in corpus datasets are addressed, along with methods for handling it. Classic sequence models such as the Recurrent Neural Network (RNN) and its variations, including bidirectional models, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) are explored. Finally, attention mechanisms and Conditional Random Fields (CRF) in sequence labeling are discussed.

A. Sequence Labeling

Sequence labeling is a text processing task for assigning appropriate labels to each token in an input sequence [5]. These labels can be used to represent information about parts of speech, named entities, emotional tendencies, and word sense disambiguation. It has wide applications involving entity recognition, part-of-speech tagging, sentiment analysis, entity relationship extraction, and syntactic parsing.

Event detection can be considered as a sequence labeling task, where the model processes a text sequence to generate a corresponding event label sequence. Essentially, it involves detecting trigger words with specified event labels in the text sequence. Multiple event trigger tokens may appear in a single input sentence [6]. In the example shown in Fig. 1, the input sequence is "John opens door, then sleeps", and the output sequence highlights two trigger words, specifically the first and the sixth tokens, "open" and "sleep". With these two trigger words and contextual information, the model can determine the corresponding event types as "open" and "sleep".

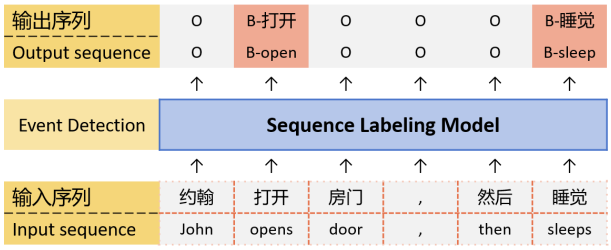


Fig. 1. Event detection based on sequence labeling.

B. Data Imbalance

Data imbalance issues are quite common in corpus datasets. In the event detection task, data imbalance refers to a large difference in the number of samples across event classes. It negatively impacts model training, as the model tends to fit the dominant classes while neglecting the less frequent classes. To address data imbalance issues in NLP, many methods have been considered, like sampling, which increases the number of samples in the minority classes [7]. Alternatively, setting sample weights assigns higher weights to imbalanced classes, encouraging the model to learn the minority classes better [8].

C. Recurrent Neural Network

A recurrent neural network is a neural network model for handling sequential data by recurrent connections, which allows information to persistently propagate within the network [9]. Due to its capability to process sequences of varying lengths, RNN excels in solving event detection based on sequence labeling.

1) Bidirectional: In a traditional RNN, only forward input can be accepted, resulting in the output being more influenced by the most recent input data, which hinders its effectiveness in processing contextual information. To address this issue, the Bidirectional Recurrent Neural Network (BRNN) model was introduced. BRNN can utilize both past and future context information in two directions when processing sequential input [10]. This architecture is beneficial in alleviating the long-range dependency problem and has improved RNN's performance.

2) Long Short-Term Memory: LSTM is another variant of RNN, which was designed to address the issues of gradient vanishing and exploding gradients [11]. LSTM achieves better performance in capturing long-term dependency by introducing a mechanism that involves gates and cell states to maintain and control internal memory. These three gates include the input gate, the forget gate, and the output gate.

3) Gated Recurrent Unit: GRU is also a variant of RNN that emerged after LSTM. It aims to address the issue of long-term dependency. The gate mechanism of GRU includes the update gate and reset gate, which control the flow and retention of information [12]. Comparing gate mechanisms, the structure of GRU is simpler than that of LSTM.

D. Self-Attention Mechanism

The attention mechanism was introduced to enhance the model's ability to process sequential data in NLP [13]. It is also useful for event detection models based on sequence labeling. The key idea is that the model can dynamically focus on different parts of the input, rather than distributing fixed weights when processing an input sequence. One of the most popular attention mechanisms is the self-attention mechanism. This mechanism is currently widely used in various sequence-to-sequence tasks such as machine translation and text generation [14]. It enables the model to capture context and dependencies within the sequence better.

E. Conditional Random Field

For event detection models based on sequence labeling, softmax can be used in the inference layer to obtain final label sequences. It is a common normalization exponential function used to transform a set of values into a probability distribution, where each value represents a potential class score. However, the probability labels obtained by softmax at different positions are treated as mutually independent, which may cause a negative impact. Conversely, CRF, as an undirected probabilistic graphical model, is capable of efficiently capturing label dependencies and contextual information compared with softmax in sequence labeling [15]. Linear CRF is a frequently employed choice in sequence labeling tasks [16].

III. Methodology

This study has been conducted using the programming language Python for data processing and model training. The data processing involves extracting the required data from JSON files through the Python JSON library, and Chinese text tokenization is performed by Jieba (Chinese NLP library) [17]. PyTorch was used to develop neural networks and perform training. The research phases are illustrated in Fig. 2. The flow

begins with data gathering, continues through data processing, model implementation, and experimentation, and concludes with result analysis.

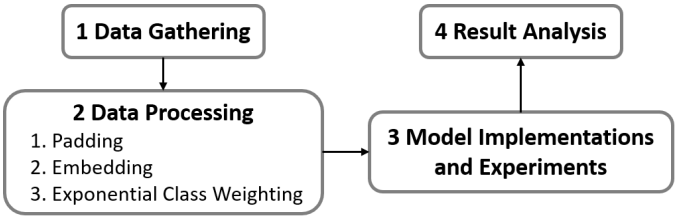


Fig. 2. Flowchart of the research framework.

A. Data Gathering

The LEVEN Chinese legal event detection dataset was obtained, comprising roughly 60,000 training samples across 108 legal event types. Each sample is a complete and lengthy sentence that contains multiple event triggers [18]. Fig. 3 demonstrates a sample of a JSON file that contains the title of the text, the unique ID, as well as the content of the text, which includes multiple sentences with corresponding tokens.


```
{
  "title": "罗华灵罗某某盗窃一审刑事判决书",
  "id": "73887dd6858d4939a66eca63faeeb968",
  "crime": "盗窃罪",
  "case_no": "（2013）珠香法刑初字第1844号",
  "content": [{
      "sentence": "2013年9月9日8时许，被告人罗华灵罗某某伙同陈英陈某把（另案处
        理），在珠海市香洲区南屏镇溢多利公司，由陈英陈某把望风，被告人罗华灵罗某某
        从该公司宿舍二楼窗户爬进205房，盗走被害人赵某一部Ipad4平板电脑和一部佳能
        Ixus105相机一部",
      "tokens": [ "2013", "年", "9", "月", "9", "日", "8", "时许", ",", "被
        告人", "罗华灵罗", "某某", "伙同", "陈英", "陈某", "把", "（", "另案处
        理", "）", ",", "在", "珠海市", "香洲区", "南屏镇", "溢", "多利", "公
        司", ",", "由", "陈英", "陈某", "把", "望风", ",", "被告人", "罗华灵
        罗", "某某", "从", "该", "公司", "宿舍", "二楼", "窗户", "爬", "进",
        "205", "房", ",", "盗走", "被害人", "赵某", "一部", "Ipad4", "平板",
        "电脑", "和", "一部", "佳能", "Ixus105", "相机", "一部" ]
    },{...},{...}],
  "events": [{
      "id": "41f833031bfb4f31bfc0cb50e9b9653d",
      "type": "入室/入户",
      "type_id": 95,
      "mention": [{"trigger_word": "进","sent_id": 0,"offset":
        [44,45],"id": "dbd9812f6da64991a0cfaf6cd36a9d97"}]
    },{...},{...},{...},{...},{...},{...},{...}],
  "negative_triggers": [{"trigger_word": "伙同","sent_id": 0,"offset":
    [12,13],"id": "89c01371f15c410cb987e575f4ca9e99"
    },{...},{...},{...},{...},{...},{...},{...},{...},{...},{...},{
      ...},{...}]
}
```


Fig. 3. An example of training data in the JSON file.

The sentence in the sample contains a close number of negative and positive triggers. During the training process, a single sentence is used as one piece of data instead of multiple sentences. The aim is to reduce the length of the input sequence and alleviate the model's computational burden.

B. Data Processing

Tokenization is an initial step in data processing. This step has been employed for both training data and testing data. It is worth noting here that the LEVEN dataset is exceptionally clean and does not require any cleaning operations. To ensure the completeness of the data information input to the model, we do not employ stop word removal, as stop words may have some semantic information for Chinese. Padding, embedding, and exponential class weighting utilized in the proposed models will be discussed in the following subsections.

1) Padding: In this study, the sequence labeling model requires that the input data fed into the model have the same length. As shown in the length distribution chart of the data samples in Fig. 4, sentence lengths range from 25 to 350 tokens, with most clustering around 50 and only a few exceeding 150. In the experiment, the model's input length was set to the maximum length observed in the dataset, and shorter samples were padded to reach this specified length.

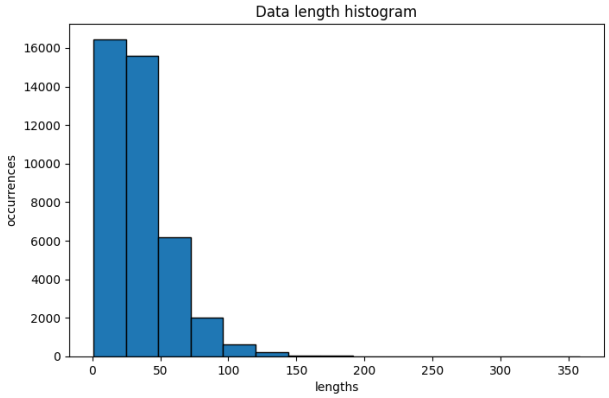


Fig. 4. Data length distribution.

Readers might wonder whether setting the input length to the maximum significantly increases training time or reduces performance. In PyTorch's RNN-based models, however, the actual length of each input can be specified, and computations for the padding portion are ignored, thereby preventing additional training overhead. Furthermore, since the test data also contains longer samples, the padding length was set to the maximum to ensure that no test input would be truncated. In sequence labeling tasks, truncation directly results in the model predicting only the first half of the data.

2) Embedding: The word embeddings used in this experiment were not trained from scratch; instead, pre-trained word embeddings [19] were utilised. These pre-trained word embeddings were trained on Baidu Encyclopedia data, and a vocabulary of 60,000 tokens was included, with each token having an embedding dimension of 300. Once the word embedding matrix is converted into a NumPy array, it occupies approximately 700 MB of memory.

For each processed text data with completed padding, it tries to look up in the word embedding matrix to obtain the corresponding word vector. Rather than pre-converting all training data into word vector format and loading it into memory, which would be impractical due to the significant space requirements, this study uses the PyTorch DataLoader API. This allows the model to dynamically load and pre-process data as needed during the training process. It's important to note that in this experiment, the pre-trained word embedding matrix was not frozen but was adjusted concurrently during the model training process.

3) Exponential class weighting: In the training data set, positive samples accounted for only 24.8% of the total samples. Negative samples refer to the instances where the event did not occur, and the positive samples encompass 108 event classes. The training dataset exhibits a pronounced class imbalance,

where the class with the fewest samples contains only 5 instances, while the category with the most samples contains around 6000, as shown in Fig. 5.

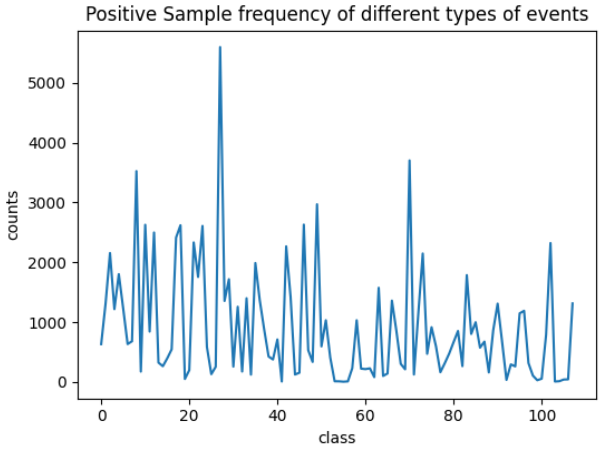


Fig. 5. Distribution of samples across different classes.

To balance the training data, it is necessary to adjust the weights of different data categories. In this experiment, an exponential weighting of class sample quantities was employed, which is defined as follows:

$$w_i = {c_i}^{-\alpha} = \frac{1}{{c_i}^{\alpha}} \tag{1}$$

where, w_i is the weight of a specific class, c_i is the sample quantity of the class and α is a hyperparameter ranging from 0 to 1. From the given formula, it can be observed that if a class has a larger sample quantity, it will receive a smaller weight, while classes with fewer samples will receive larger weights. This allows the model to pay more attention to classes with fewer samples during training. When choosing the value of α, a balanced consideration is required. If α approaches 0, the weights for all classes tend to be close to 1, effectively resulting in no weight adjustment. If α approaches 1, classes with fewer samples will receive significantly larger weights compared to classes with more samples, potentially leading to overfitting. Considering the experiment's requirements, α was ultimately set to 0.1 to balance the model's focus on different class samples. Thus, in this study, α is set to 0.1; the larger the sample size of a class, the smaller its weight.

C. Model Implementations and Experiments

A total of nine models (see Table I) with various enhancements based on BiLSTM were implemented and experimented with in this study. These enhancements include modifying the basic architecture to BiGRU, introducing different multi-head attention mechanisms, replacing softmax with CRF, and adding weight adjustment to classes based on data distribution. Each model employed dropout [20]. The dropout rate was set to 0.3 or 0.4, adjusted based on the degree of model overfitting. The loss function is the multi-class cross-entropy loss function. The models were all compiled using the Adam optimizer [21].

Table I provides information on the nine models. The inference layer of BiLSTM and BiGRU is softmax, while in other models, softmax was replaced with CRF to enhance the dependency information between labels. The A+BiGRU+CRF model introduced the first version of attention, which is self-attention after the embedding layer, before the BiGRU layer. The BiGRU+A+CRF model introduced the second version of attention, which is self-attention after BiGRU, before the CRF. The BiGRU+EA+CRF introduced the third version of attention, which is multi-head attention, after the BiGRU, before the CRF. In BiGRU+A+CRF+W, the W represents the utilization of exponential class weights.

TABLE I. NAMES AND DETAILS OF MODELS

No.	Model	Detail
1	BiLSTM	BiLSTM Model+Softmax
2	BiGRU	BiGRU Model+Softmax
3	BiLSTM+CRF	BiLSTM Model+CRF
4	BiGRU+CRF	BiGRU Model+CRF
5	A+BiGRU+CRF	First Version Attention+BiGRU+CRF
6	BiGRU+A+CRF	BiGRU+Second Version Attention+CRF
7	BiGRU+EA+CRF	BiGRU+Third Version Attention+CRF
8	BiGRU+A+CRF+W	BiGRU+A+CRF with Class Weight
9	BiGRU+EA+CRF+W	BiGRU+EA+CRF with Class Weight

All models used the same dataset. The training dataset comprises 395,322 samples, while the validation dataset contains 92,451 samples, representing approximately 20% of the total data. In this experiment, the default validation dataset served as the testing set and did not participate in model training. The graph in Fig. 6 illustrates the percentage distribution of positive samples from different classes in the training and testing datasets. The class distribution in the training and testing sets is very similar, ensuring an effective evaluation of the model's performance in this experiment.

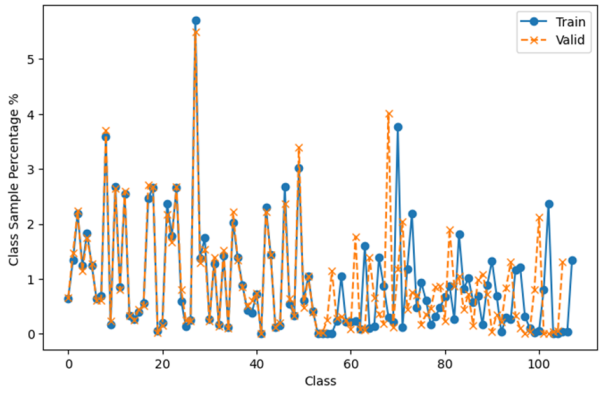


Fig. 6. Percentage distribution of different classes in the training and validation datasets.

The pipeline of the BiGRU+EA+CRF model is shown in Fig. 7. First, the input tokens are passed through the embedding layer to obtain word vectors. Subsequently, the word vectors pass through the BiGRU layer to obtain hidden vectors. Next, in the multi-head attention layer, the query is the word vectors before BiGRU, and the key and value are the hidden vectors after BiGRU. This attention process allows hidden vectors to adjust information with word vectors. Finally, vectors weighted by attention pass through the CRF layer to generate the label

sequence. Three different versions of multi-head attention mechanisms have been employed in experiments. The first version is self-attention after the embedding layer, before the BiGRU layer.

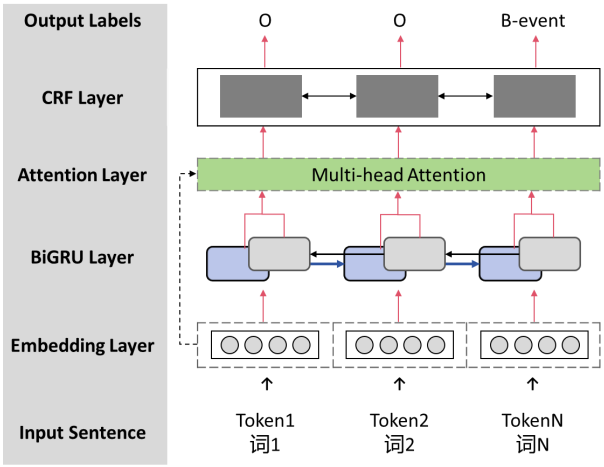


Fig. 7. Framework of BiGRU+EA+CRF model.

The second version is self-attention after BiGRU before the CRF. Placing a self-attention layer after the embedding layer and before the BiGRU layer aims to create a more informed and contextually rich representation of the input sequence early in the network. This allows the BiGRU layer to focus on modeling temporal and sequential dependencies without the additional burden of capturing long-range relationships, potentially leading to improved performance on various NLP tasks.

The third version is multi-head attention after BiGRU before the CRF, while the query is word vectors before the BiGRU layer, but key and value are the hidden vectors after the BiGRU layer. By adding a multi-head attention layer after the BiGRU layer, representation learning would be enhanced. This increases the model's ability to capture diverse and complex dependencies and improve its performance.

IV. Result Analysis

A. Evaluation

To evaluate the results obtained from the experiments, precision, recall, and F1-score based on micro and macro standards were used, which are defined as follows:

$$precision = \frac{TP}{TP + FP} \tag{2}$$

$$recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\text{-}score = \frac{2 * precision * recall}{precision + recall} \tag{4}$$

TP as True Positive is the total number of correctly predicted positive samples. FP as False Positive is the total number of incorrectly predicted positive samples. FN as False Negative is the total number of incorrectly predicted negative samples. Micro refers to micro-averaging, which calculates metrics by aggregating results across all classes. Macro refers to macro-averaging, which calculates metrics separately for each class.

TABLE II. Confusion Matrix for Binary Classification Problem

Confusion Matrix		Actual Values	
		1	0
Model Predictions	1	TP	FP
	0	FN	TN

Table II illustrates a confusion matrix, where the rows represent the model's predicted values and the columns represent the true label values of the samples.

B. Analysis

To analyze the obtained results, the performance of the nine models was compared and presented in Table III. The performance of BiLSTM and BiGRU is extremely close by both micro and macro standards. Similarly, the performance between BiLSTM+CRF and BiGRU+CRF shows similar results. This demonstrates that, despite the structural differences between BiLSTM and BiGRU, only a small performance discrepancy was observed in the experiments. By comparing the performance of the BiLSTM and BiGRU models with and without CRF, it is clear that CRF results in a slight increase in precision, but a decrease in recall, while the F1-scores remain unchanged. This indicates that precision is enhanced, but recall is compromised by CRF, without any overall performance loss. Consequently, CRF can be considered beneficial in contexts where precise identification of event types is required while maintaining overall performance.

Among the three versions of attention mechanisms integrated in the BiGRU+CRF model, minimal performance discrepancies were observed. However, the third version model, BiGRU+EA+CRF, achieves the highest precision across both macro and micro criteria. This suggests the benefit of attention mechanisms for precisely identifying event types. The third attention mechanism integrates both shallow and deep layer information, reduces interference in shallow layers, and more effectively adjusts information in deep layers compared with the other two attention mechanisms.

The results from utilizing exponential class weights in BiGRU+A+CRF+W and BiGRU+EA+CRF+W highlight an enhancement in recall by 5-7% and F1-score by 3-4%, without losing precision by macro criteria. This emphasizes the significant improvement of exponential class weights in event detection tasks with imbalanced datasets.

Based on the results of the nine models presented in Table III, the BiGRU+EA+CRF+W model yields the best overall performance. In spite of the moderate performance on the micro standard, it achieved the highest recall and F1 score and ranked second on precision on the macro standard. Results obtained in Table III were also compared with the published results of four models [18], namely DMCNN, BERT, BERT+CRF, and DMBERT (see Table IV). Based on the results of the four models published by other researchers

TABLE III. THE PERFORMANCE OF NINE MODELS ON THE TESTING DATASET

No.	Model	Micro			Macro		
		Precision	Recall	F1-score	Precision	Recall	F1-score
1	BiLSTM	81.73	86.17	83.90	77.20	76.73	76.96
2	BiGRU	81.98	**86.29**	84.08	78.47	75.95	76.13
3	BiLSTM+CRF	85.29	83.44	**84.36**	79.02	73.50	75.25
4	BiGRU+CRF	84.35	84.01	84.18	79.86	73.98	75.80
5	A+BiGRU+CRF	85.47	80.64	82.99	79.44	70.20	73.51
6	BiGRU+A+CRF	85.17	82.23	83.67	80.21	72.06	74.62
7	BiGRU+EA+CRF	**86.41**	79.66	82.90	**81.61**	70.67	74.61
8	BiGRU+A+CRF+W	82.45	84.43	83.43	80.66	76.36	77.21
9	BiGRU+EA+CRF+W	83.58	83.51	83.55	81.29	**77.07**	**78.07**

TABLE IV. THE PERFORMANCE OF OTHER FOUR MODELS [18]

No.	Model	Micro			Macro		
		Precision	Recall	F1-score	Precision	Recall	F1-score
1	DMCNN	**85.88**	79.70	82.67	80.55	73.31	75.03
2	BERT	84.19	84.31	84.25	79.61	76.76	77.33
3	BERT+CRF	82.82	84.56	84.19	79.77	77.65	77.83
4	DMBERT	83.77	**86.22**	**85.48**	**81.57**	**80.90**	**80.34**

(as presented in Table IV), the large BERT-based model DMBERT demonstrates remarkable performance. However, compared to the best RNN-based model in this study which is BiGRU+EA+CRF+W, DMBERT performs very closely by the macro precision and shows only a slight 2%-3% improvement by the micro recall and F1 score. On the other hand, BERT-based models have much larger parameters than the RNN-based models. Consequently, the BiGRU+EA+CRF+W model can train and infer faster for similar performance.

C. Discussion

Fig. 8 illustrates the capability of the proposed model, BiGRU+EA+CRF+W, by showcasing one of the output samples in which legal event types were extracted from an input Chinese text. In addition to showcasing the result of the Chinese sample, the corresponding English translation is provided to aid reader comprehension.

The Chinese text in the sample contains about 100 tokens. The blue tokens indicate legal event triggers, the red tokens indicate corresponding event types, and the green tokens indicate non-legal event triggers that are not detected by the model. The BiGRU+EA+CRF+W model successfully detects all 8 legal events. Non-legal events such as “stop”, “lubricate”, “starting”, and “roll”, although representing events, are not relevant to legal matters; thus, the model does not extract them. This example demonstrates the model’s exceptional performance in accurately identifying several legal occurrences within a lengthy Chinese text. It is worth noting that the proposed RNN-based models are currently unable to handle English event detection.



Fig. 8. Event detection result of model BiGRU+EA+CRF+W on the sample. Blue tokens indicate legal event triggers. Red tokens indicate corresponding event types. Green tokens indicate non-legal event triggers that are not detected by the model.

V. CONCLUSION

This study presents research on Chinese event detection. Most event detection research focuses on English corpora, with a limited exploration of Chinese corpora. This study aims to enhance the performance of the sequential labeling model BiLSTM for Chinese event detection. Four modifications, including replacing the base model with the BiGRU, incorporating the attention mechanism, replacing softmax with CRF, and utilising exponential class weights, were introduced. Experiments were conducted using the largest Chinese legal text event detection dataset, LEVEN. The experimental results indicate a significant improvement over the basic model. The proposed model performs nearly comparably to the state-of-the-art models based on BERT, yet with fewer parameters.

However, this study has several limitations. Firstly, experiments were conducted solely on the Chinese dataset LEVEN,

which is limited to legal events. Future research could explore other Chinese event detection datasets and even consider creating datasets due to scarcity. Secondly, the exploration of attention mechanisms was limited to multi-head attention, which exhibits modest effectiveness. Future work could explore alternative attention mechanisms and modifications of them. Lastly, further investigation into different methods for data imbalance in event detection is needed, though the exponential class weights approach effectively alleviates the issue.

ACKNOWLEDGMENT

This publication is supported by the Xiamen University Malaysia Research Fund (XMUMRF/2022-C9/IECE/0034).

REFERENCES

[1] H. Chen, K. Liu, P. Liu, J. Wang, and N. Ge, "Robustness analysis and evaluation study of chinese text event detection models," in *2023 9th International Conference on Big Data and Information Analytics (BigDIA)*, 2023, pp. 632–639.

[2] Q. Li, J. Li, J. Sheng, S. Cui, J. Wu, Y. Hei, H. Peng, S. Guo, L. Wang, A. Beheshti, and P. S. Yu, "A survey on deep learning event extraction: Approaches and applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 5, pp. 6301–6321, 2024.

[3] S. Lou, X. Xie, W. Liu, and W. Jiang, "Enhancing chinese event prediction with prompt-driven knowledge augmentation," *Applied Sciences*, vol. 15, no. 23, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/15/23/12543

[4] F. Yao, C. Xiao, X. Wang, Z. Liu, L. Hou, C. Tu, J. Li, Y. Liu, W. Shen, and M. Sun, "LEVEN: A large-scale Chinese legal event detection dataset," in *Findings of the Association for Computational Linguistics: ACL 2022*, 2022, pp. 183–201. [Online]. Available: https://aclanthology.org/2022.findings-acl.17

[5] Z. He, Z. Wang, W. Wei, S. Feng, X. Mao, and S. Jiang, "A survey on recent advances in sequence labeling from deep learning models," *arXiv preprint arXiv:2011.06727*, 2020.

[6] Y. Zhou, Z. Qu, Y. Liang, Y. Zhang, J. Zhang, and L. Ni, "Towards a word-granularity paradigm for chinese event detection: Targeting long-tail challenges in syntax and semantics," *Alexandria Engineering Journal*, vol. 134, pp. 135–151, 2026. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1110016825011718

[7] S. Pouyanfar, Y. Tao, A. Mohan, H. Tian, A. S. Kaseb, K. Gauen, R. Dailey, S. Aghajanzadeh, Y.-H. Lu, S.-C. Chen *et al.*, "Dynamic sampling in convolutional neural networks for imbalanced data classification," in *2018 IEEE conference on multimedia information processing and retrieval (MIPR)*. IEEE, 2018, pp. 112–117.

[8] S. Henning, W. Beluch, A. Fraser, and A. Friedrich, "A survey of methods for addressing class imbalance in deep-learning based natural language processing," *arXiv preprint arXiv:2210.04675*, 2022.

[9] T. Xu, K. De Barbaro, D. H. Abney, and R. F. A. Cox, "Finding structure in time: Visualizing and analyzing behavioral time series," *Frontiers in Psychology*, vol. 11, p. 1457, 2020.

[10] C. Birnie and F. Hansteen, "Bidirectional recurrent neural networks for seismic event detection," *Geophysics*, vol. 87, no. 3, p. KS97–KS111, 2022. [Online]. Available: http://dx.doi.org/10.1190/geo2020-0806.1

[11] G. Van Houdt, C. Mosquera, and G. Nápoles, "A review on the long short-term memory model," *Artificial Intelligence Review*, vol. 53, no. 8, p. 5929–5955, 2020. [Online]. Available: http://dx.doi.org/10.1007/s10462-020-09838-1

[12] I. D. Mienye and T. G. Swart, "A comprehensive review of deep learning: Architectures, recent advances, and applications," *Information*, vol. 15, no. 12, 2024. [Online]. Available: https://www.mdpi.com/2078-2489/15/12/755

[13] D. Soydaner, "Attention mechanism in neural networks: where it comes and where it goes," *Neural Computing and Applications*, vol. 34, no. 16, pp. 13 371–13 385, 2022.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[15] W. A. Shakir, "Enhancing named entity recognition through neural architectures," in *2024 21st International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*. IEEE, 2024, pp. 1–5.

[16] J. Lafferty, A. McCallum, F. Pereira *et al.*, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Icml*, vol. 1, no. 2. Williamstown, MA, 2001, p. 3.

[17] fxsjy, "Jieba: Chinese text segmentation library," 2023. [Online]. Available: https://github.com/fxsjy/jieba

[18] F. Yao, C. Xiao, X. Wang, Z. Liu, L. Hou, C. Tu, J. Li, Y. Liu, W. Shen, and M. Sun, "Leven: A large-scale chinese legal event detection dataset," *arXiv preprint arXiv:2203.08556*, 2022.

[19] S. Li, Z. Zhao, R. Hu, W. Li, T. Liu, and X. Du, "Analogical reasoning on chinese morphological and semantic relations," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, 2018, pp. 138–143. [Online]. Available: http://aclweb.org/anthology/P18-2023

[20] X. Shen, X. Tian, T. Liu, F. Xu, and D. Tao, "Continuous dropout," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 9, pp. 3926–3937, 2018.

[21] Y. Hong and J. Lin, "On convergence of adam for stochastic optimization under relaxed assumptions," in *Advances in Neural Information Processing Systems*, A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, Eds., vol. 37. Curran Associates, Inc., 2024, pp. 10 827–10 877. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2024/file/14bb27f680bee45d83bc769738e7f9b5-Paper-Conference.pdf

SIFChain: A Decentralized Framework for Secure Storage Sharing and Dynamic Access Control in Virtual Power Plants

Xiaochuan Xu[1]*, Xiao Xin[2], Jie Liu[3], Ruiqi Fang[4], Dekai Liu[5], Zhixin Li[6]
School of Cyber Science and Engineering, Southeast University, Nanjing, 211118, China[1, 4]
Shandong Future Group Co., Ltd, Jinan, 250100, China[2, 3, 6]
School of Information Science and Engineering, University of Jinan, Jinan, 250022, China[5]

***Abstract*—Virtual Power Plants (VPPs) face significant challenges in secure data management and sharing, including risks of centralized control, single points of failure, and dynamic access requirements among multiple stakeholders. To address these issues, this study proposes SIFChain, a decentralized framework that integrates Hyperledger Fabric, the InterPlanetary File System (IPFS), and a revocable Ciphertext-Policy Attribute-Based Encryption (CP-ABE) scheme with collaborative key generation. Unlike existing solutions such as Filecoin or Storj, SIFChain introduces a dual-channel blockchain architecture that separates public operational data from sensitive attribute information, mitigating privacy leakage and access policy exposure. The framework achieves fine-grained, dynamic access control with forward and backward security through an enhanced CP-ABE mechanism. Experimental evaluation demonstrates that SIFChain provides scalable performance: data upload/download times increase linearly from 1 MB to 1 GB, blockchain transaction latency remains under 5 ms for typical operations (registration, access requests, policy updates), and attribute-based encryption/decryption overhead scales linearly with policy complexity. These results confirm the practicality of SIFChain for secure, cross-organizational data sharing in Virtual Power Plant ecosystems.**

***Keywords*—Virtual Power Plant; access control; CP-ABE; IPFS; blockchain; distributed energy; privacy protection**

I. Introduction

With the rapid advancement of renewable energy integration and smart grid technologies, Virtual Power Plants (VPPs) have emerged as critical components in modern energy systems [1]. VPPs aggregate distributed energy resources (DERs) such as solar panels, wind turbines, battery storage systems, and flexible loads into unified, controllable entities that participate in electricity markets and grid services [2]. These systems generate massive volumes of operational data daily, including generation forecasts, real-time power measurements, market bids, grid interactions, and participant settlement information [21].

According to energy industry analysis [3], well-managed VPPs can improve grid stability by 20-30% while reducing operational costs by 15-25%, significantly enhancing energy efficiency and renewable integration [22]. However, current VPP data management systems face multiple security and governance challenges. From a technical perspective, traditional VPP architectures often rely on centralized control systems, creating single points of failure and exposing critical energy infrastructure to risks of cyber-attacks, data tampering, and unauthorized access [4], [15]. The threat model includes malicious insiders, external adversaries targeting communication links, and compromised DER devices that could leak sensitive operational data. These vulnerabilities make it difficult to ensure data integrity and reliability in multi-stakeholder energy environments involving generators, aggregators, grid operators, regulators, and consumers.

Furthermore, from an access control standpoint, VPP data management is typically controlled by single entities or limited consortia, leaving other legitimate stakeholders without verifiable and open access to necessary operational and financial data. As energy markets become increasingly decentralized and dynamic, with frequent changes in participation, ownership structures, and regulatory requirements, the need for secure, fine-grained, and adaptable access control mechanisms has become paramount. The evolving nature of VPP collaborations—where participants may join or leave, assets may change ownership, and access privileges must adapt to real-time market conditions—requires fundamentally new approaches to data security and sharing.

Blockchain technology offers promising solutions to these challenges through its distributed ledger architecture, which authenticates and records data via consensus algorithms without requiring trusted third parties [5]. The immutable nature of blockchain records provides strong guarantees against data tampering, while smart contracts enable automated execution of energy agreements, settlements, and access policies. Hyperledger Fabric, as a permissioned blockchain framework [6], is particularly suitable for VPP applications due to its efficient consensus mechanisms, multi-channel architecture for data isolation, and support for complex organizational structures typical in energy ecosystems.

The InterPlanetary File System (IPFS) complements blockchain technology by providing decentralized storage capabilities, addressing the inherent storage limitations of blockchain systems for large-scale energy data [7]. By storing encrypted VPP operational data on IPFS and maintaining only cryptographic references on the blockchain, energy organizations can achieve both scalability and security while preserving data availability. Additionally, Attribute-Based Encryption (ABE) technology [8], particularly Ciphertext-Policy Attribute-

*Corresponding author.

Based Encryption (CP-ABE) [9], enables fine-grained access control by allowing data owners to define access policies based on user roles, organizational affiliations, and operational contexts rather than specific identities.

Despite these advancements, existing decentralized storage systems such as Filecoin and Storj rely on proof-of-storage mechanisms that introduce high computational overhead and lack fine-grained access control tailored to multi-stakeholder VPP environments. Moreover, typical IPFS-Ethereum integrations suffer from on-chain storage bottlenecks and lack privacy-preserving attribute management. Many systems still rely on centralized key management authorities or lack efficient mechanisms for access privilege revocation and updates as market conditions change [10], [11], [16], [17]. These limitations create security vulnerabilities and administrative overhead in practical VPP deployments where access requirements frequently evolve due to changing market participants [12], regulatory compliance needs, asset ownership transfers, and real-time operational conditions.

This study addresses these gaps by proposing SIFChain, a comprehensive framework for secure storage sharing and dynamic access control in Virtual Power Plants. Our solution integrates Hyperledger Fabric, IPFS, and an enhanced revocable CP-ABE scheme with collaborative key generation, which mitigates the risk of single points of failure in key distribution. The main contributions of this work are as follows:

- We propose SIFChain, an efficient dynamic storage sharing framework for VPP consortium blockchains that enables granular and flexible access control for sensitive energy data through an enhanced Ciphertext attribute-based encryption algorithm.
- We design a multi-channel architecture comprising public channels and attribute channels to address access policy and attribute leakage challenges in on-chain attribute encryption. This architecture enables separate storage of different data types (operational, financial, regulatory), ensuring enhanced privacy management through dedicated attribute nodes representing different VPP stakeholders.
- We modify the Certificate Authority (CA) structure to enable collaborative attribute key generation between the CA and attribute authorization nodes, enhancing the security of user attribute private key generation and distribution in distributed energy environments.
- We implement a comprehensive security strategy that combines symmetric encryption for IPFS-stored VPP data with attribute-based encryption for blockchain-managed access control, ensuring data security while facilitating efficient storage and sharing across organizational boundaries in energy ecosystems.

The remainder of this study is organized as follows: Section II reviews related woks. Section III details the specific implementation of the SIFChain framework in VPP contexts. Section IV presents experimental results and performance analysis. Finally, Section V concludes the study and outlines directions for future research.

II. Related Work

Decentralized storage and access control for energy systems have attracted significant research interest. Filecoin and Storj provide blockchain-based storage markets but focus on proof-of-storage rather than fine-grained access policies, making them less suitable for dynamic multi-stakeholder VPPs. IPFS combined with Ethereum has been proposed for IoT data sharing [13], yet on-chain storage costs and lack of attribute-based privacy controls limit scalability. In the context of VPPs, consortium blockchains like Hyperledger Fabric have been explored for energy trading [11] and data management [14], but these works often assume static access rights and do not address attribute revocation or collaborative key generation.

CP-ABE schemes with revocation capabilities have been studied extensively [8], [9]. However, most implementations rely on a single authority for key distribution, creating a central point of trust. Collaborative key generation between multiple authorities, as in SIFChain, reduces this vulnerability. Furthermore, existing VPP data management frameworks seldom integrate IPFS for scalable storage alongside dynamic attribute-based access control [18]. Our work bridges this gap by proposing a holistic architecture that combines Fabric's permissioned channels, IPFS's content-addressed storage, and a revocable CP-ABE scheme with distributed key generation, thereby advancing the state-of-the-art in secure VPP data sharing.

III. SIFChain Framework for Virtual Power Plants

A. System Architecture and Stakeholders

Our solution enhances the storage and sharing of VPP data among multiple stakeholders, including distributed energy resource owners, VPP aggregators, grid operators, electricity market participants, regulators, and consumers. The system consists of four main entities: the Fabric consortium blockchain, the IPFS distributed file system, data owners (typically VPP operators or asset owners), and data consumers (various authorized stakeholders).

We enhance dynamic access control through the revocable Ciphertext-Policy Attribute-Based Encryption (CP-ABE) mechanism, where decryption key parameters are generated collaboratively by the Fabric Certificate Authority (Fabric-CA) and the attribute authorities AA_l representing different stakeholder groups. This approach reduces the risk of data leakage and enhances security in the distributed energy environment.

For the consortium blockchain, we employed a dual-channel architecture that distinctly separates public operational data (e.g., aggregated power forecasts, market prices) from sensitive attribute data (e.g., individual asset performance, participant identities, financial settlements), encryption policies, and other privacy-related information, thereby ensuring enhanced data security. This solution also utilizes the IPFS distributed file system for secure and efficient storage of VPP operational data, including high-frequency sensor data, detailed asset performance logs, and historical market interactions. Data owners encrypt their sensitive VPP data and upload it to IPFS, creating a dedicated, secure IPFS network that guarantees data confidentiality while improving storage efficiency for large-scale energy datasets.

B. Data Flow and Access Control Mechanism

In terms of data storage, IPFS only stores the ciphertext of the original VPP data, while the blockchain maintains the address in IPFS, along with symmetric keys and hash values of the original data. When an endorsing node in Hyperledger Fabric executes a smart contract for data access, it combines the data user attribute set with the channel anchor point to verify the ciphertext decryption. If the attribute set meets the established access policy (e.g., "Grid Operator AND Real-time Operations"), the consumer can successfully decrypt the ciphertext and gain access to the data; otherwise, the endorsing node will return a failure indicator, thereby denying the access request. This design facilitates secure data storage and fine-grained attribute-based access control tailored to VPP requirements [13].

C. Relevant Symbols

To facilitate the discussion, we first present the required parameters and relevant symbols in Table I.

TABLE I. NOTATION FOR SIFCHAIN IN VPP CONTEXT

Notation	Description
PP	System common parameter
ID_i	ID of user i in the VPP system
AA_l	Attribute authorization node l representing a stakeholder group
S	Collection of attributes (roles, organizations, access levels)
PK	Attribute encryption public key in the system
MSK	Attribute encryption master key in the system
ω	Attribute set of user ID_i in the system
$SK_{ID_i,\omega}$	Attribute private key of user ID_i
m	Original sensitive VPP data (sensor readings, bids, etc.)
H_1	Hash value of the original sensitive data
CT_m	Symmetrically encrypted ciphertext of the original data
URL	Address where IPFS stores encrypted data
k	Symmetric key used to encrypt data
m_1	New data formed by combining URL, k, and H_1
R_l	Revocation lists maintained in the system
P^*_ω	Access policy for attribute encryption
CT_{ABE}	Ciphertext encrypted with data m_1 attribute
$CT^{'}_{ABE}$	Recalculated attribute encrypted ciphertext
$leaf_j$	Access the leaf node j in the policy
$Root$	Access the root node in the policy
x	Accessing non-leaf nodes in a policy
z	Accessing child nodes of non-leaf nodes
H_2	Hash value computed by the data user for verification
DO	Data owner in the system (VPP operator/asset owner)
DU	Data users in the system (various stakeholders)
CA	Certificate Management Center in Fabric

D. System Model for VPP Applications

The proposed storage sharing system for Virtual Power Plants is illustrated in Fig. 1. The Fabric consortium blockchain features two core channels: a public channel for operational data and an attribute channel for sensitive information [14], each with its own distinct ledger.

We classify nodes into two categories: public nodes (maintaining the public ledger with grid and market data), and attribute nodes (joining the attribute channel to manage attribute data for different stakeholder groups).

Before VPP data is uploaded to the blockchain, the data owner predefines the access policy based on energy market rules and regulatory requirements. The sensitive VPP data is encrypted using a symmetric algorithm and subsequently stored in IPFS. The IPFS address, symmetric encryption key, and data hash are then uploaded to the blockchain using attribute encryption. In the blockchain, the ciphertext and revocation lists are maintained as public data within the public channel, while sensitive information—such as encryption policies, participant attributes, and financial details—is stored in the attribute channel, managed by attribute nodes representing different stakeholder interests. Data within the same channel is synchronized in real-time across all relevant nodes.

When a data user (e.g., grid operator, regulator, market participant) requests access to VPP data, the attribute node evaluates the user's attribute set within the attribute channel utilizing Ciphertext-Policy Attribute-Based Encryption (CP-ABE) to verify whether the consumer satisfies the access policy. If verification succeeds, the system performs decryption and grants access; otherwise, a failure message is returned. Access rights in VPPs may change frequently due to market conditions, regulatory updates, or participant status changes. If a consumer loses privileges, the system uses the revocation list to re-encrypt the ciphertext with a random value, replacing the original to ensure both forward and backward secrecy [19].

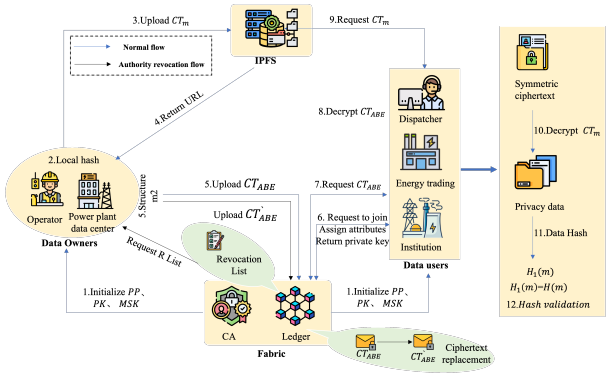


Fig. 1. SIFChain system model for Virtual Power Plants.

E. Components of the Solution

This solution consists of six stages: system initialization, key generation, data encryption and storage, data encryption upload, attribute revocation, and decryption. The specific operations are detailed as follows:

1) System initialization: The Fabric-CA selects a bilinear mapping $e : G_1 \times G_1 \to G_2$, where G_1, G_2 are two cyclic groups of order q. Then, it selects generators $g, h, v \in G_1$ and random numbers $\alpha, \beta \in Z_q$, and computes $Z = e(g, g)^{\beta}, v^{\beta}$. Finally, it outputs the system public parameters $PP = (G_1, G_2, e, g, h, v, Z)$.

For each authenticated VPP participant ID_i in the system, the CA selects a unique secret parameter $\lambda_i \in Z_q$ and securely sends (λ_i, β) to the user, while sending (α, h) to the attribute authorization node AA_l in Fabric, where $l \in \{1, \ldots, n\}$ is the index of each attribute authorization node, and n is the total number of attribute authorization nodes in the system.

Define the attribute domain S representing VPP roles and permissions (e.g., "Grid Operator", "Market Participant", "Asset Owner", "Regulator"). For any attribute $j \in S$, if the attribute authorization node AA_l holds the key distribution authority for it, i.e., $j \in AA_l$, then AA_l selects a secret random

number $t_j \in Z_q^*$, computes $T_j = g^{t_j}$, and finally outputs the attribute public key $PK = (\{T_j\}_{j \in AA_l})_{l \in (1,2,\ldots,n)}$ and the attribute master secret key $MSK = (\{t_j\}_{j \in AA_l})_{l \in (1,2,\ldots,n)}$.

2) Key generation: A VPP participant ID_i with attribute set ω (e.g., $\omega = \{\text{Asset Owner}, \text{Solar Farm}, \text{North Region}\}$) sends λ_i to the attribute authority node AA_l to request the corresponding private key. For any attribute $j \in \omega \cap AA_l$, the computation is conducted as follows:

$$S_{i,j} = g^{\frac{\alpha \lambda_i}{t_j}} h^{\frac{\alpha}{t_j}}, \quad W_i = g^{\alpha \lambda_i} \tag{1}$$

After receiving $S_{i,j}$ and W_i, the user computes $W_i^* = W_i \cdot g^{-\beta}$. The final private key for VPP participant ID_i is:

$$SK_{ID_i,\omega} = (\lambda_i, W_i^*, S_{i,1}, \ldots, S_{i,\|\omega\|}) \tag{2}$$

In this process, since each attribute authority node AA_l cannot obtain the secret parameter β, and the Certificate Authority (CA) cannot access the attribute key t_j, it is ensured that untrusted attribute authority nodes and the system center cannot decrypt the user's ciphertext.

3) Data encryption and storage: The VPP data owner employs the Secure Hash Algorithm 256 (SHA-256) hash function to hash the original data m (e.g., power generation data, market bids), denoted as $H(m) \rightarrow H_1$, where H_1 is utilized for subsequent verification. The data owner symmetrically encrypts the data as $Enc(k, m) \rightarrow CT_m$, where k is the encryption key, and CT_m is the resulting ciphertext. CT_m is uploaded to IPFS for storage.

Upon receiving CT_m, IPFS evaluates the size of the data. If the size exceeds 256 KB, it performs chunking on the data and stores the chunks across nodes located around the world. This process generates a unique hash value based on the content, referred to as the storage address, represented as a URL. Finally, the URL is returned to the data owner.

The data owner executes a concatenation operation, specifically $m_1 = (URL \| k \| H_1)$, where m_1 represents the new data, URL is the unique address returned by IPFS, k is the symmetric encryption key, and H_1 is the hash value.

4) Data encryption and upload: Obtain the latest attribute revocation list from all attribute authority nodes AA_l, represented as $R_l = \{(j, (\lambda_{rev}, L_{j,rev})_{rev \in List_j})\}$. The data owner selects random numbers $s, r \in Z_q^*$ and calculates:

$$C_0 = m_1 \cdot Z^{sr}, \quad C^* = g^{sr} \tag{3}$$

The data owner selects an access tree policy P_{ω^*} that defines which attributes are required to access the VPP data. Starting from the root node, an $n_x - 1$ degree polynomial q_x is selected for each internal node from top to bottom, as detailed in the access policy generation algorithm shown in Fig. 2.

For each leaf node $leaf_j$ in the access tree P_{ω^*}, compute:

$$C_{1,j} = T_j^{r \cdot q_{leaf_j}(0)}, \quad C_{2,j} = L_j^{r \cdot q_{leaf_j}(0)} \tag{4}$$

where, $L_j = v \cdot g^{t_j}$.

For attributes that have been revoked (i.e., $j \in \omega^*, List_j \neq \emptyset$), compute:

$$C_{1,j,rev} = T_j^{r \cdot (q_{leaf_j}(0) - u_{rev})}, \quad C_{2,j,rev} = L_j^{r \cdot (q_{leaf_j}(0) - u_{rev})} \tag{5}$$

where. u_{rev} is a random number associated with the revoked user.

The final attribute-based encrypted ciphertext is:

$$CT_{ABE} = \langle C_0, C^*, \{C_{1,j}, C_{2,j}\}_{j \in \omega^*, \text{List}_j = \emptyset}, \{C_{1,j,\text{rev}}, C_{2,j,\text{rev}}\}_{j \in \omega^*, \text{rev} \in \text{List}_j} \rangle \tag{6}$$

Once the data owner encrypts m_1, they upload CT_{ABE}, along with the encryption policy P_{ω^*}, to the blockchain. The attribute-encrypted ciphertext is stored in the public channel, while the access policy and user attributes are securely stored in the attribute channel. The public channel also holds the revocation list for managing access permissions. The comprehensive architecture for this data encryption and upload process is illustrated in Fig. 3.

Algorithm 1: Attribute-based encryption based on decision trees

1. Root :$q_{root}(0) = m_1$,Other n_{root}-1 coefficients are randomly selected.
2. Internal Node x:$q_x(0) = q_{parent}(index(x))$.
3. Leaf Node $leaf_i$: $q_{leaf_i}(0) = q_x(index(leaf_i))$.
4. **for** $j \in \omega^*, List_j = \emptyset$ **do**
   Compute:
   $C_{1,j} = T_j^{q_{leaf_j}(0) \cdot r}$, $C_{2,j} = L_j^{q_{leaf_j}(0) \cdot r}$.
5. **for** $j \in \omega^*, List_j \neq \emptyset$ **do**
   Traverse j revocation list $List_j$, select $|List_j|$ random numbers $u_1 + u_2 + \ldots + u_{|List_j|} = q_{leaf_j}(0)$.
   Compute:
   $\{C_{1,j,rev} = T_j^{u_{rev} \cdot r}, C_{2,j,rev} = L_{j,rev}^{u_{rev} \cdot r}\}_{j \in \omega^*, rev \in List_j}$
6. **output**
   $CT_{ABE} = < C_0, C^*, (C_{1,j}, C_{2,j})_{j \in \omega^*, List_j = \emptyset}, (C_{1,j,rev}, C_{2,j,re})_{j \in \omega^*, rev \in List_j} >$.

Fig. 2. Access policy generation algorithm for VPP data.

5) Attribute revocation: When a VPP participant loses decryption privileges or access rights change (e.g., a market participant leaves the VPP consortium), the data owner can re-encrypt the ciphertext without updating the decryption keys. The system supports two revocation scenarios:

a) Case 1: Attribute not previously revoked: ($j \in \omega^*, List_j = \emptyset$) Select a random number u_{rev}^* and compute the re-encrypted ciphertext $CT_{ABE}^{'}$:

$$C_{1,j,rev} = C_{1,j} \cdot T_j^{-u_{rev}^* \cdot r} \tag{7}$$

$$C_{2,j,rev} = C_{2,j} \cdot L_j^{-u_{rev}^* \cdot r} = L_j^{(q_{leaf_j}(0) - u_{rev}^*) \cdot r} \tag{8}$$

$$C_{1,j,rev^*} = T_j^{u_{rev}^* \cdot r}, \quad C_{2,j,rev^*} = L_j^{u_{rev}^* \cdot r} \tag{9}$$

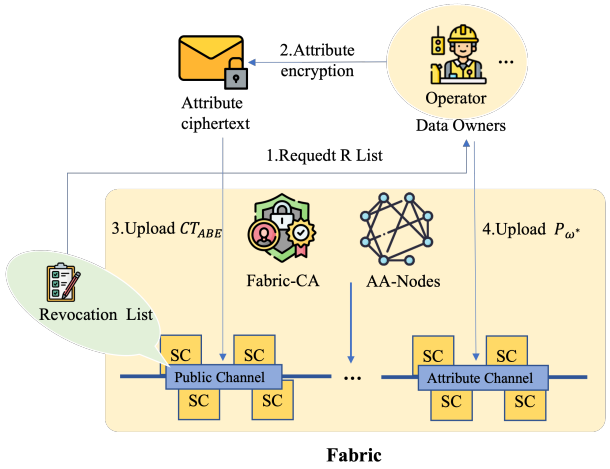


Fig. 3. Data encryption and upload architecture for VPP.

b) Case 2: Attribute already revoked: ($j \in \omega^*, List_j \neq \emptyset$) Select a random number u^*_{rev} and compute the re-encrypted ciphertext $CT^{'}_{ABE}$:

$$C_{1,j,rev} = C_{1,j,rev} \cdot T_j^{-u^*_{rev} \cdot r} = T_j^{(u_{rev} - u^*_{rev}) \cdot r} \tag{10}$$

$$C_{2,j,rev} = C_{2,j,rev} \cdot L_j^{-u^*_{rev} \cdot r} = L_j^{(u_{rev} - u^*_{rev}) \cdot r} \tag{11}$$

$$C_{1,j,rev^*} = T_j^{u^*_{rev} \cdot r}, \quad C_{2,j,rev^*} = L_j^{u^*_{rev} \cdot r} \tag{12}$$

The original attribute-encrypted ciphertext should be replaced with the newly computed ciphertext $CT^{'}_{ABE}$ on the blockchain.

6) Data decryption: The decryption algorithm operates recursively on the access tree P_{ω^*}. For each leaf node $leaf_j$, query the attribute revocation list R.

If the user's attribute has not been revoked ($j \in \omega \cap \omega^*, List_j = \emptyset$), compute:

$$\begin{aligned} \text{DecryptNode}\big(CT_{ABE}, SK_{ID,\omega}, \text{leaf}_j\big) &= \frac{e\big(S_{i,j} \cdot v^{\beta}, C_{1,j}\big)}{e\big(g, C_{2,j}\big)} \\ &= e(g,g)^{\alpha r \lambda_i \cdot q_{\text{leaf}_j}(0)} \end{aligned} \tag{13}$$

If the user's attribute has been revoked ($j \in \omega \cap \omega^*, List_j \neq \emptyset$), compute:

$$\begin{aligned} &\text{DecryptNode}(CT_{ABE}, SK_{ID,\omega}, \text{leaf}_j) \\ &= \prod_{rev \in \text{List}_j} \left(\frac{e(S_{i,j} \cdot v^{\beta}, C_{1,j,rev})}{e(g, C_{2,j,rev})} \right)^{\frac{\lambda_i}{\lambda_i - \lambda_{rev}}} \\ &= e(g,g)^{\alpha r \lambda_i \cdot q_{\text{leaf}_j}(0)} \end{aligned} \tag{14}$$

For a non-leaf node x in the access tree P_{ω^*} where the child node is z, call the result of $DecryptNode(CT_{ABE}, SK_{ID,\omega}, z)$ and denote it as F_Z.

$$\begin{aligned} F(x) &= \prod_{z \in S_x} F(z)^{\Delta_{i, S^{(0)}_x}} \\ &= \prod_{z \in S_x} \left(e(g,g)^{\alpha r \lambda_i \cdot q_z(0)}\right)^{\Delta_{i, S^{(0)}_x}} \\ &= e(g,g)^{\alpha r \lambda_i \cdot q_x(0)} \end{aligned} \tag{15}$$

The user is able to recursively compute $F(root) = e(g,g)^{\alpha r \lambda_i s}$, using the Lagrange interpolation method if and only if their attribute set ω satisfies the access policy P_{ω^*}.

The plaintext m_1 is ultimately recovered through the subsequent calculation:

$$\begin{aligned} m_1 &= \frac{C_0 \cdot e(C^*, W_i^*)}{F(root)} \\ &= \frac{m_1 \cdot Z^{sr} \cdot e(g^{sr}, g^{\alpha \lambda_i - \beta})}{e(g,g)^{\alpha \lambda_i r s}} \\ &= \frac{m_1 \cdot e(g,g)^{\beta s r} \cdot e(g^{sr}, g^{\alpha \lambda_i - \beta})}{e(g,g)^{\alpha \lambda_i r s}} \end{aligned} \tag{16}$$

The data user decomposes the retrieved data $m_1 = (URL \| k \| H_1)$ into three distinct components: the unique address URL, the symmetric key k, and the hash value H_1. The user will locally store k and H_1, and use the URL to query IPFS and download the encrypted data CT_m.

The data user performs the decryption algorithm $Decrypt(CT_m, k) \rightarrow m$, where CT_m is the ciphertext downloaded from IPFS, k is the symmetric key provided by the data owner, and m is the original VPP data that the user intends to access. After obtaining m, the user performs a hash operation on it, resulting in H_2. By comparing H_2 with H_1, if the two are found to be consistent, the data can be confirmed as authentic and valid. The data can then be utilized for various VPP purposes, such as a grid operator making dispatch decisions, a market participant submitting bids, or a regulator auditing VPP operations.

IV. Implementation and Performance Evaluation

We conducted the experiments on a server equipped with an 11th-generation 64-bit Intel i7-11800H processor (2.30 GHz, 8 GB RAM). The Hyperledger Fabric blockchain network was implemented on Ubuntu 22.04 using Fabric version 1.4.12 and Fabric CA version 1.4.12. All nodes operated on virtual machines via VMware Workstation. We selected a Java-based pairing cryptography library version 2.0.0 with an A-type curve for bilinear pairing operations relevant to VPP security requirements.

A. Data Upload and Download Performance

We evaluated data sizes typical for VPP applications: 1 MB (configuration files), 5 MB (hourly operational reports), 10 MB (daily aggregates), 50 MB (market bidding data), 100 MB (high-frequency sensor logs), 500 MB (historical performance data), and 1 GB (comprehensive datasets). The results, shown in Fig. 4, demonstrate linear scaling suitable for VPP data volumes.

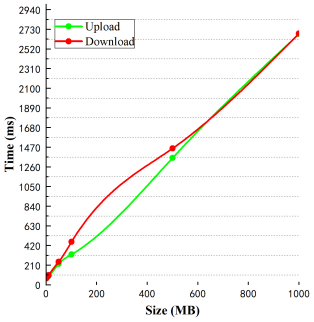
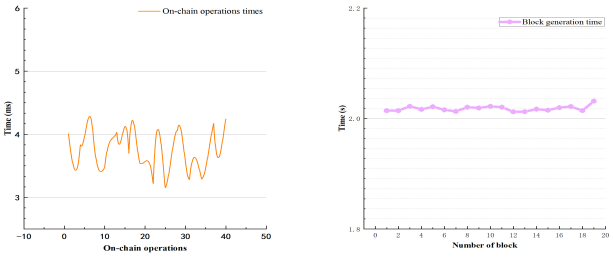


Fig. 4. Data upload and download performance for VPP data on IPFS.

B. Blockchain Performance for VPP Operations

We measured blockchain operations relevant to VPP scenarios: participant registration (2.8 ms), data access requests (3.1 ms), policy updates (3.5 ms), and revocation operations (4.0 ms). As depicted in Fig. 5, the block generation time remained stable at 2.02 seconds for VPP transaction batches, highlighting the system's operational efficiency. Each block contains VPP-specific metadata: operation type, stakeholder identifiers, timestamp, previous/current hashes, and data hashes.



(a) VPP on-chain operation times. (b) VPP block generation times.

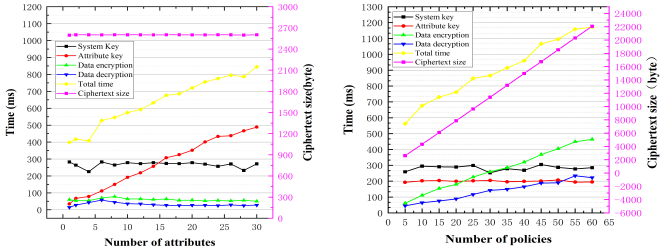
Fig. 5. Blockchain performance for VPP operations.

C. Access Control Performance Analysis

We evaluated system performance under VPP-specific attribute configurations: stakeholder roles (owner, operator, regulator, consumer), access levels (real-time, historical, financial), and organizational affiliations. With 5 policies and 10 attributes fixed, we measured initialization time (stabilizing at 255 ms), encryption time (proportional to policy complexity), decryption time (meeting real-time VPP requirements), and ciphertext size (scaling linearly with policies). The performance impacts of increasing attributes and policies are presented in Fig. 6. These results confirm the system's suitability for dynamic VPP environments where policies frequently evolve.

D. Discussion on Practical Deployment

While the experimental results demonstrate the feasibility of SIFChain, several practical considerations warrant discussion beyond the performance analysis presented above.



(a) VPP attribute scaling performance. (b) VPP policy scaling performance.

Fig. 6. SIFChain performance under VPP parameter variations.

First, the experiments were conducted on a server-grade processor, whereas real VPP deployments involve numerous low-power IoT devices at the edge. The computational overhead of CP-ABE operations, particularly pairing and exponentiation, may be significant on such constrained hardware. However, in practice, encryption is performed by data owners (typically VPP operators with sufficient computational resources), while decryption is carried out by consumers such as grid operators and regulators, who also possess adequate capacity. For resource-constrained edge nodes, lightweight ABE variants or offloading cryptographic operations to gateway devices could be considered as mitigations.

Second, our implementation uses Hyperledger Fabric v1.4.12, which, although stable, has been superseded by newer versions (2.x and 3.x) with different performance characteristics. The framework design is version-agnostic, and we expect similar trends in newer releases, but actual latency and throughput may vary. Future work should evaluate SIFChain on updated Fabric versions to validate its performance in modern production environments.

Third, the revocation mechanism requires communication between the data owner and attribute authorities to update the ciphertext. In a wide-area network, this may introduce additional latency proportional to the number of revoked attributes. Our experiments measured revocation operation time at 4.0ms in a local network; however, in geographically distributed settings, network delays could increase this to tens or hundreds of milliseconds. Nevertheless, revocation is an infrequent operation in VPPs, and the impact on overall performance is acceptable given the security benefits.

Finally, the evaluation was conducted in a controlled environment without a geographically distributed VPP consortium. Realistic deployment would involve multiple sites with heterogeneous network conditions and IoT devices, which could affect both performance and security. We plan to conduct field trials in a pilot VPP testbed as future work to validate the framework under real-world conditions.

E. Security Analysis

The security of SIFChain relies on standard cryptographic assumptions, including the bilinear Diffie-Hellman (BDH) assumption for the CP-ABE scheme and the collision resistance of SHA-256. We provide an informal security analysis covering the following guarantees:

1) Data confidentiality: Through the combination of symmetric encryption for IPFS-stored data and CP-ABE for access control, sensitive VPP data remains confidential even if IPFS nodes or blockchain channels are compromised.

2) Forward and backward security: The revocable CP-ABE scheme ensures that when a VPP participant's access privileges are revoked, they cannot decrypt newly encrypted data (forward security) nor previously encrypted data that they could previously access (backward security). Even if a user caches old symmetric keys, the re-encryption of the ciphertext with fresh randomness invalidates those keys, preserving backward secrecy.

3) Collusion resistance: Even if multiple VPP participants collude by combining their attribute keys, they cannot decrypt ciphertext unless at least one of them individually satisfies the access policy.

4) Decentralized trust: The collaborative key generation between CA and attribute authorization nodes eliminates single points of trust, making the system resilient to compromise of any single authority.

5) Data integrity: The use of cryptographic hashes ensures that any tampering with VPP data stored on IPFS can be detected during the decryption and verification process.

6) Privacy leakage: Although the dual-channel architecture isolates sensitive attributes, blockchain transactions themselves may leak metadata such as access patterns. An adversary monitoring the network could infer relationships between data owners and consumers. To mitigate this, future iterations could incorporate obfuscation techniques or zero-knowledge proofs [20]; however, this remains an open challenge.

We emphasize that a full formal proof of the CP-ABE scheme under bilinear assumptions is beyond the scope of this study and is left as future work. Nevertheless, the design follows established practices from prior art [8], [9].

V. Conclusion

This study presents SIFChain, a decentralized data management framework for Virtual Power Plants that integrates Fabric blockchain, IPFS, and enhanced revocable CP-ABE to ensure secure storage and dynamic access control. The key innovation is the collaborative key generation mechanism between CA and attribute nodes, reducing centralization vulnerabilities in distributed energy systems. The dual-channel architecture preserves confidentiality while supporting multi-stakeholder VPP operations. The revocable CP-ABE scheme enables fine-grained access control adapting to dynamic VPP environments. By storing encrypted data on IPFS, we achieve scalable security without semi-trusted third parties. Experimental results demonstrate linear scalability and low-latency operations suitable for VPP workloads. We also discussed practical considerations for IoT hardware, newer Fabric versions, and network overhead, and identified privacy leakage as an area for future enhancement. Future work will focus on decentralized attribute encryption, integration with real VPP systems, enhanced key management for large-scale energy applications, and formal security proofs.

References

[1] P. D. Lund, J. Lindgren, J. Mikkola, and J. Salpakari, "Review of energy system flexibility measures to enable high levels of variable renewable electricity," Renewable and Sustainable Energy Reviews, vol. 45, pp. 785–807, 2015.

[2] E. Mengelkamp, J. Gärtner, K. Rock, et al., "Designing microgrid energy markets: A case study: The Brooklyn Microgrid," Applied Energy, vol. 210, pp. 870–880, 2018.

[3] M. Andoni, V. Robu, D. Flynn, et al., "Blockchain technology in the energy sector: A systematic review of challenges and opportunities," Renewable and Sustainable Energy Reviews, vol.100,pp.143–174,2019.

[4] C. Zhang, J. Wu, C. Long, and M. Cheng, "Review of existing peer-to-peer energy trading projects," Energy Procedia, vol. 105, pp. 2563–2568, 2017.

[5] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[6] E. Androulaki, A. Barger, V. Bortnikov, et al., "Hyperledger fabric: A distributed operating system for permissioned blockchains," in Proc. 13th EuroSys Conf., 2018, pp. 1–15.

[7] J. Benet, "IPFS - content addressed, versioned, P2P file system," arXiv preprint arXiv:1407.3561, 2014.

[8] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute-based encryption," in 2007 IEEE Symp. on Security and Privacy (SP'07), 2007, pp. 321–334.

[9] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proc. 13th ACM Conf. on Comput. and Commun. Security, 2006, pp. 89–98.

[10] S. Wang, A. F. Taha, J. Wang, et al., "Energy crowdsourcing and peer-to-peer energy trading in blockchain-enabled smart grids," IEEE Trans. Syst., Man, Cybern., Syst., vol. 49, no. 8, pp. 1612–1623, 2019.

[11] Z. Li, J. Kang, R. Yu, et al., "Consortium blockchain for secure energy trading in industrial internet of things," IEEE Trans. Ind. Informat., vol. 14, no. 8, pp. 3690–3700, 2018.

[12] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home," in 2017 IEEE Int. Conf. on Pervasive Comput. and Commun. Workshops (PerCom Workshops), 2017, pp. 618–623.

[13] Y. Zhang and J. Wen, "The IoT electric business model: Using blockchain technology for the internet of things," Peer-to-Peer Netw. and Appl., vol. 10, no. 4, pp. 983–994, 2017.

[14] X. Luo, B. Zhang, and Z. Zhang, "A secure and efficient data sharing scheme based on blockchain in vehicular social networks," IEEE Trans. Veh. Technol., vol. 69, no. 6, pp. 5826–5835, 2020.

[15] M. Al-Shatri, A. Al-Fuqaha, and H. S. Al-Raweshidy, "Virtual power plants: Challenges, opportunities, and profitability assessment in current energy markets," Electricity, vol. 5, no. 2, pp. 370–384, 2024.

[16] A. Bedi, J. Ramprabhakar, R. Anand, V. P. Meena, and I. A. Hameed, "Empowering net zero energy grids: A comprehensive review of virtual power plants, challenges, applications, and blockchain integration," Discover Applied Sciences, vol. 7, Art. no. 252, 2025.

[17] C. Feng, J. Cheng, and S. Chen, "Blockchain-based transaction mechanism in virtual power plant: Considering users' privacy and reputation," Electric Power Systems Research, vol. 249, Art. no. 111988, 2025.

[18] X. Wang, L. Zhang, and J. Wu, "Hyperledger Fabric-based multi-channel structure for data exchange in internet of vehicles," Electronics, vol. 14, no. 3, Art. no. 572, 2025.

[19] J. Xu, W. Susilo, and J. K. Liu, "An efficient traceable and revocable access control scheme for smart grids," Symmetry, vol. 17, no. 2, Art. no. 294, 2025.

[20] Q. Li, H. Wang, and B. Cao, "Advancing user privacy in virtual power plants: A novel zero-knowledge proof-based distributed attribute encryption approach," Electronics, vol. 13, no. 7, Art. no. 1283, 2024.

[21] W. Luan, L. Tian, B. Zhao, and Q. Ai, "A multi-timescale blockchain-based virtual power plant trading framework for building integrated photovoltaic prosumers," Applied Energy, vol. 398, Art. no. 126422, 2025.

[22] M. Song, X. Xu, C. Gao, et al., "Two-stage stochastic scheduling of virtual power plant based on transactive control," CSEE Journal of Power and Energy Systems, vol. 11, no. 3, pp. 1442–1453, 2025.

AI-Driven Refactoring: Semantic Reconstruction of Domain Models Using LLM Reasoning

Mohamed El BOUKHARI[1], Nassim KHARMOUM[2], Soumia ZITI[3]
IPSS Research Team-Faculty of Sciences, Mohammed V University in Rabat, Rabat, Morocco[1,3]
IPSS Research Team-Faculty of Sciences, National Center for Scientific and Technical Research (CNRST), Rabat, Morocco[2]

***Abstract*—This study examines the application of large language models (LLMs) for automating domain layer reconstruction in legacy systems, with a specific focus on a case study involving water consumption management. The process begins with a deliberately disordered JSON representation that conflates domain, application, and infrastructure issues. An LLM, specifically GPT-5.2, was employed to identify misplaced methods, inconsistent naming, DTO misuse, incoherent aggregates, and unrelated modules, and subsequently reorganize the model into a structure aligned with Domain-Driven Design (DDD). The structure includes entities, value objects, aggregates, domain services, domain events, and repositories. The methodology involves encoding the legacy model as JSON, applying an LLM-based diagnosis and reconstruction pipeline, and producing both a refined domain model and a categorized catalogue of corrections. A comparative analysis of candidate LLMs, informed by recent code-centric benchmarks, such as SWE-bench and LiveCodeBench, supports the selection of GPT-5.2 as the primary model for this study. The findings indicate that the LLM can swiftly recover key domain concepts and achieve semantically consistent refactoring, a task that typically requires extensive manual effort. This suggests that LLM-assisted domain reconstruction is a promising adjunct to traditional refactoring practices and can facilitate continuous architectural improvements in organizations.**

***Keywords*—Domain-Driven Design; large language models; AI-driven software refactoring legacy systems modernization; semantic code analysis; architecture reconstruction; GPT; LLM; domain layer reconstruction; AI-assisted software engineering**

I. Introduction

Legacy software systems represent a significant challenge in modern software engineering, where aging codebases continue to serve critical business functions despite outdated architectures and technologies. These systems often lack proper documentation, use obsolete programming languages or frameworks, and contain years of accumulated technical debt, making them difficult and risky to modify. Domain extraction is a crucial refactoring technique that involves identifying and separating distinct business domains or functional areas within monolithic systems to improve maintainability and enable modernization. However, extracting domains from legacy systems presents unique obstacles compared to working with newer codebases, including unclear business logic boundaries, tight coupling between components, and the absence of clear architectural patterns in the code. This process requires careful analysis to avoid introducing bugs or breaking existing functionality, while successfully isolating cohesive business domains that can be independently developed and deployed.

A. Domain Concepts in Modern Software Architecture

The concept of a domain in software engineering pertains to the comprehensive body of knowledge relevant to the problem at hand, encompassing the entire business area or specific support areas where particular categories of software systems are employed [1][2]. More specifically, a domain can be characterized as an area of expertise abstracted from the implementation details of the software, representing the minimal set of properties that accurately delineate a family of problems for which computer applications are necessary [1][5].

The domain layer is central to the domain-centric software architecture, encompassing the core logic of the developed software and serving as a pivotal reference throughout the application [6]. The domain model offers a representation of real-world entities and their interrelationships, which define the problem space of a system. It functions as a communication tool for all stakeholders, facilitating the emergence of concepts, such as ubiquitous language in Domain-Driven Design [3][2].

Domain engineering involves domain analysis, focusing on developing and maintaining reusable infrastructure within specific domains, and domain implementation, which includes application and component engineering [1]. The objective is to enhance domain information accessibility and extract relevant components from repositories rather than building new components. This methodology prioritizes the reusability of the analysis and design, not solely the code.

B. Challenges in Domain Extraction for Refactoring

The task of extracting domains to refactor legacy systems involves unique challenges that hinder the modernization efforts. A key challenge is determining service boundaries within monolithic architectures, which leads to inadequate service decomposition. This affects the system performance, scalability, and maintainability [4]. Legacy applications are not built with service-oriented principles but contain reusable functions that can become services [7].

A significant challenge is the structural disparity between the code organization and domain concepts. Source code rarely indicates the system's high-level components, and syntactical analysis can produce clusters that do not correspond to domain concepts owing to differing organizational patterns. This gap complicates the identification of refactoring opportunities where the benefits are evident [9].

Migration remains largely manual and labor-intensive, with methodologies providing guidance at the architectural level, often overlooking the code-level challenges [8]. When domain

models supplement static code analysis, their misalignment with the source code complicates the extraction process [7].

C. AI Domain Knowledge Extraction Capabilities

Modern AI systems effectively derive domain-specific insights from source codes using complementary techniques. Abstract Syntax Trees (ASTs) represent code's structural characteristics, enabling machine learning algorithms to discern syntactic relationships and semantic elements [10][19]. These structures allow AI models to leverage the organized nature of programming languages rather than processing code as linear text [20].

The construction of knowledge graphs is a pivotal capability in the extraction of domain knowledge, where AI systems develop graph representations with nodes that denote code entities and directed edges that define various relationships among program components [14][15]. These knowledge graphs empower AI to model complex inter-function dependencies and cross-file relationships, which are vital for a comprehensive understanding of program behavior [12].

AI systems can derive extensive semantic insights by integrating various analytical methodologies. Through static program analysis, these systems can identify essential elements, such as method overriding relationships, method calls, interface implementations, and type usage patterns [18]. Additionally, more sophisticated systems employ semantic dependency analysis to create semantic networks from imported libraries and public application programming interfaces (APIs). This process aids in the disambiguation of identifier senses, similar to the disambiguation of word senses in natural language processing [11].

The integration of ontological representations establishes a formal framework for modeling software system architecture and semantics, thereby enhancing the clarity and precision of the depiction of system relationships. This approach also facilitates dependency tracking, which is crucial for maintenance and refactoring tasks [17]. These capabilities extend across various applications, including vulnerability prediction, semantic-based code search, code summarization, classification, clone detection, and semantic error identification [13][16].

D. Semantic Analysis

In contemporary AI, advanced neural architectures conduct in-depth semantic analysis of source code beyond basic syntactic parsing. Transformer models, such as CodeBERT and CodeT5, have emerged as tools for semantic understanding, using self-supervised objectives to capture semantic and syntactic details [22][25]. These models interpret both programming and natural language contexts, while operating on raw source codes [22].

Attention mechanisms are a key technique in semantic analysis, enabling models to focus on relevant code segments during the output generation [29]. These mechanisms help AI systems better comprehend the code context in tasks such as variable naming and code summarization. Advanced methods use contrastive learning, comparing positive and negative code samples to improve representation learning and understanding of the target variables [29][23].

AI systems utilize multimodal analysis engines capable of concurrently processing various types of software artifacts, thereby discerning the semantic relationships among code implementations, requirement specifications, test cases, and documentation [21]. This holistic analysis approach considers the entire context of software development activities rather than examining the code in isolation.

Program analysis methods, such as abstract syntax trees (ASTs), control flow graphs (CFGs), and static analysis, help AI systems identify code patterns and suggest improvements [26]. Large language models (LLMs) have been effective in software engineering tasks, including code generation, summarization, clone detection, and automated repair, demonstrating their ability to understand code structure and semantics [28] [24] [27].

E. Planned Contributions and Paper Structure

This study presents three principal contributions: 1) the development of an end-to-end LLM pipeline for reconstructing domain layers from legacy artifacts, which automates the generation of entities/DTOs and the organization of subdomains; 2) the establishment of a semantic classification framework that employs function analysis and schema-driven prompts to ensure precise class placement; and 3) the initiation of an empirical evaluation framework with benchmarks for assessing fidelity, modularity, and scalability within enterprise contexts.

Section I introduces the research problem, outlines the motivation for automating domain reconstruction in legacy systems, and highlights the value of LLM-assisted refactoring. Section II reviews the related work on AI-driven refactoring and Domain-Driven Design extraction. Section III details the methodology, covering the artifact extraction, large language model prompting, and validation processes. Section IV presents a case study of a legacy monolithic system. Section V reports the evaluation results and analysis. Section VI discusses the findings. Section VII presents the limitations and future research directions, and Section VIII concludes the study.

II. Related Work

Cao et al. enhanced early neural methodologies by transitioning from basic networks for concept assignment to the application of LSTMs and Siamese RNNs for entity normalization and the development of knowledge graphs from codes [30]. Kabore et al. and Zhang et al. further refined AST-based approaches, such as ASTNN, by segmenting large trees into statement sequences [31][32], which were subsequently encoded using B-RNNs.

Keller et al. integrated multimodal representations, including lexical tokens, abstract syntax trees (ASTs), program trees and code visuals. They demonstrated the application of WySiWiM in utilizing code screenshots for image classification, achieving a performance comparable to syntax-based methods [31][34].

Zhang et al. (2024), Sachdev et al. (2018), and Ling et al. (2020) advanced the field of code search by transitioning from TF-IDF to the utilization of graph neural networks and attention mechanisms [33][35][37]. Furthermore, Rahman et al. (2023) and Ling et al. (2020) facilitated pattern transfer

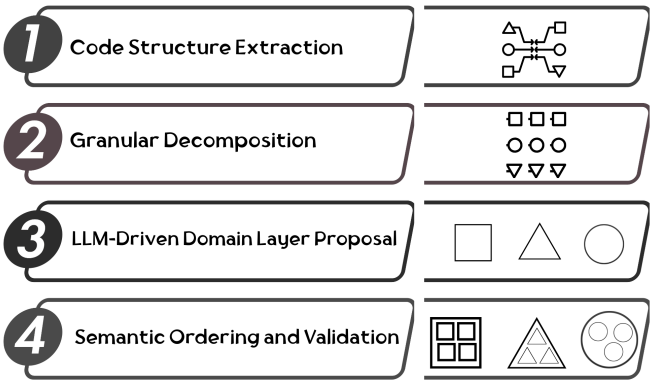


Fig. 1. Reconstruction pipeline.

in AdaCS by implementing domain embeddings and syntax matching [36][37].

Chowdhury et al. (2025), Chimalakonda et al. (2023), Chaieb et al. (2024), and Al-Debagy (2021) have integrated Abstract Syntax Trees (ASTs), Control Flow Graphs (CFGs), and Data Flow Graphs (DFGs) to capture behavior, thereby extending code2vec embeddings for the extraction of microservices from monolithic architectures [38][44][42][43]. This study directly informs the reconstruction of domain layers from legacy artifacts using large language models (LLMs). These studies underscore the semantic gaps that our LLM schema pipeline addresses in the generation of entities and Data Transfer Objects (DTOs).

A growing body of research explores how AI, and more recently, LLM-based techniques can automate refactoring and testing within software development cycles. Boukhlif et al. showed that intelligent testing approaches can reduce manual validation efforts in safety-critical medical systems [39], while El Boukhari et al. demonstrated AI-assisted refactoring for CQRS-based architectures and healthcare systems using NLP and design patterns to guide structural improvements and enhance testability [41][40]. These contributions support the feasibility of automating substantial portions of refactoring and testing, motivating the use of LLMs as a more semantically capable foundation for domain-driven reconstruction, as pursued in this study.

III. Methodology

The methodology delineates a systematic LLM-driven refactoring pipeline for legacy codebases, as shown in Fig. 1. This process transforms monolithic structures or incomplete domains into semantically coherent, lightweight domain layers.

A. LLM Choice

An advanced large language model serves as the primary reasoning engine, which is chosen for processing structured JSON, deducing domain concepts, and performing semantic transformations on software models. The model is applied throughout the pipeline, ensuring that the analysis, restructuring, and explanation are based on a cohesive semantic understanding. A comparative study of state-of-the-art language models is needed to evaluate their performance on code and architecture tasks to identify the most suitable model.

B. Reconstruction Pipeline

The methodology is delineated into four principal stages that outline the utilization of the LLM in reconstructing the domain layer as follows:

1) Code structure extraction: The initial step involves analyzing a JSON representation of the structure of the legacy codebase, with the assumption that method names convey meaningful semantic information (e.g., "calculateUserDiscount" suggests discount-related logic). This JSON encapsulates hierarchical components, including classes, methods, parameters, dependencies, and call graphs, within a standardized schema: {"classes": [{"name": str, "methods": [{"name": str, "params": list, "body_summary": str, "dependencies": list}]}]}. Parsing tools such as tree-sitter or AST analyzers are employed to extract this information without modifying the source code, thereby facilitating the subsequent semantic analysis.

2) Granular decomposition: The extracted structure was decomposed into the smallest viable functions, adhering to the single-responsibility principle. Each method is recursively divided by identifying cohesive blocks, such as loops and conditionals, through the semantic clustering of method names and body summaries. Fragments are renamed to preserve context, for example, "processDiscount" is divided into "validateEligibleDiscount" and "computeDiscountRate". Atomicity is ensured by limiting functions to a single semantic intent, which is verified through validation prompts such as "Is this function doing one thing only?" The output is a refined JSON that contains nested microfunctions.

3) LLM-driven domain layer proposal: By utilizing the decomposed JSON, a large language model (LLM) proposes an innovative domain layer structure that is semantically aligned with the intended purpose of the codebase. An LLM identifies domain concepts and adheres to the principles of domain-driven design (DDD).

4) Semantic ordering and validation: The proposed structure was enhanced using LLM-ordered topological sorting. The prompt validation process involves iterating through simulated refactoring paths, verifying semantic preservation, such as the cosine similarity of name embeddings before and after refactoring, and refining the process using few-shot examples from refactoring datasets. The final output is an executable JSON/YAML scaffold prepared for code generation and maintaining traceability to the original legacy elements. This pipeline improves automation in empirical refactoring studies, thereby reducing the cognitive load on engineers.

IV. Case Study

The case study examined a legacy water management system, as presented in Fig. 2, wherein the domain model was restructured into a JSON representation of the DomainLayer. This JSON format encapsulates the domain concepts in Fig. 3, including clients, water meters, and consumption records, as well as services, value objects, aggregates, and events, while highlighting design and layering issues. The WaterModule delineates the primary domain entities: a client characterized by identification and categorization attributes with a one-to-many relationship with the WaterMeter; a WaterMeter associated with location and linked consumption data; and a

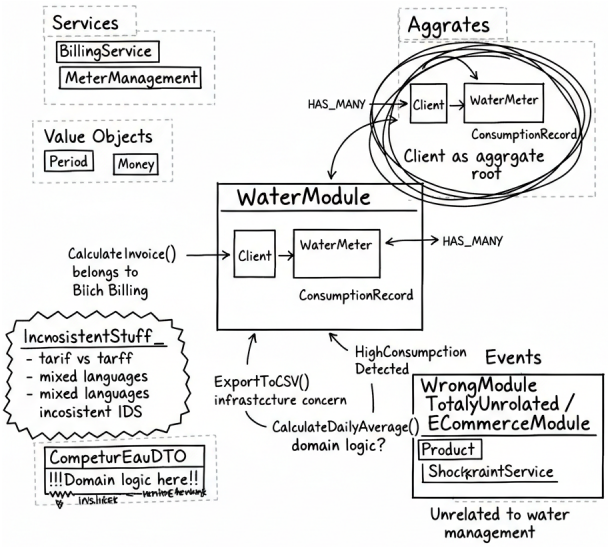


Fig. 2. Water meter system domain.

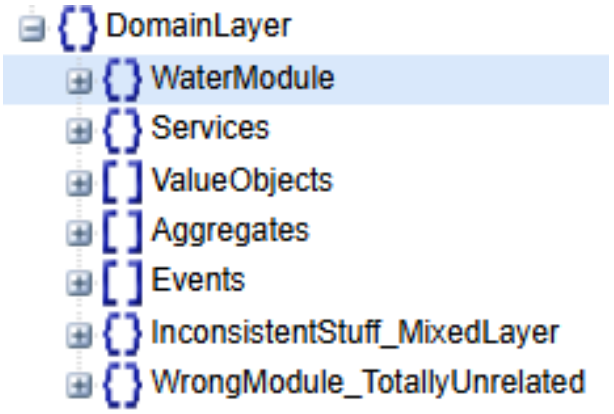


Fig. 3. Initial domain structure.

ConsumptionRecord that details metered usage over billing periods. The representation also enumerates domain services (e.g., BillingService and MeterManagement), value objects (e.g., Period and Money), aggregates rooted in Client, and domain events such as HighConsumptionDetected, offering a structural overview of the system.

The JSON format encodes layering violations and misplaced responsibilities through wrongMethods annotations across entities, services, value objects, aggregates, and events. The Client entity contains invoicing and notification functionalities (CalculateInvoice, SendEmailNotification) that are better suited for billing or infrastructure layers, whereas the WaterMeter entity includes export functionality (ExportToCSV) that is specific to application concerns rather than domain logic. Value objects such as Period and Money contain inappropriate persistence methods, and aggregates and events include messaging behaviors. A mixed layer, InconsistentStuff_MixedLayer, documents naming and language inconsistencies, whereas WrongModule_TotallyUnrelated shows an e-commerce module erroneously integrated with the water domain. This imperfect model serves as an input for the LLM-based approach to identify and realign responsibilities for an improved domain layer architecture.

V. Results

GPT-5.2 was identified as the primary large language model based on the comparative analysis presented in Table I. Among the models evaluated [45][46][47], GPT-5.2 exhibited superior performance on code-centric benchmarks, such as SWE-bench and LiveCodeBench, and excelled in repository-scale reasoning, semantic refactoring and domain-oriented architecture synthesis. These capabilities are directly aligned with the requirements of the proposed method. Although alternative models, including Claude 4.5 Sonnet and Kimi-Dev-72B, were retained as secondary references to contextualize the results, the detailed case study focused on GPT-5.2 to ensure a comprehensive and consistent evaluation.

The LLM initially performs a thorough semantic analysis of the existing JSON domain model, identifying misplaced methods, incorrect or unrelated modules, inconsistent naming conventions, mixed layers, and anti-patterns, such as Data Transfer Objects (DTOs) within the domain, entities containing service logic, value objects exhibiting persistence behavior, incoherent aggregates, and behavioral events. Subsequently, the model is transformed into a Domain-Driven Design (DDD)-aligned Domain Layer by reorganizing the content into canonical building blocks, including entities, value objects, aggregates, domain services, domain events, and repositories. This process involves relocating misplaced methods to the appropriate layers, standardizing naming conventions, and isolating unrelated modules as out-of-scope while maintaining traceability.

The LLM implements a series of targeted corrections that can be categorized into nine distinct areas. Initially, it removes **misplaced methods** from entities, value objects, aggregates, events, and DTOs (e.g., notifications, exports, persistence, webhooks), relocating authentic domain logic into domain services, and documenting technical concerns as out-of-scope infrastructure hints. Subsequently, it isolates **wrong or unrelated modules**, notably extracting the e-commerce module into a separate "UnrelatedModules" section with a clear rationale. **Naming conventions** are standardized by aligning identifiers, field names, value-object attributes, categories, and types to consistent English, PascalCase, and coherent ID conventions. To **address mixed layers**, infrastructure responsibilities (email, SMS, webhooks, CSV/PDF, persistence) are removed from the domain, with repository interfaces introduced to replace the entity-level data access methods. **DTOs are relocated** to a dedicated DTO section and stripped of domain logic, while **entities are cleaned** of calculation, notification, and retrieval behavior, which is centralized into dedicated domain services such as billing and meter management. **Value objects** are restored to immutability and behavioral purity by eliminating persistence-related operations. Aggregates are clarified so that a Client aggregate coherently contains WaterMeter, ConsumptionRecord, and Invoice, a WaterMeter aggregate encapsulates ConsumptionRecord, and all aggregate behaviors are removed. Finally, **domain events** are restricted to payload and description only, with behavioral aspects (e.g., webhooks) relocated to the appropriate application or integration layers.

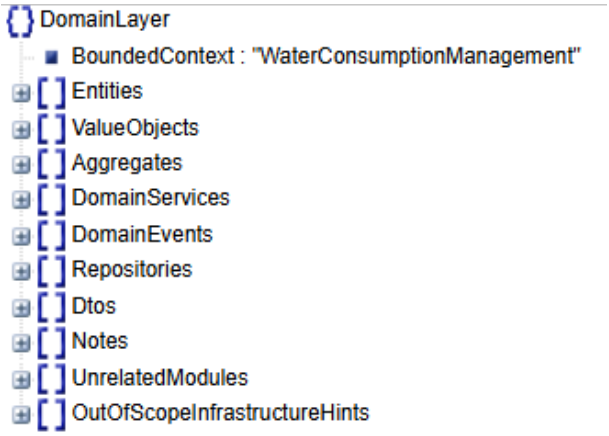


Fig. 4. Refactored domain tree view.

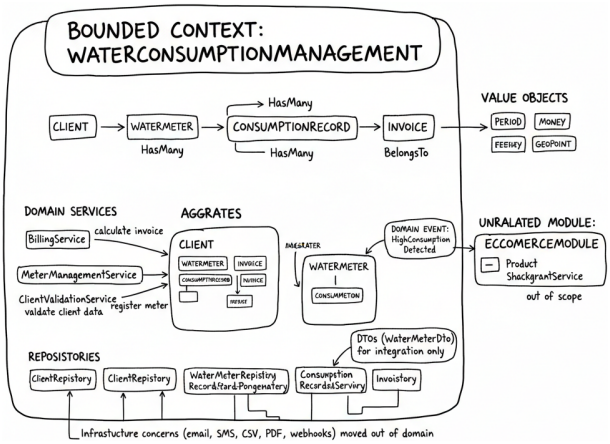


Fig. 5. Water meter system: new refactored domain.

The two figures present the LLM-driven reconstruction of the domain layer. Fig. 4 shows a refactored JSON tree view of the DomainLayer, with a cleaned bounded context and separated core domain elements (entities, value objects, aggregates, services, events, and repositories). Fig. 5 shows the domain architecture as a conceptual diagram, depicting the WaterConsumptionManagement bounded context with its main entities (Client, WaterMeter, ConsumptionRecord, Invoice), aggregates, value objects, and domain services, illustrating the final DDD-aligned structure.

The reconstructed model effectively illustrates the capability of large language models (LLMs) to rapidly identify key domain components with minimal guidance from humans. From the unstructured JSON data, the model consistently identified the central entities—Client, WaterMeter, ConsumptionRecord, and Invoice—along with their associated value objects, such as Period, Money, and GeoPoint. Furthermore, it recognized primary behavioral anchors, including billing and meter management services, even when these concepts were dispersed across various layers and inconsistent naming conventions. This automatic detection and clustering of core domain elements provide a robust foundation for architects, enabling them to refine aggregates and responsibilities rather than manually rediscovering the fundamental structure of a domain.

In a single execution, the LLM adeptly generated both the refined domain model and a list of necessary corrections. In contrast, the manual process of refactoring a domain layer of equivalent complexity generally requires a considerably greater investment of time from architects and domain experts.

VI. Discussion

The selection of GPT-5.2 as the primary model influenced the validity and generalizability of this study. Models in the GPT-5 class achieve state-of-the-art results on code benchmarks, such as SWE-bench and LiveCodeBench [45][46][47], which assess repository-scale reasoning and multi-file modification. These capabilities align with the requirements of this study, where the LLM must reason over JSON representations and identify architectural issues to reconstruct domain models. Utilizing this tool allows for the observation of the maximum potential of current LLM technology for domain-driven refactoring. Claude 4.5 Sonnet and Kimi-Dev-72B provide context within the broader model landscape, although the absence of a full parallel analysis limits cross-model conclusions. Future research should test multiple LLMs to evaluate how the choice of model affects the reconstruction quality.

The LLM exhibits two-phase behavior, aligned with the evidence of LLM-based refactoring capabilities [48][49]. Initially, it performs semantic analysis of the JSON domain model, detecting misplaced methods, unrelated modules, inconsistent naming, mixed layers, and anti-patterns, including embedded DTOs, entities with service logic, and events containing behavior. Studies have confirmed that LLMs can identify code smells and refactoring needs across code bases, particularly for recurring patterns. In the second phase, the LLM reorganizes the model into a DDD-aligned Domain Layer by mapping elements to canonical building blocks, relocating methods, normalizing names, and isolating unrelated modules while maintaining traceability. This aligns with research showing that LLMs effectively enforce architectural rules and layered separations when explicitly prompted.

The findings suggest that the LLM can execute a substantial, semantically coherent restructuring of the domain model in a significantly reduced timeframe compared to that typically required by human architects. Empirical research on LLM-based refactoring and automated tools demonstrates that these systems are particularly adept at addressing systematic, repetitive issues and can produce extensive refactoring suggestions at a much faster rate than developers [48], who may require several hours or days to implement comparable changes in nontrivial systems.

The reconstructed model effectively illustrates the capability of large language models to rapidly surface key domain components with limited human supervision. From the unstructured and inconsistent JSON, the LLM consistently recovered the central entities (Client, WaterMeter, ConsumptionRecord, Invoice), their associated value objects (Period, Money, GeoPoint), and the main behavioral anchors related to billing and meter management, despite these concepts being scattered across layers and expressed with heterogeneous naming conventions. This aligns with recent empirical work

TABLE I. COMPARISON OF LLMS FOR LEGACY CODE REFACTORING AND DOMAIN MODELING

Aspect	GPT-5.2 (OpenAI)	Claude 4.5 Sonnet (Anthropic)	Kimi-Dev-72B (Open-source)
Code benchmarks (e.g., SWE-bench, LiveCodeBench)	Among the top scores on SWE-bench and LiveCodeBench, repository-scale reasoning and transformation were very strong.	Competitive on coding and reasoning; slightly below frontier models on the hardest coding tasks.	Best open-source scores on SWE-bench Verified (around 60%); below frontier closed models but strong for OSS.[1]
Large-repository understanding	128k–200k+ context window; robust handling of mixed code and documentation for architecture tasks.	Very strong long-context reasoning (up to approximately 200k tokens); effective for narrative architecture analysis.	They have good context and stability but are more limited; large codebases typically require chunking strategies.
Refactoring and decomposition	It excels at stepwise refactoring, function extraction, and semantic renaming using chain-of-thought prompting.	Strong in explaining and refactoring code but slightly less aggressive by default on structural transformations.	Good at classic refactorings; more difficulty with deep semantic decomposition without careful prompting.
Domain and architecture modeling	It is very good at proposing DDD-like aggregates, domain layers, and JSON/YAML blueprints from the code structure.	Excellent at high-level conceptual modeling and textual rationale and strong for architecture narratives and trade-offs.	It can infer domains, but abstractions are less consistent and require more prompt engineering.
Tooling and ecosystem	Mature APIs and assistants for coding, IDE integration, CI, and code-review workflows.	Solid integration of conversational and analysis-centric workflows, including documentation and reviews.	Full local control and good OSS tooling integration, but higher DevOps and infrastructure overheads.
Openness and reproducibility	Closed weights: experiments reproducible via documented prompts and configs, but not locally hostable.	Closed model with reproducibility constraints similar to those of other proprietary systems.	Open weights; well-suited for fully reproducible, self-hosted experimental pipelines in research.

[51][50][5][48], which shows that LLMs can reliably extract domain concepts and modular structures from existing code and documentation, often matching or approaching human performance in tasks such as entity discovery, feature localization, and architectural view reconstruction, while requiring significantly less time. In practice, this automatic detection and clustering of core domain elements provides a strong starting point for architects, who can concentrate on refining aggregates, invariants, and responsibility boundaries instead of manually rediscovering the fundamental structure of the domain—a role for LLMs that is also emphasized in studies on AI-augmented software architecture and DDD refactoring in industrial contexts.

Although this study does not explicitly quantify cost or time, the capability of the LLM to generate a coherent domain reconstruction in a single iteration implies that it can potentially supplant numerous hours of detailed refactoring work typically undertaken by multiple developers. In practical applications, developers' efforts can be predominantly concentrated on validating and refining the LLM's proposal rather than executing the restructuring itself [51][48][52]. This shift in focus reduces the manual workload and, indirectly, the staffing and time costs that are generally associated with large-scale refactoring projects.

VII. LIMITATIONS AND FUTURE DIRECTIONS

A. Limitations

Despite promising outcomes, several challenges persist in the utilization of LLMs for domain-layer reconstruction. The model's decision-making process remains partially opaque: while it identifies recurring anti-patterns (e.g., misplaced methods), its rationale for nuanced design decisions, such as determining aggregate boundaries or allocating responsibilities between services, can be difficult to audit. The performance of LLMs is sensitive to prompt design and input representation; changes in the JSON structure or constraint phrasing may yield different reconstructions, raising concerns about stability compared to traditional refactoring approaches. Current models struggle with implicit domain knowledge and non-obvious invariants that are not explicitly encoded in the inputs, potentially oversimplifying business rules, as noted in studies of AI-augmented software architecture and DDD refactoring.

B. Future Direction

AI-driven refactoring is evolving towards more autonomous and intelligent systems that function as collaborative partners rather than suggestion tools.

Agentic coding tools represent a significant advancement in the role of artificial intelligence in refactoring processes, acting as collaborators that autonomously plan, execute, test, and refine complex development tasks with minimal supervision. Despite this potential, current agentic refactoring is largely limited to low-level, consistency-driven modifications, such as renaming and type adjustments, with agents performing fewer high-level design transformations than human efforts. To realize the potential of agents as "software architects", substantial progress is required to enable autonomous architecturally aware restructuring that addresses higher-level design challenges.

Future advancements are anticipated to focus on developing advanced AI-driven tools that effectively assist in both development and maintenance processes. These tools include automated debugging assistants, refactoring advisors, code summarizers, and documentation generators. The design of these tools must consider complex factors, such as historical usage patterns, module dependencies, backward compatibility requirements, and runtime constraints. Therefore, it is crucial for these tools to integrate seamlessly with developer workflows, including version control systems, CI/CD pipelines and issue trackers.

The field is advancing towards the development of more sophisticated automated refactoring tools capable of proposing specific refactoring strategies. These tools may, for instance, suggest the division of large modules into smaller, interconnected units with enhanced cohesion, with AI systems potentially executing these recommendations autonomously (Khanzadeh et al., 2023). Future developments are expected to include the advent of domain-specific AI models, increased transparency of AI-driven solutions, and collaborative tools

designed to integrate refined and updated development methodologies.

VIII. Conclusion

This study illustrates the effective application of advanced large language models (LLMs) in automating the semantic reconstruction of domain models from legacy software artifacts. Through the implementation of a structured LLM-driven pipeline, this approach adeptly identifies and reorganizes key domain elements, rectifies architectural inconsistencies, and aligns legacy codebases with Domain-Driven Design principles. This automation substantially diminishes the manual effort traditionally associated with domain extraction and refactoring, facilitating a more rapid, consistent, and semantically coherent modernization of legacy systems. A case study on a sample water management system demonstrates the method's capability to detect misplaced responsibilities, standardize naming conventions, and isolate unrelated modules while maintaining traceability. Future advancements in AI-driven refactoring hold the promise of increasingly autonomous and intelligent tools that will further transform software refactoring into an adaptive, explainable, and efficient process, fostering collaboration between human architects and AI agents.

References

[1] F. Losavio, A. Matteo, and I. Pacilli Camejo, "Unified Process for Domain Analysis integrating Quality, Aspects and Goals," CLEIej, vol. 17, no. 2, Aug. 2014, doi: 10.19153/cleiej.17.2.1.

[2] H. S. Da Silva, G. Carneiro, and M. Monteiro, "Towards a Roadmap for the Migration of Legacy Software Systems to a Microservice based Architecture:," in Proceedings of the 9th International Conference on Cloud Computing and Services Science, Heraklion, Crete, Greece: SCITEPRESS - Science and Technology Publications, 2019, pp. 37–47. doi: 10.5220/0007618400370047.

[3] S. Meliá, R. Reyes, and C. Cachero, "The Effect of Developers' General Intelligence on the Understandability of Domain Models: An Empirical Study," IEEE Access, vol. 11, pp. 70153–70167, 2023, doi: 10.1109/ACCESS.2023.3293199.

[4] I. Trabelsi, B. Mahmoudi, J. B. Minani, N. Moha, and Y.-G. Guéhéneuc, "A Systematic Literature Review of Machine Learning Approaches for Migrating Monolithic Systems to Microservices," Aug. 21, 2025, arXiv: arXiv:2508.15941. doi: 10.48550/arXiv.2508.15941.

[5] O. Özkan, Ö. Babur, and M. Van Den Brand, "Refactoring with domain-driven design in an industrial context: An action research report," Empir Software Eng, vol. 28, no. 4, p. 94, July 2023, doi: 10.1007/s10664-023-10310-1.

[6] S. Flaga and K. Pacholczak, "Demonstrator of a Digital Twin for Education and Training Purposes as a Web Application," Adv. Sci. Technol. Res. J., vol. 16, no. 5, pp. 110–119, Nov. 2022, doi: 10.12913/22998624/152927.

[7] H. Mili et al., "Service-Oriented Re-engineering of Legacy JEE Applications: Issues and Research Directions," June 03, 2019, arXiv: arXiv:1906.00937. doi: 10.48550/arXiv.1906.00937.

[8] R. Peixoto, F. F. Correia, T. Rosa, E. Guerra, and A. Goldman, "Refactoring Towards Microservices: Preparing the Ground for Service Extraction," Oct. 03, 2025, arXiv: arXiv:2510.03050. doi: 10.48550/arXiv.2510.03050.

[9] D. van der Leij, J. Binda, R. van Dalen, P. Vallen, Y. Luo, and M. Aniche, "Data-Driven Extract Method Recommendations: A Study at ING," July 22, 2021, arXiv: arXiv:2107.05396. doi: 10.48550/arXiv.2107.05396.

[10] E. Spirin, E. Bogomolov, V. Kovalenko, and T. Bryksin, "PSIMiner: A Tool for Mining Rich Abstract Syntax Trees from Code," Mar. 23, 2021, arXiv: arXiv:2103.12778. doi: 10.48550/arXiv.2103.12778.

[11] A. Saeidi, J. Hage, R. Khadka, and S. Jansen, "On the Effect of Semantically Enriched Context Models on Software Modularization," Programming, vol. 2, no. 1, p. 2, Aug. 2017, doi: 10.22152/programming-journal.org/2018/2/2.

[12] S. Khemka and M. Architect, "Legacy Modernization with AI - Mainframe modernization".

[13] L. Buch and A. Andrzejak, "Learning-Based Recursive Aggregation of Abstract Syntax Trees for Code Clone Detection," in 2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER), Hangzhou, China: IEEE, Feb. 2019, pp. 95–104. doi: 10.1109/SANER.2019.8668039.

[14] A. Qayum, S. U. R. Khan, Inayat-Ur-Rehman, and A. Akhunzada, "FineCodeAnalyzer: Multi-Perspective Source Code Analysis Support for Software Developer Through Fine-Granular Level Interactive Code Visualization," IEEE Access, vol. 10, pp. 20496–20513, 2022, doi: 10.1109/ACCESS.2022.3151395.

[15] J. Zhang, R. Xie, W. Ye, Y. Zhang, and S. Zhang, "Exploiting Code Knowledge Graph for Bug Localization via Bi-directional Attention," in Proceedings of the 28th International Conference on Program Comprehension, Seoul Republic of Korea: ACM, July 2020, pp. 219–229. doi: 10.1145/3387904.3389281.

[16] Z. Bilgin, "Code2Image: Intelligent Code Analysis by Computer Vision Techniques and Application to Vulnerability Prediction," May 07, 2021, arXiv: arXiv:2105.03131. doi: 10.48550/arXiv.2105.03131.

[17] A. S. Da Silva, R. E. Garcia, and L. C. Botega, "Bug Localization Model in Source Code Using Ontologies," IEEE Access, vol. 11, pp. 98542–98557, 2023, doi: 10.1109/ACCESS.2023.3313598.

[18] University of Oviedo, Computer Science Department, c/Calvo Sotelo s/n, 33007, Oviedo, Spain, F. Ortin, J. Escalada, and O. Rodriguez-Prieto, "Big Code: New Opportunities for Improving Software Construction," JSW, vol. 11, no. 11, pp. 1083–1008, Nov. 2016, doi: 10.17706/jsw.11.11.1083-1088.

[19] Y. Shido, Y. Kobayashi, A. Yamamoto, A. Miyamoto, and T. Matsumura, "Automatic Source Code Summarization with Extended Tree-LSTM," June 20, 2019, arXiv: arXiv:1906.08094. doi: 10.48550/arXiv.1906.08094.

[20] U. Alon, M. Zilberstein, O. Levy, and E. Yahav, "A General Path-Based Representation for Predicting Program Properties," Apr. 22, 2018, arXiv: arXiv:1803.09544. doi: 10.48550/arXiv.1803.09544.

[21] M. Hariharan, "Semantic mastery- Enhancing LLMs with Advanced Natural Language Understanding".

[22] S. Nursapa, A. Samuilova, A. Bucaioni, and P. T. Nguyen, "ROSE: Transformer-Based Refactoring Recommendation for Architectural Smells," July 20, 2025, arXiv: arXiv:2507.12561. doi: 10.48550/arXiv.2507.12561.

[23] H. Liu et al., "RefBERT: A Two-Stage Pre-trained Framework for Automatic Rename Refactoring," May 28, 2023, arXiv: arXiv:2305.17708. doi: 10.48550/arXiv.2305.17708.

[24] N. Jiang, T. Lutellier, and L. Tan, "CURE: Code-Aware Neural Machine Translation for Automatic Program Repair," in 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE), May 2021, pp. 1161–1173. doi: 10.1109/ICSE43902.2021.00107.

[25] Z. Feng et al., "CodeBERT: A Pre-Trained Model for Programming and Natural Languages," Sept. 18, 2020, arXiv: arXiv:2002.08155. doi: 10.48550/arXiv.2002.08155.

[26] M. R. Tapader, M. M. Rahman, A. I. Shiplu, M. F. I. Amin, and Y. Watanobe, "Code Refactoring with LLM: A Comprehensive Evaluation With Few-Shot Settings," Nov. 26, 2025, arXiv: arXiv:2511.21788. doi: 10.48550/arXiv.2511.21788.

[27] E. Mashhadi and H. Hemmati, "Applying CodeBERT for Automated Program Repair of Java Simple Bugs," Mar. 30, 2021, arXiv: arXiv:2103.11626. doi: 10.48550/arXiv.2103.11626.

[28] S. Afrin, M. Z. Haque, and A. Mastropaolo, "A Systematic Literature Review of Parameter-Efficient Fine-Tuning for Large Code Models," Aug. 15, 2025, arXiv: arXiv:2504.21569. doi: 10.48550/arXiv.2504.21569.

[29] B. Nyirongo, Y. Jiang, H. Jiang, and H. Liu, "A Survey of Deep Learning Based Software Refactoring".

[30] K. Cao and J. Fairbanks, "Unsupervised Construction of Knowledge Graphs From Text and Code," Aug. 25, 2019, arXiv: arXiv:1908.09354. doi: 10.48550/arXiv.1908.09354.

[31] A. K. Kaboré, E. T. Barr, J. Klein, and T. F. Bissyandé, "CodeGrid: A Grid Representation of Code," in Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis, Seattle WA USA: ACM, July 2023, pp. 1357–1369. doi: 10.1145/3597926.3598141.

[32] J. Zhang, X. Wang, H. Zhang, H. Sun, K. Wang, and X. Liu, "A Novel Neural Source Code Representation Based on Abstract Syntax Tree," in 2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE), Montreal, QC, Canada: IEEE, May 2019, pp. 783–794. doi: 10.1109/ICSE.2019.00086.

[33] F. Zhang, M. Li, H. Wu, and T. Wu, "Intelligent code search aids edge software development," J Cloud Comp, vol. 13, no. 1, p. 78, Apr. 2024, doi: 10.1186/s13677-024-00629-5.

[34] P. Keller, L. Plein, T. F. Bissyandé, J. Klein, and Y. L. Traon, "What You See is What it Means! Semantic Representation Learning of Code based on Visualization and Transfer Learning," Feb. 07, 2020, arXiv: arXiv:2002.02650. doi: 10.48550/arXiv.2002.02650.

[35] S. Sachdev, H. Li, S. Luan, S. Kim, K. Sen, and S. Chandra, "Retrieval on source code: a neural code search," in Proceedings of the 2nd ACM SIGPLAN International Workshop on Machine Learning and Programming Languages, Philadelphia PA USA: ACM, June 2018, pp. 31–41. doi: 10.1145/3211346.3211353.

[36] M. M. Rahman, Y. Watanobe, A. Shirafuji, and M. Hamada, "Exploring Automated Code Evaluation Systems and Resources for Code Analysis: A Comprehensive Survey," July 08, 2023, arXiv: arXiv:2307.08705. doi: 10.48550/arXiv.2307.08705.

[37] X. Ling et al., "Deep Graph Matching and Searching for Semantic Code Retrieval," ACM Trans. Knowl. Discov. Data, vol. 15, no. 5, pp. 1–21, Oct. 2021, doi: 10.1145/3447571.

[38] J. M. Chowdhury, A. F. S. Chowdhury, H. B. Monwar, and M. Naznin, "ATLAS: Automated Tree-based Language Analysis System for C and C++ source programs," Dec. 14, 2025, arXiv: arXiv:2512.12507. doi: 10.48550/arXiv.2512.12507.

[39] Boukhlif, M., Kharmoum, N., Hanine, M., Elasri, C., Rhalem, W., & Ezziyyani, M. (2024). Exploring the application of classical and intelligent software testing in medicine: A literature review. In the Proceedings of the International Conference on Advanced Intelligent Systems for Sustainable Development (pp. 37-46). Springer, Cham. DOI: 10.1007/978-3-031-52388-5-4

[40] M. EL Boukhari, N. Kharmoum, S. N. Lagmiri, and S. Ziti, "Enhancing CQRS Implementation Through AI-Driven Refactoring: Query and Command Classification," in Smart Business and Technologies, S. N. Lagmiri, M. Lazaar, and F. M. Amine, Eds., Cham: Springer Nature Switzerland, 2025, pp. 384–390.

[41] M. El Boukhari, S. Retal, N. Kharmoum, F. Saoiabi, S. Ziti, and W. Rhalem, "An Approach for Refactoring System Healthcare Using CQRS, GoF, and Natural Language Processing," in International Conference on Advanced Intelligent Systems for Sustainable Development (AI2SD'2023), M. Ezziyyani, J. Kacprzyk, and V. E. Balas, Eds., Cham: Springer Nature Switzerland, 2024, pp. 47–55.

[42] M. chaieb and M. A. Saied, "Migration to Microservices: A Comparative Study of Decomposition Strategies and Analysis Metrics," Feb. 13, 2024, arXiv: arXiv:2402.08481. doi: 10.48550/arXiv.2402.08481.

[43] O. Al-Debagy and P. Martinek, "A Microservice Decomposition Method Through Using Distributed Representation of Source Code," SCPE, vol. 22, no. 1, pp. 39–52, Feb. 2021, doi: 10.12694/scpe.v22i1.1836.

[44] S. Chimalakonda, D. Das, A. Mathai, S. Tamilselvam and A. Kumar, "The Landscape of Source Code Representation Learning in AI-Driven Software Engineering Tasks," 2023 IEEE/ACM 45th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), Melbourne, Australia, 2023, pp. 342-343, doi: 10.1109/ICSE-Companion58688.2023.00098. keywords: Surveys;Codes;Source coding;Pipelines;Semantics;Tutorials;Syntactics;Code Representation;Machine Learning,

[45] R. Aleithan, H. Xue, M. M. Mohajer, E. Nnorom, G. Uddin, and S. Wang, "SWE-Bench+: Enhanced Coding Benchmark for LLMs," Oct. 10, 2024, arXiv: arXiv:2410.06992. doi: 10.48550/arXiv.2410.06992.

[46] N. Jain et al., "LiveCodeBench: Holistic and Contamination Free Evaluation of Large Language Models for Code," June 06, 2024, arXiv: arXiv:2403.07974. doi: 10.48550/arXiv.2403.07974.

[47] Z. Yang et al., "Kimi-Dev: Agentless Training as Skill Prior for SWE-Agents," Dec. 08, 2025, arXiv: arXiv:2509.23045. doi: 10.48550/arXiv.2509.23045.

[48] B. Liu, Y. Jiang, Y. Zhang, N. Niu, G. Li, and H. Liu, "An Empirical Study on the Potential of LLMs in Automated Software Refactoring," Nov. 07, 2024, arXiv: arXiv:2411.04444. doi: 10.48550/arXiv.2411.04444.

[49] R. Morales, F. Khomh, and G. Antoniol, "RePOR: Mimicking humans on refactoring tasks. Are we there yet?," May 17, 2019, arXiv: arXiv:1808.04352. doi: 10.48550/arXiv.1808.04352.

[50] M. Robredo, M. Esposito, F. Palomba, R. Peñaloza, and V. Lenarduzzi, "What Were You Thinking? An LLM-Driven Large-Scale Study of Refactoring Motivations in Open-Source Projects," Sept. 09, 2025, arXiv: arXiv:2509.07763. doi: 10.48550/arXiv.2509.07763.

[51] "AI-Augmented Software Architecture: Autonomous Refactoring with Design Pattern Awareness," IJETCSIT, vol. 6, 2025, doi: 10.63282/3050-9246.IJETCSIT-V6I3P113.

[52] S. Abrahão, J. Grundy, M. Pezzè, M.-A. Storey, and D. A. Tamburri, "Software Engineering by and for Humans in an AI Era," ACM Trans. Softw. Eng. Methodol., vol. 34, no. 5, pp. 1–46, June 2025, doi: 10.1145/3715111.

Enhancing Telecom Churn Prediction Using Emotion-Driven and Behavioral Engagement Features

Huthaifa Aljawazneh
Faculty of Business-Department of Business Intelligence, Al-Zaytoonah University of Jordan, Amman, Jordan

***Abstract*—Accurate churn prediction enables service providers to develop effective retention strategies and promotes revenue stability in the telecommunication industry. This study enhances churn prediction performance by extracting five emotion-driven and behavioral engagement features from a telecom churn dataset. The new features represent derived, experience-oriented indicators constructed from operational usage data rather than direct psychological or survey-based measurements. To assess the effect of these engineered features on predictions, three powerful classifiers (i.e., CatBoost, Random Forest, and XGBoost) were trained and tested in a structured three-stage experimental design. In the first stage, the classifiers were trained and tested using the original dataset (original features only). In the second stage, the original dataset was enriched with five newly derived features (i.e., frustration index, trust score, satisfaction index, service usage score, and international experience index). Finally, in the third stage, only the engineered features were used in the classification process to evaluate their standalone predictive capability. Because the dataset is imbalanced, SMOTE and SMOTE-Tomek were applied to address this issue. The results demonstrate that incorporating these engineered features improves churn prediction performance across the reported evaluation metrics (accuracy, precision, recall, and specificity) for the classifiers and balancing techniques combinations presented. The enriched dataset (original + engineered features) achieves the strongest overall performance compared to using either original features only or engineered features only. Compared to the original features only, the enriched dataset achieved improvements of up to 3.6% in accuracy and 5.8% in recall. These findings indicate that emotion-driven and behavioral engagement features provide meaningful complementary information that enhances churn prediction effectiveness.**

***Keywords*—Customer churn prediction; emotion-driven features; feature engineering; imbalanced data; classification**

I. Introduction

In the telecom sector, where acquiring new customers is more expensive than retaining existing ones, customer attrition remains a major challenge [1]. Subscriber churn can be viewed as an outcome of the overall service experience that the subscriber has with the service provider, for instance, in Jordan's telecom environment, switching to another service provider is affected by the market competitiveness, costs, and service delivery [2]. Accordingly, the main purpose of telecom churn prediction is to support the development of effective retention strategies. In addition, it allows telecom companies to address customer concerns and meet expectations. Therefore, by predicting churn, service providers can identify and pay more attention to customers who are likely to churn to a competing provider, and develop targeted solutions and personalized offers to improve retention [3]. From a practical perspective, churn prediction models are most useful when they not only predict churn accurately, but also help explain the underlying reasons for customer churn.

However, most existing studies have focused on improving churn prediction using sampling techniques, feature selection, or model optimization, while relying mainly on original telecom attributes [3], [4], [5], [6], [7], [8]. These attributes provide descriptive information about service usage, but do not fully reflect customers' experiences or level of engagement with the service. In particular, traditional telecom features typically capture what customers did, while offering limited insight into how customers perceived the service, such as whether they were frustrated or satisfied. Accordingly, many churn models achieve strong predictive performance, but provide limited explainability from a customer experience perspective. Therefore, a clear research gap exists in the literature regarding the development and evaluation of experience-oriented features that represent customers' emotional states and engagement levels using operational telecom data.

To address this limitation, this study proposes a set of emotional and behavioral indicators derived from telecom data to reflect customer experience and engagement. In other words, five engineered features are grouped into two categories: emotion-driven features (i.e., frustration index, satisfaction index, and trust score) and behavioral engagement features (i.e., service usage score and international experience index) are derived from existing telecom data and aim to better represent customer experience, which supports a more behavior-oriented churn prediction.

In addition, to evaluate the effects of the proposed features on the classification performance, three ensemble classifiers (i.e., Random Forest [9], CatBoost [10], and XGBoost [11]) were trained and tested in three sequential experimental stages. In the first stage, classifiers were trained using the original telecom dataset to provide an initial reference for churn prediction performance. In the second stage, the original dataset was enriched with the proposed emotion-driven and behavioral engagement features. In the third stage, only the newly engineered features were used to train and test the classifiers to assess their standalone predictive capability. Moreover, since the dataset used in this study is imbalanced, SMOTE [12] and SMOTE-Tomek [13] balancing techniques were applied to address the data balancing problem. Finally, as accuracy alone is insufficient for evaluating churn prediction models on imbalanced data, additional performance metrics including

precision, recall, and specificity were employed to ensure a reliable evaluation.

Accordingly, the main contributions of this study are summarized as follows: 1) Extracting emotion-driven and behavioral engagement features that reflect customer experience using telecom usage data, 2) demonstrating the effects of combining the newly generated features with the original telecom features on the classifiers' performance. To achieve this, a structured experimental design is conducted, which compares original telecom data, enhanced telecom data (original + emotion-driven and behavioral engagement features), and the newly generated features only in telecom churn classification tasks.

The remainder of this study is organized as follows: Section II reviews related work on telecom customer churn prediction, with particular focus on machine learning approaches and feature engineering techniques. Section III describes the dataset used in this study and outlines its main characteristics. Section IV presents the proposed methodology, including the feature engineering process, the classifiers employed, the data balancing techniques, and the evaluation metrics. Section V reports and analyzes the experimental results obtained from the three-stage evaluation framework. Finally, Section VI concludes the study by summarizing the main findings and discussing limitations and directions for future research.

II. Related Works

Several researchers have addressed customer churn prediction in the telecommunications sector using machine learning and deep learning models. In this context, Wagh et al. [3] developed a machine learning-based system to predict customer churn in the telecom sector using Random Forest, K-Nearest Neighbors, and Decision Tree, relying on the original dataset to train and test classifiers. Furthermore, Gaur and Dubey [4] employed four machine learning algorithms, including Logistic Regression, SVM, Random Forest, and Gradient Boosted Trees, to build and compare effective churn prediction systems. Similarly, Poudel et al. [5] demonstrated that telecom customer churn can be effectively predicted using a Gradient Boosting Machine (GBM) enhanced by explainer models and visualization tools for deeper insights. Moreover, Lalwani et al. [6] proposed a six-phase methodology for telecom churn prediction using machine learning models (i.e., Logistic Regression, Support Vector Machine, Random Forest) and boosting techniques.

However, a limited number of studies have focused on feature selection and feature transformation, which can reveal deeper patterns related to dissatisfaction or reduced service quality. For instance, Huang et al. [14] proposed a feature-selection approach that prioritizes churn-related attributes in an imbalanced telecom dataset. Their method outperformed conventional selection techniques such as Chi-Square and Information Gain, showing that carefully identifying discriminative features can significantly enhance the prediction performance. Moreover, Sana et al. [15] demonstrated that combining univariate feature selection with several transformation methods produces more informative feature representations. Their approach improved the prediction accuracy of telecom churn.

More recent studies have focused on hybrid feature-selection strategies. For example, Mozaffari et al. [16] combined several feature selection methods and aggregated their outputs to identify the most relevant and reliable features. The results show that this hybrid feature selection significantly improves churn prediction performance.

Therefore, these studies highlight the effectiveness of feature engineering in improving churn prediction, but they mainly transformed existing dataset attributes or removed weak ones. None explicitly attempted to derive advanced emotion-related features that capture customer frustration, satisfaction, trust, usage sentiment, or service engagement. Therefore, the literature lacks structured attempts to quantify psychological or experience-based patterns that could affect churn decision-making. Addressing this limitation, the present study introduces newly engineered emotion-driven and behavioral engagement features and evaluates their effect on telecom churn prediction performance.

III. Dataset Description

The dataset used in this study was obtained from the Kaggle community. It consists of 3,333 customer records and 20 attributes, 16 of which are numerical and 4 are categorical. The dataset is imbalanced, with 2,850 non-churner customers and 483 churners. In addition, three of the categorical attributes are binary (Yes/No), which were converted into a binary indicator (0/1), while the remaining categorical attribute (i.e., *State*) was excluded from the analysis due to its high cardinality and limited contribution to predictive performance. Furthermore, the data were divided into stratified training and testing sets, and the class balancing problem was addressed using SMOTE and SMOTE-Tomek, applied only to the training data. All engineered features were derived exclusively from the original input variables to prevent data leakage. However, more details about the dataset used in this study are available in: https://www.kaggle.com/datasets/mnassrib/telecom-churn-datasets, and no missing values were observed.

IV. Methodology

In this work, we extracted five engineered features from the original dataset: three emotion-driven features (frustration index, trust score, and satisfaction index) and two behavioral engagement features (service usage score and international experience index). The extracted features were then inserted into the original dataset, and were used to train and test three powerful classifiers (i.e., CatBoost, RF, and XGBoost). The dataset is highly imbalanced; thus, two advanced balancing techniques were used to handle this issue (i.e., SMOTE, and SMOTE-Tomek). Moreover, four metrics were used to evaluate the performance of classifiers in predicting churn (i.e., accuracy, precision, recall, and specificity). On the other hand, to observe the impact of adding the emotion-driven and behavioral engagement features to the original dataset on the classifiers' performance, the original dataset (with no emotion-driven and behavioral engagement features) was used to train and test classifiers. Furthermore, to obtain a broader view on the impact of the emotion-driven features on the classification performance, the emotion-driven and behavioral engagement features were used exclusively to train classifiers.

Fig. 1 illustrates the overall research framework adopted in this study.

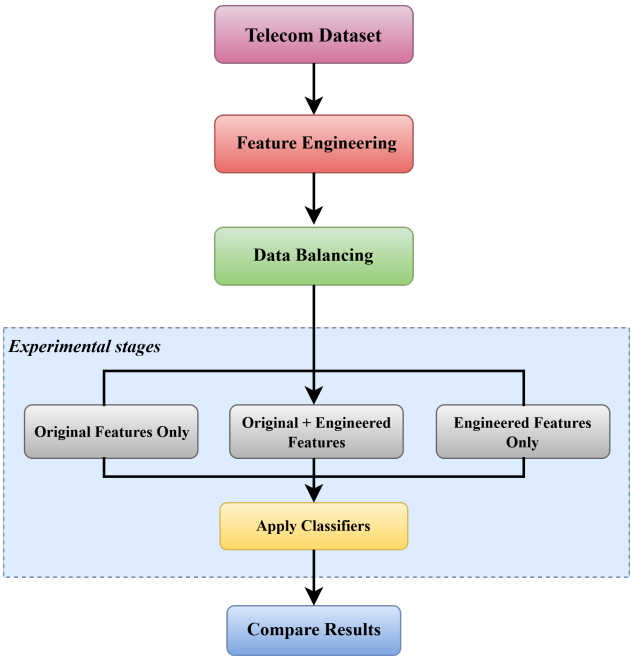


Fig. 1. Overall research framework of the proposed telecom churn prediction methodology.

A. Emotion-Driven and Behavioral Engagement Feature Engineering

To enrich the telecom churn dataset, a set of emotion-driven and behavioral engagement features was extracted from existing attributes and added to the original dataset. These features are grouped into two categories:

- Emotion-driven features: *frustration index*, *satisfaction index*, and *trust score*.
- Behavioral engagement features: *service usage score*, and *international experience index*.

The following list explains the extraction process for each newly derived feature.

1) Frustration index: Customer frustration is extracted from the Customer Service Calls feature, which measures how many times a subscriber contacted support. Therefore, frequent calls indicate unresolved problems or dissatisfaction with service quality.

$$Frustration_c = \frac{\min(CustomerServiceCalls_c, 10)}{10} \quad (1)$$

where, $CustomerServiceCalls_c$ is the number of service calls made by customer c. The $min(x, 10)$ function limits the number of calls to 10 to prevent excessive values from affecting the results. The division by 10 rescales the values; 0 corresponds to no calls, whereas 1 corresponds to 10 or more calls.

2) Trust score: Trust is derived from the Account length feature, which represents the number of subscription days. A longer subscription period indicates stronger loyalty and confidence in the provider. To measure Trust score, a min/max normalization was applied.

$$Trust_c = \frac{AccountLength_c - \min(AccountLength)}{\max(AccountLength) - \min(AccountLength)} \quad (2)$$

In this formula, $AccountLength_c$ presents the subscription duration of customer c. The minimum and maximum values of $AccountLength$ represent the shortest and longest subscription periods found in the dataset.

3) Satisfaction index: Satisfaction is extracted from the efficiency of daytime service usage, which is expressed as the ratio of *Total day minutes* to *Total day charge* features. Customers who receive more minutes per unit cost are assumed to have higher satisfaction. The satisfaction index is then scaled between 1 and 5.

$$Satisfaction_c = 1 + 4 \times \frac{\frac{TotalDayMinutes_c}{TotalDayCharge_c + \varepsilon}}{\max\left(\frac{TotalDayMinutes}{TotalDayCharge + \varepsilon}\right)} \quad (3)$$

The formula includes $TotalDayMinutes_c$ which represents daytime minutes used by customer c, and $TotalDayCharge_c$ which represents the corresponding cost. Also, the formula includes a small constant ε to avoid division by zero. The denominator represents the highest efficiency ratio found in the dataset, which ensures that the customer with the highest efficiency gets a satisfaction score of 5, while all remaining customers are proportionally scaled between 1 and 5.

4) Service usage score: The calculation of off-peak hours billing fairness through balance ratio requires the integration of evening and night usage data with their respective charges according to the following formula:

$$Balance_c = \frac{TotalEveMinutes_c + TotalNightMinutes_c}{TotalEveCharge_c + TotalNightCharge_c + \varepsilon} \quad (4)$$

For customer c, $TotalEveMinutes_c$ and $TotalNightMinutes_c$ show the total evening and night call times, while $TotalEveCharge_c$ and $TotalNightCharge_c$ represent the corresponding charges. A tiny value of ε was added to avoid division by zero. This balance ratio was used to calculate the $ServiceUsageScore_c$, as shown in the following formula:

$$ServiceUsageScore_c = \frac{Balance_c - \min(Balance)}{\max(Balance) - \min(Balance)} \quad (5)$$

where, $Balance_c$ represents the ratio of off-peak usage to charge for customer c, while $min(Balance)$ and

$max(Balance)$ represent the minimum and maximum balance ratios found in the dataset. The scores increase with better off-peak efficiency, indicating improved billing performance.

5) International experience index: Satisfaction with international services was represented by the average duration of international calls. This was calculated for each customer and then normalized across the dataset, as described in the following formulas:

$$IntlExp_c = \frac{TotalIntlMinutes_c}{TotalIntlCalls_c + \varepsilon} \tag{6}$$

$$IntlExperienceIndex_c = \frac{IntlExp_c - \min(IntlExp)}{\max(IntlExp) - \min(IntlExp)} \tag{7}$$

Here, $IntlExp_c$ represents the average international call duration for the customer c. The total international call duration for customer c is represented by $TotalIntlMinutes_c$. Whereas the total number of international calls made by customer c is denoted as $TotalIntlCalls_c$. The fraction computes the average international call length, with ε included to avoid division by zero. The minimum and maximum observed averages, $min(IntlExp)$ and $max(IntlExp)$, are then used to rescale the measure into the [0,1] interval. Higher values reflect better international call experiences.

B. Classifiers Employed

1) Categorical boosting (CatBoost): It is a powerful classification algorithm that builds ensembles of binary decision trees to perform gradient boosting operations. It introduces a technique called ordered boosting, where the model is trained on specific permutations of the dataset. For each data point, CatBoost computes predictions and target statistics using only the preceding examples within the permutation. This prevents target leakage, leading to better generalization and reduced bias [10].

2) eXtreme gradient boosting (XGBoost): It is an efficient and scalable machine learning algorithm based on the gradient boosting framework. Its procedure is based on building an ensemble of decision trees sequentially, where each new tree attempts to correct the errors made by the previous ones. Furthermore, the addition of regularization techniques in XGBoost enhances its performance compared to conventional boosting algorithms [11]. Additionally, it provides parallel processing capabilities, internal handling of missing data and optimized data structures which enhance speed and accuracy in different classification tasks [17].

3) Random forest (RF): It is a supervised machine learning algorithm that generates decision trees and combines their outputs to improve the accuracy and reliability of the model. Each tree is trained on a randomly selected subset of the dataset, and at each decision point within the tree, only a random subset of features is considered. The random selection process creates different trees which work together to prevent overfitting and improve predictions for new data points [9]. Additionally, in classification, it selects the most common prediction among all trees, while in regression, it calculates the average of their outputs [18].

However, hyperparameter tuning was not performed in this study. All classifiers were implemented using their default parameter settings as provided in their respective libraries to ensure a fair comparison across the three experimental stages.

C. Balancing Techniques Used

1) Synthetic Minority Over-sampling Technique (SMOTE): It is an advanced oversampling technique that generates new synthetic samples through interpolation between minority class instances and their *k*-nearest neighbors instead of replicating the existing minority class samples. This is done by selecting a random point on the line segment connecting the two feature vectors. Consequently, this technique effectively expands the decision space for the minority class and reduces the risk of overfitting [12].

2) Synthetic Minority Over-sampling Technique (SMOTE) with Tomek Links (SMOTE-Tomek): SMOTE-Tomek is a hybrid resampling technique that combines SMOTE with Tomek Links to better handle class balancing issue and enhance classification performance. As stated before, SMOTE generates synthetic samples by interpolating between minority class instances and their nearest neighbors, thereby increasing the representation of the minority class. On the other hand, Tomek Links identify pairs of instances belonging to different classes that are each other's closest neighbors; these pairs often occur near class boundaries and are likely to be noisy. Removing such pairs produces a cleaner dataset with clearer class separation, which helps classifiers achieve more robust performance [13].

D. The Evaluation Metrics Considered

The performance of the classifiers was evaluated using four metrics: accuracy, precision, recall, and specificity [19]. These metrics provide a comprehensive evaluation of model performance, particularly for imbalanced classification problems.

- Accuracy: measures the overall proportion of correctly classified instances in both classes; churn and non-churn.
- Recall: measures the proportion of actual minority class instances (churn) that are correctly identified by the classifier.
- Specificity: measures the proportion of majority class instances (non-churn) that are correctly identified.
- Precision: identifies the proportion of predicted minority class instances (churn) that are truly positive, indicating the reliability of positive predictions.

V. Results

This section presents the experimental findings obtained in three stages of analysis. The first experiment evaluates classifier performance using the original telecom dataset with traditional features only. The second experiment introduces emotion-driven and behavioral engagement features alongside the original attributes to assess their added value. Finally, the third experiment examines the predictive power of emotion-driven and behavioral engagement features in isolation. The results are compared across multiple classifiers and balancing

TABLE I. EVALUATION METRICS OBTAINED FROM THE COMBINATIONS OF CLASSIFIERS AND BALANCING TECHNIQUES ON THE ORIGINAL DATASET ONLY.

Model	Accuracy	Precision	Recall	Specificity
CatBoost + SMOTE	0.9272	0.9620	0.8895	0.9649
CatBoost + SMOTE-Tomek	0.9488	0.9686	0.9276	0.9700
RF + SMOTE	0.9395	0.9700	0.9070	0.9719
RF + SMOTE-Tomek	0.9559	0.9657	0.9452	0.9665
XGBoost + SMOTE	0.9421	0.9755	0.9070	0.9772
XGBoost + SMOTE-Tomek	0.9559	0.9674	0.9435	0.9683

techniques to highlight the role of the newly generated features in enhancing churn prediction.

A. Results of the First Experiment: Original Dataset Only

In the first part of the experiments, customer churn prediction was performed using CatBoost, RF, and XGBoost on the original telecom dataset without the inclusion of the emotion-driven and behavioral engagement features. To address class imbalance, two resampling methods, SMOTE and SMOTE-Tomek, were applied. The performance of the classifiers was evaluated using accuracy, precision, recall, and specificity metrics. As is common in similar works [20], we compare the performance of classifiers to determine the best performer.

As shown in Table I, all classifiers yielded strong performance in predicting customer churn, and SMOTE-Tomek consistently outperformed SMOTE. The performance of the combinations of CatBoost , RF, and XGBoost with SMOTE-Tomek performs better than the combinations of the same classifiers with SMOTE in predicting the majority class and the minority class; the combinations with SMOTE-Tomek obtained higher recall and specificity. With respect to precision, the combinations of RF and XGBoost with SMOTE obtained slightly higher values compared to SMOTE-Tomek. Accordingly, these results indicate that all three classifiers are effective in predicting customer churn in telecom. However, SMOTE-Tomek obtained more balanced outcomes as stated in recall and specificity. This confirms the importance of advanced balancing techniques in improving classifier performance, particularly in predicting the minority class (i.e., churn).

B. Results of the Second Experiment: Incorporating Emotion-Driven and Behavioral Engagement Features

In the second part of experiments, three emotion-driven features (i.e., *frustration index*, *trust score*, *satisfaction index*), and two behavioral engagement features (i.e., *service usage score*, and *international experience index*) were derived from the original dataset and added as new features. As in the first part of experiments, SMOTE and SMOTE-Tomek were employed to solve class balancing problem, and the classifiers' performance was evaluated using accuracy, precision, recall, and specificity metrics.

The results presented in Table II, show that the integration of emotion-driven and behavioral engagement features with the original dataset improved the performance of all classifiers compared to the previous experiment. However, in this part of the experiments all of the combinations of classifiers with

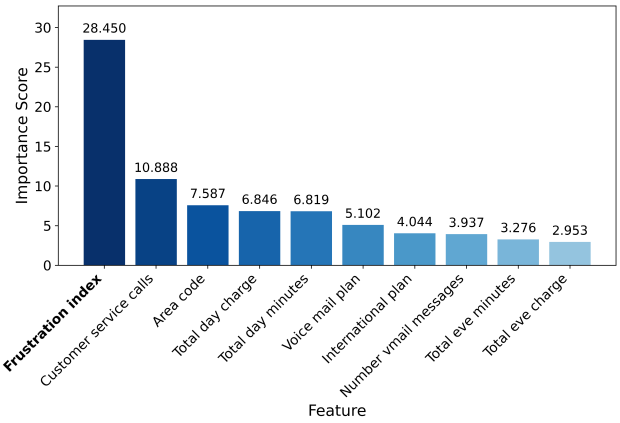


Fig. 2. Ten most important features in the classification process using CatBoost combined with SMOTE.

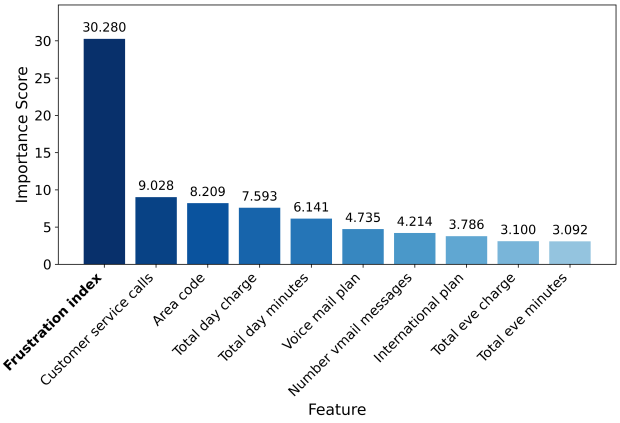


Fig. 3. Ten most important features in the classification process using CatBoost combined with SMOTE-Tomek.

SMOTE and SMOTE-Tomek balancing techniques obtained a significant accuracy value; all of them obtained more than 0.96, except the combination of RF with SMOTE, it obtained slightly lower value. In addition, the combination of CatBoost with SMOTE-Tomek showed superior performance in predicting the minority class (i.e., churn); it obtained the highest recall value. On the other hand, the combination of XGBoost with SMOTE showed the best performance in predicting the majority class (i.e., non-churn), as stated in the specificity metric values.

TABLE II. EVALUATION METRICS OBTAINED FROM THE COMBINATIONS OF CLASSIFIERS AND BALANCING TECHNIQUES APPLIED ON THE ORIGINAL DATASET + EMOTION-DRIVEN AND BEHAVIORAL ENGAGEMENT FEATURES

Model	Accuracy	Precision	Recall	Specificity
CatBoost + SMOTE	0.9632	0.9783	0.9474	0.9789
CatBoost + SMOTE-Tomek	0.9673	0.9783	0.9558	0.9788
RF + SMOTE	0.9570	0.9745	0.9386	0.9754
RF + SMOTE-Tomek	0.9629	0.9712	0.9541	0.9718
XGBoost + SMOTE	0.9675	0.9819	0.9526	0.9825
XGBoost + SMOTE-Tomek	0.9656	0.9765	0.9541	0.9771

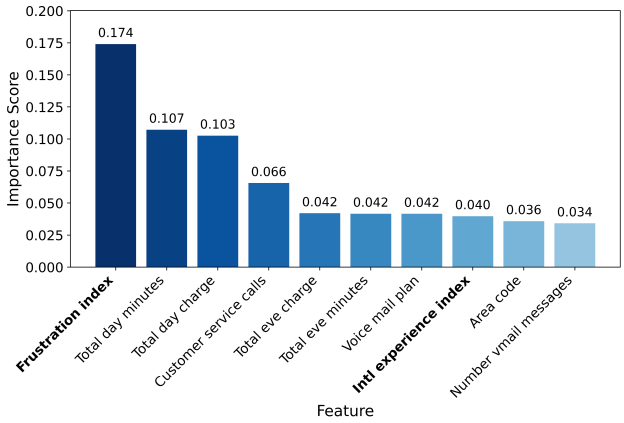


Fig. 4. Ten most important features in the classification process using RF combined with SMOTE.

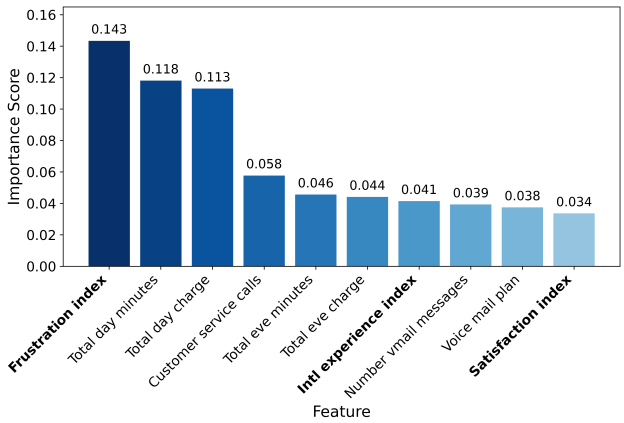


Fig. 5. Ten most important features in the classification process using RF combined with SMOTE-Tomek.

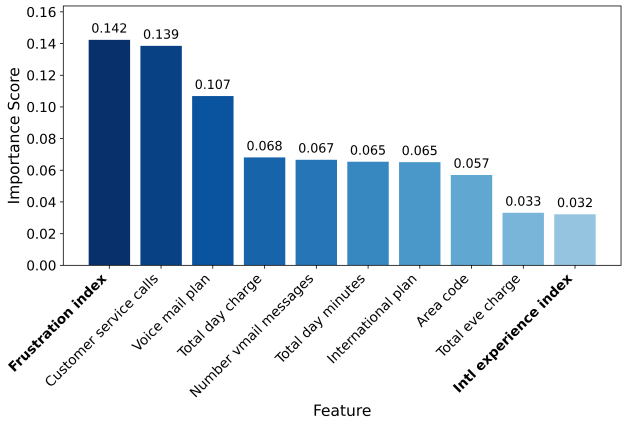


Fig. 6. Ten most important features in the classification process using XGBoost combined with SMOTE.

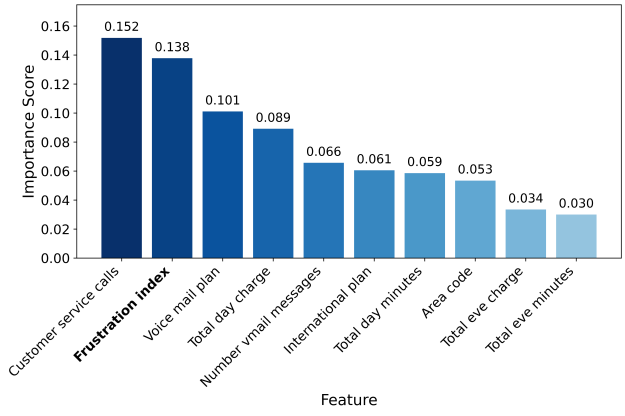


Fig. 7. Ten most important features in the classification process using XGBoost combined with SMOTE-Tomek.

Furthermore, feature importance analysis demonstrates how new emotion-driven and behavioral engagement features affect classifiers' performance. As shown in Fig. 2 to Fig. 7, the frustration index was the most important feature for all of the classifiers during the classification process, surpassing all of the original dataset's features, except with the combination of XGBoost with SMOTE, it was the second most important. The Satisfaction index and International experience index appeared in the top feature rankings obtained from RF + SMOTE, RF + SMOTE-Tomek, and XGBoost + SMOTE. Consequently, the repeated appearance of multiple emotional features in the top-ranked features for classifiers demonstrates their ability to improve the performance of classifiers in predicting customer churn.

A firm summary constructed from the second part of the experiments is that adding emotion-driven and behavioral engagement features consistently improves model performance in predicting both classes (i.e., churn and non-churn). This result is consistent with prior research indicating that customer experience and perceived service interactions play a critical role in shaping customer retention behavior and future decisions [21], [22].

TABLE III. EVALUATION METRICS OBTAINED FROM THE COMBINATIONS OF CLASSIFIERS AND BALANCING TECHNIQUES APPLIED ON THE EMOTION-DRIVEN AND BEHAVIORAL ENGAGEMENT FEATURES ONLY

Model	Accuracy	Precision	Recall	Specificity
CatBoost + SMOTE	0.8202	0.8163	0.8263	0.8140
CatBoost + SMOTE-Tomek	0.8444	0.8450	0.8435	0.8453
RF + SMOTE	0.8728	0.8571	0.8947	0.8509
RF + SMOTE-Tomek	0.8803	0.8707	0.8932	0.8674
XGBoost + SMOTE	0.8263	0.8142	0.8456	0.8070
XGBoost + SMOTE-Tomek	0.8508	0.8547	0.8453	0.8564

C. Results of the Third Experiment: Emotion-Driven and Behavioral Engagement Features Only

In the third part of the experiments, only five newly generated features were used to train and test classifiers. In addition, SMOTE and SMOTE-Tomek were adopted to solve the data balancing problem. Table III shows the results obtained by the combinations of classifiers and balancing techniques on the emotion-driven and behavioral engagement features.

Regarding the evaluation metrics values, the combinations of RF with SMOTE and SMOTE-Tomek outperformed all other combinations; they obtained the highest values for all evaluation metrics. On the other hand, the other combinations of classifiers with balancing techniques obtained considerable results with respect to predicting customer churn using the emotion-driven and behavioral engagement features only. Accordingly, the results obtained in this part of experiments proved that using the new features only to train and test classifiers to predict churn is not sufficient to achieve the same high levels of accuracy, recall, and specificity as when combined with original telecom dataset features. But it can still obtain relatively high values for the evaluation metrics.

D. Comparative Summary of the Three Experimental Stages

The three parts of the experiments gradually demonstrated the impact of inserting the emotion-driven and behavioral engagement features to the original dataset on the performance of classifiers in predicting churn. In the first part of experiments, only the original dataset's features were used to train and test the classifiers to predict churn. Thus, all classifiers showed strong performance, particularly when combined with SMOTE-Tomek. The second experiment introduced the emotion-driven and behavioral engagement features into the dataset, and yielded the highest performance across all models compared to the first part of experiments. In other words, there is an improvement in the performance of all classifiers and balancing techniques combinations regarding predicting both classes. All of the combinations trained and tested on the original dataset + emotion-driven and behavioral engagement features obtained better evaluation metrics values compared to the original dataset only. Moreover, feature importance analysis revealed that emotion-driven features, particularly, frustration index and satisfaction index, consistently ranked among the top ten important features in the classification process. This indicates the effectiveness of the emotion-driven features in predicting customer churn. Furthermore, the third part of experiments relied exclusively on the engineered features, and resulted in lower accuracy, recall, and specificity compared to the first and second parts of experiments. This suggests that while emotion-driven features provide significant improvement in the classifiers' performance, they are most effective when integrated with original telecom dataset features rather than used in isolation. Accordingly, the three parts of experiments confirm that the integration of emotion-driven features offers consistent added value for churn prediction.

VI. CONCLUSION

This study demonstrates that integrating standard (original) telecom features with five emotion-driven and behavioral engagement features, namely: frustration index, trust score, satisfaction index, service usage score, and international experience index, consistently improves churn prediction performance. Moreover, in three stages of evaluation, the results show that combining the telecom churn data with the proposed features yields better predictive outcomes than relying on the original telecom attributes only.

The experimental findings further indicate that emotion-driven and behavioral engagement features provide meaningful information that complements customer usage features (original data features), particularly in improving the prediction of customers at risk of churn. The consistent performance enhancement observed across Random Forest, CatBoost, and XGBoost indicates that the proposed features are effective across different classifiers, rather than being customized to a single classifier. In addition, SMOTE and SMOTE-Tomek proved their efficiency in addressing the data balancing problem, ensuring a reliable evaluation of churn prediction performance using multiple metrics (i.e., accuracy, precision, recall, and specificity).

From a practical perspective, the proposed features provide clear indicators of customer experience and engagement using existing operational data. Those indicators can be integrated into current churn prediction systems to improve performance and support retention strategies, without replacing existing prediction models.

Future research may extend this work by extracting similar features from richer data sources, such as customer surveys, textual interactions, or real-time service logs, and by evaluating their impact on deep learning–based churn prediction models. In addition, future research may analyze model explanations at both individual customer and customer segment levels to support more targeted and actionable retention strategies.

Declaration on the Use of Generative AI

The author declares that generative artificial intelligence (AI) tools were used solely for language refinement and grammatical improvement during the preparation of this manuscript. All scientific ideas, methodological design, feature engineering, experimental implementation, data analysis, interpretation of results, and conclusions were entirely conceived and developed by the author. The author has critically reviewed and edited all AI-assisted content and takes full responsibility for the accuracy, originality, and integrity of the work.

References

[1] A. Omari, O. Al-Omari, T. Al-Omari, and S. M. Fati, "A predictive analytics approach to improve telecom's customer retention," *Frontiers in Artificial Intelligence*, vol. 8, p. 1600357, 2025.

[2] L. R. N. Deiranieh, E. A. Alahmad, M. M. Qadri, A. Abdallah, F. Omeish, N. N. Ali, N. S. Al-Akash, and A. Ahmad, "Factors influencing switching behavior of customers in the telecommunication industry of jordan: The mediating role of csr," *Tec Empresarial*, vol. 19, no. 1, pp. 45–74, 2024.

[3] S. K. Wagh, A. A. Andhale, K. S. Wagh, J. R. Pansare, S. P. Ambadekar, and S. Gawande, "Customer churn prediction in telecom sector using machine learning techniques," *Results in Control and Optimization*, vol. 14, p. 100342, 2024.

[4] A. Gaur and R. Dubey, "Predicting customer churn prediction in telecom sector using various machine learning techniques," in *2018 International Conference on Advanced Computation and Telecommunication (ICACAT)*. IEEE, 2018, pp. 1–5.

[5] S. S. Poudel, S. Pokharel, and M. Timilsina, "Explaining customer churn prediction in telecom industry using tabular machine learning models," *Machine Learning with Applications*, vol. 17, p. 100567, 2024.

[6] P. Lalwani, M. K. Mishra, J. S. Chadha, and P. Sethi, "Customer churn prediction system: a machine learning approach," *Computing*, vol. 104, no. 2, pp. 271–294, 2022.

[7] H. Aljawazneh and F. A. Al-Ragheb, *The Effect of Neural Networks Depth on Forecasting Performance: A Study on Customer Churn Prediction in Telecom Industry*. Cham: Springer Nature Switzerland, 2025, pp. 111–123.

[8] C. Wang, C. Rao, F. Hu, X. Xiao, and M. Goh, "Risk assessment of customer churn in telco using fclcnn-lstm model," *Expert Systems with Applications*, vol. 248, p. 123352, 2024.

[9] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, October 2001.

[10] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: unbiased boosting with categorical features," *Advances in neural information processing systems*, vol. 31, 2018.

[11] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[13] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 20–29, 2004.

[14] Y. Huang, B. Huang, and M. Kechadi, "A new filter feature selection approach for customer churn prediction in telecommunications," in *2010 IEEE International Conference on Industrial Engineering and Engineering Management*. IEEE, 2010, pp. 338–342.

[15] J. K. Sana, M. Z. Abedin, M. S. Rahman, and M. S. Rahman, "A novel customer churn prediction model for the telecommunication industry using data transformation methods and feature selection," *Plos one*, vol. 17, no. 12, p. e0278095, 2022.

[16] F. Mozaffari, I. R. Vanani, P. Mahmoudian, B. Sohrabi *et al.*, "Application of machine learning in the telecommunications industry: Partial churn prediction by using a hybrid feature selection approach," *Journal of Information Systems and Telecommunication*, 2023.

[17] S. Ramraj, N. Uzir, R. Sunil, and S. Banerjee, "Experimenting xgboost algorithm for prediction and classification of different datasets," *International Journal of Control Theory and Applications*, vol. 9, no. 40, pp. 651–662, 2016.

[18] H. A. Salman, A. Kalakech, and A. Steiti, "Random forest algorithm overview," *Babylonian Journal of Machine Learning*, vol. 2024, pp. 69–79, 2024.

[19] B. J. Erickson and F. Kitamura, "Magician's corner: 9. performance metrics for machine learning models," p. e200126, 2021.

[20] A. Althunibat, R. Amro, B. Hawashin, H. AlNuhait, S. Almanasra, and H. A. Al-Khawaja, "Automated classification of user requirements written in arabic using machine learning algorithms," *Applied Mathematics & Information Sciences*, vol. 17, no. 6, pp. 1155–1170, 2023.

[21] P. C. Verhoef, K. N. Lemon, A. Parasuraman, A. Roggeveen, M. Tsiros, and L. A. Schlesinger, "Customer experience creation: Determinants, dynamics and management strategies," *Journal of Retailing*, vol. 85, no. 1, pp. 31–41, 2009.

[22] K. N. Lemon and P. C. Verhoef, "Understanding customer experience throughout the customer journey," *Journal of Marketing*, vol. 80, no. 6, pp. 69–96, 2016.

Coronary Heart Disease Prediction Using Machine Learning Algorithms

Inooc Rubio Paucar[1], Cesar Yactayo-Arias[2], Laberiano Andrade-Arenas[3]
Facultad de Ingeniería y Negocios, Universidad Privada Norbert Wiener, Lima, Perú[1]
Departamento de Estudios Generales, Universidad Continental, Lima, Perú[2]
Facultad de Ciencias e Ingeniería, Universidad de Ciencias y Humanidades, Lima, Perú[3]

***Abstract*—Cardiopathy is one of the most serious diseases worldwide with its high morbidity and mortality rates posing a latent risk over time. The objective of this research focuses on evaluating Machine Learning (ML) models such as Random Forest (RF), Extreme Gradient Boosting (XGBoost), and Logistic Regression (LR) for the prediction of coronary heart disease (CHD), with the aim of identifying the most efficient model for this prediction. The model construction followed the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, which comprises five stages: business understanding, data understanding, data preparation, modeling, and evaluation. The modeling results revealed the superior predictive capability of the XGBoost algorithm for detecting coronary heart disease, compared to Random Forest and Logistic Regression. The assessment of performance metrics (Accuracy, Precision, Sensitivity, and F1 Score) established XGBoost as the reference model, highlighting an F1 Score of approximately 90.8%. This superiority is attributed to its robustness in capturing nonlinear interactions among clinical variables. Consequently, the XGBoost model is selected as the optimal tool for integration into future medical decision support systems. In summary, this ML-based approach provides a highly predictive tool capable of identifying subtle risk patterns from real clinical data. The XGBoost model is a promising candidate for integration into decision support systems and for the optimization of primary prevention protocols for coronary heart disease.**

***Keywords*—Cardiovascular disease; machine learning; prediction; random forest; XGBoost**

I. Introduction

CHD continues to be one of the leading causes of death worldwide. According to the World Health Organization (WHO), cardiovascular diseases account for approximately 17.9 million deaths annually, representing 32% of all global deaths, and within this group, coronary heart disease constitutes the largest proportion of cases. Similarly, the American Heart Association (AHA) estimates that one in five deaths in the United States is related to coronary events, reflecting the magnitude of the problem in both developed and developing countries [1]. Furthermore, the European Society of Cardiology (ESC) [2] has indicated that despite therapeutic advances, late detection and the lack of accurate predictive tools remain key limitations in reducing the global burden of the disease.

On the other hand, the problem is exacerbated in low- and middle-income regions, where access to early diagnostics and specialized follow-up systems is limited, contributing to a sustained increase in premature mortality and a significant burden of associated disability [3]. Additionally, deficits in hospital infrastructure, the shortage of specialized professionals, and economic barriers to access advanced technologies create a substantial gap compared to high-income countries. In Latin America, reports from the Pan American Health Organization (PAHO) [4] indicate that cardiovascular diseases are responsible for more than 30% of deaths, with a significant proportion attributed to CHD. Added to this is the high economic impact, both from direct healthcare costs and indirect productivity losses, further increasing pressure on health systems [5]. These data underscore the urgent need for innovative strategies and accurate predictive tools to improve early detection and diagnosis in heterogeneous clinical contexts, with the aim of reducing health inequities and optimizing medical decision-making.

In this scenario, ML has emerged as a promising approach in healthcare, offering the ability to analyze large volumes of clinical data and detect nonlinear patterns that traditional statistical methods do not always capture. Recent research has applied algorithms such as LR [6], RF [7], and XGBoost [8] for cardiovascular disease prediction, showing promising results in terms of accuracy, sensitivity, and specificity. However, most of these studies have been conducted on specific populations, with small sample sizes and without external validation, limiting the generalizability of their findings. Furthermore, there is a noticeable scarcity of studies focused on Latin American populations, creating a significant gap in the literature and highlighting the need for contextualized research.

In this context, the present study aims to evaluate and compare the performance of ML algorithms LR, RF, and XGBoost in the prediction of coronary heart disease, with the purpose of identifying the most efficient model and providing evidence that contributes to the development of tools supporting early diagnosis in clinical practice. In this way, the study seeks not only to strengthen the field of medical informatics but also to address the need for solutions applicable to local contexts, with potential impact on reducing cardiovascular mortality and optimizing healthcare resources.

II. Literature Review

This section is organized into two main parts. First, a review of related works is presented, analyzing previous research in the field of cardiovascular disease prediction. This analysis allows for contextualizing the object of study and provides a comprehensive view of the current state of knowledge. Second, the theoretical foundations are addressed, including the main ML algorithms and the clinical aspects associated with CHD, with the purpose of conceptually

supporting the research. The following subsections elaborate on these aspects:

A. Related Work

In recent literature, various authors [9] have explored the potential of ML models in the detection and prediction of complications associated with cardiovascular diseases. In particular, [10] developed an XGBoost-based model aimed at anticipating unplanned one-year readmission in elderly patients with CHD. The study, conducted on a cohort of 2,137 individuals, reported a predictive performance with an Area Under the Receiver Operating Characteristic curve (AUROC) [11] of 0.704 and an AUPRC of 0.392 [12]. Moreover, the most clinically relevant variables were identified as length of hospital stay (LOS), age-adjusted Charlson comorbidity index (ACCI), monocyte count, blood glucose, and red blood cell (RBC) count, highlighting the utility of these parameters as significant predictors in assessing readmission risk. In contrast, another study aimed to develop an ML-based predictive model to estimate frailty risk in elderly patients with CHD [13],[14]. The main predictors identified were Activities of Daily Living (ADL) score [15], hemoglobin [16], low-density lipoprotein cholesterol (LDL) [17], total cholesterol [18], depression, and cardiac function classification [19], which were incorporated into an LR model and four ML algorithms (XGBoost, RF, LightGBM, and AdaBoost). Among them, AdaBoost showed the best performance, with an Area Under the Curve (AUC) of 0.803 in the validation set, demonstrating its robustness as a tool for clinical frailty risk assessment. Consistent with studies highlighting lipoprotein (a) as a cardiovascular risk marker, this study applied a decision tree ML model (Chi-square automatic interaction detection) to a multicenter registry of 2,301 patients with CHD or at high risk. Six clinical clusters were identified based on LDL cholesterol, CHD presence, family history, and age [20],[21]; these clusters were subsequently regrouped into three categories with progressive increases in Lp (a) and statistically significant differences $p < 0.001$. Consequently, the findings indicate that clinical clustering is an effective tool to stratify risk associated with elevated Lp (a) levels [22],[23]. Focusing on individuals aged 10 to 15 years, one study aimed to predict mortality from CHD using ML techniques [24]. The most relevant variables were selected, and based on them, various ML algorithms were trained, including LR [25], Support Vector Machines (SVM) [26], k-Nearest Neighbors (KNN) [27], Extra Trees Classifier (ETC) [28], and RF [29]. To ensure the validity of the results, 10-fold cross-validation was applied. The analysis showed that the KNN algorithm (k = 18) achieved the best performance, reaching an accuracy of 76.05%, an AUC of 79.41%, sensitivity of 81.08%, specificity of 71.28%, and precision of 73.48%.

Depression is also a key factor in the development of CHD, and in this context, ML emerges as an effective predictive tool [30],[31]. Various algorithms, including LR, RF, gradient boosting, SVM, XGBoost, decision trees, KNN, and neural networks, have been applied to improve risk estimation. Consequently, the RF model demonstrated the best performance, with an AUC of 0.987 in training and 0.996 in validation, as well as adequate calibration and an AUC of 0.928 after internal validation, confirming its high predictive capacity for assessing CHD risk in patients with depression. Another relevant aspect in the study of cardiopathy is exploring the feasibility of integrating ML algorithms for myocardial ischemia diagnosis. In a retrospective analysis of images from 206 patients, [32],[33] applied ML models such as XGBoost [34], RF [35], and logistic regression [36], all achieving accuracy levels above 0.80. Results showed that XGBoost performed best, with an accuracy of 0.903, an F1 score of 0.774, and an AUC of 0.931, demonstrating its high diagnostic potential for detecting myocardial ischemia.

A key element in cardiopathy diagnosis is the combination of artificial intelligence expert systems with neural networks, enabling early detection of at-risk patients and optimizing clinical decision-making. Data from the Framingham Heart Study, comprising 78,001 records, were used to construct a medical database. ML models such as RF, Decision Trees, and Neural Networks were implemented [37]. Consequently, the Decision Tree model achieved an accuracy of 90.08%, while the Neural Network reached 84.56%, demonstrating their capacity to predict the progression of coronary heart disease. Another study highlighted that cardiac disease classification improves the efficiency of medical decision support systems, while reducing associated diagnostic costs. Seven ML algorithms were evaluated, including LR, SVM, KNN, RF, decision trees, Naïve Bayes, and Gradient Boosting Classifier [38]. Hyperparameter optimization techniques such as Grid Search, Random Search, and Bayes Search were applied to maximize model performance [39]. Consequently, RF achieved the best performance, with an average accuracy of 92.85%, reaching up to 94.96% when ensemble methods were applied, evidencing its reliability and efficacy for cardiac disease classification in clinical settings.

Authors in studies such as [40],[41] developed ML models for predicting coronary disease in individuals with periodontitis. This study analyzed data from 3,245 patients, randomly dividing the sample into training and validation sets and considering variables such as age, race, history of myocardial infarction, chest pain, use of lipid-lowering medication, and serum levels of uric acid and creatinine. Five ML algorithms were trained: LR, Gradient Boosting Machine (GBM), SVM, KNN, and Classification and Regression Tree (CART), among which the KNN model showed the best performance, achieving an AUC of 0.977. Calibration metrics, Brier score, and decision curve analysis (DCA) confirmed the model's accuracy and clinical applicability, demonstrating the effectiveness of ML in predicting coronary disease in patients with periodontitis. Early detection of CHD has also been addressed using deep learning and ML techniques [42], employing seven classifiers, including KNN, SVM, LR, Convolutional Neural Networks, Gradient Boost, XGBoost, and RF. Class imbalance handling strategies were applied, with particular optimization of the XGBoost model [43], which achieved outstanding performance: 98.50% accuracy, 99.14% precision, 98.29% sensitivity, and 98.71% F1-score, demonstrating high efficacy for early cardiovascular disease identification. Continuing these investigations, the identification of congenital heart disease patients in large hospital databases involved ML analysis of 19,187 patients, of whom 3,784 were confirmed as CHD [44],[45]. Various algorithms, including Gradient Boosting Decision Tree, SVM, and Decision Tree, were evaluated and compared with regularized LR, using area under the precision-recall curve (AUPRC) as the main metric [46]. External validation

with data updated until 2010 showed that the Gradient Boosting Decision Tree model performed best, with an AUPRC of 99.3%, sensitivity of 98.0%, and specificity of 99.7%, demonstrating the ability of ML methods to efficiently automate the identification of patients with complex diseases like CHD. In summary, cardiac disease enables the identification of high-risk patients and guides more effective prevention strategies. Likewise, cardiac disease prediction has been explored using RF-based classifiers to identify the most relevant features associated with the disease [47],[48]. A recent study employed an RF classifier optimized via hyperparameter tuning using a grid search approach. This model was evaluated in terms of accuracy [49], error rate, and recall [50], and compared with a traditional system. Results showed that while the traditional system achieved accuracies between 81.97% and 90.16%, the hyperparameter-tuned model reached higher accuracies, ranging from 84.22% to 96.53%, demonstrating that ML-based methodology allows for more precise prediction of cardiac disease by optimally identifying the most relevant features.

In summary, the authors proposed an innovative approach for cardiac disease prediction using the Grey Wolf Optimization (GWO) algorithm combined with stacked ensemble techniques to enhance diagnostic accuracy [51]. A patient database was used to evaluate model performance compared to traditional methods. The model demonstrated significantly superior performance, achieving 93% accuracy, with strong metrics in precision (91%), sensitivity (95.3%), F1-score (92.9%), a Matthews correlation coefficient of 0.83, and reduced Log Loss of 2.87, confirming it as a more reliable and effective alternative for early cardiac condition diagnosis. Additionally, the study incorporated the Multi-Criteria Decision-Making (MCDM) approach using the Combined Compromise Solution (CoCoSo) technique to comprehensively evaluate different ML models applied to cardiac disease prediction [52]. This procedure identified LR and SVM as the most consistent and effective algorithms across three clinical datasets, demonstrating superior performance compared to other alternatives [53]. Evaluation was based on six performance metrics, confirming that both models offer greater robustness and reliability for constructing predictive systems aimed at early cardiac disease diagnosis. Similarly, identifying the most suitable model for addressing cardiovascular diseases is a significant challenge in healthcare, motivating the application of ensemble methods and deep learning frameworks on a test dataset obtained from the Kaggle platform [54]. Key predictive variables such as age, gender, cholesterol levels, blood pressure, and lifestyle-related factors were considered. The evaluation showed that both ensemble methods and deep learning outperformed traditional models, achieving higher levels of accuracy, sensitivity, and area under the ROC curve (AUC-ROC), highlighting their potential as more effective tools for early cardiovascular disease diagnosis. Despite these advances, most of the reviewed studies present limitations such as small sample sizes, lack of external validation, and limited application in Latin American populations, restricting the generalizability of the models.

B. Theoretical Bases

1) ML algorithms: ML has established itself as a rapidly expanding scientific discipline, driven by the design of advanced algorithms and its ability to uncover hidden relationships within complex datasets. Through the use of mathematical models and statistical approaches, this field enables computational systems to autonomously adjust their performance, emulating human cognitive processes [55],[56]. Its growing importance lies in its potential to contribute significantly to the generation of new knowledge and the strengthening of decision-making processes across multiple application domains [57]. However, building ML models requires the implementation of concepts derived from specialized programming languages, among which Python [58] stands out for its versatility and widespread adoption in the scientific community. Table I presents the most representative algorithms in the field of ML, highlighting their main characteristics according to different analytical criteria. Consequently, its relevance is based on the ability to handle processes and meet needs autonomously, providing value without requiring direct human intervention.

2) Coronary heart disease: CHD is defined as a cardiovascular disease caused, in most cases, by atherosclerosis of the coronary arteries. This pathological process involves the progressive accumulation of lipids, cholesterol, and other deposits on the arterial wall, leading to the narrowing or blockage of blood flow to the myocardium. As a result, the heart receives an insufficient supply of oxygen and nutrients, which can clinically manifest as angina pectoris, arrhythmias, or acute myocardial infarction. It is important to note that coronary heart disease is the leading cause of death in developed countries, and its impact is closely associated with the increase of risk factors such as hypertension [59], obesity [60], and sedentary lifestyle [61]. Furthermore, early detection through tools based on clinical data and artificial intelligence techniques represents a promising approach to improve the diagnosis and prevention of this disease.

Table II presents the percentage distribution by provinces and regions of Peru, ordered from highest to lowest value. It can be observed that the Constitutional Province of Callao tops the list with 18.2%, followed by Metropolitan Lima and the Department of Lima with 16.3% each. In the intermediate range, Lambayeque, Cajamarca, Ica, and Arequipa stand out, with values ranging from 15.1% to 16.0%. Conversely, the regions with the lowest percentages are Ucayali (8.3%), Huancavelica (9.2%), and Junín (9.5%), falling below the reference threshold of 14.2%. This classification allows a clear differentiation between regions with higher concentration and those with lower relative representation.

III. Methodology

A. Definition of the CRISP-DM Methodology

Regarding the definition of the research topic, the CRISP-DM methodology emerges as a highly viable option for its application [63]. This methodological approach, widely accepted in data mining projects, provides a systematic and flexible framework that facilitates problem understanding, data analysis, and the orderly construction of predictive models. Furthermore, several authors highlight that the CRISP-DM methodology constitutes a fundamental tool for structuring data mining projects, as it enables systematic analysis of information and organizes the process into clearly defined

TABLE I. COMPARISON OF ML ALGORITHMS

Algorithm	Description	Advantages	Limitations
Logistic Regression	A statistical model used for binary classification problems, estimating the probability of a categorical outcome.	- Simple and interpretable - Efficient with linearly separable data - Low computational cost	- Limited to linear relationships - Sensitive to multicollinearity - Poor performance with complex data
Random Forest	An ensemble method based on multiple decision trees combined using bagging.	- Handles high-dimensional data - Reduces overfitting compared to single decision trees - Provides feature importance	- Less interpretable than single trees - Computationally intensive with large datasets - May require tuning of hyperparameters
XGBoost	A gradient boosting framework optimized for efficiency and accuracy in classification and regression tasks.	- High predictive performance - Handles missing values well - Scales efficiently to large datasets	- Complex to tune (many hyperparameters) - Higher computational cost than simpler models - Less interpretable than linear models

TABLE II. PERCENTAGE DISTRIBUTION BY PROVINCES AND REGIONS IN PERU [62].

Rank	Province/Region	%	Above/Below 14.2%
1	Constitutional Province of Callao	18.2	Above
2	Metropolitan Lima	16.3	Above
3	Department of Lima	16.3	Above
4	Lambayeque	16.0	Above
5	Cajamarca	15.5	Above
6	Ica	15.5	Above
7	Arequipa	15.1	Above
8	Tumbes	14.6	Above
9	Piura	14.3	Above
10	Moquegua	13.9	Below
11	Tacna	13.6	Below
12	La Libertad	13.4	Below
13	Cusco	11.5	Below
14	San Martín	11.3	Below
15	Áncash	11.2	Below
16	Amazonas	11.1	Below
17	Apurímac	10.7	Below
18	Ayacucho	10.4	Below
19	Pasco	10.3	Below
20	Huánuco	10.3	Below
21	Puno	10.2	Below
22	Madre de Dios	9.9	Below
23	Loreto	9.6	Below
24	Junín	9.5	Below
25	Huancavelica	9.2	Below
26	Ucayali	8.3	Below

phases: Business Understanding [64], Data Understanding [65], Data Preparation, Modeling, and Evaluation. This methodological framework not only provides an orderly and repeatable workflow but also allows each stage to be adapted to the specific characteristics of the data and the study objectives, as illustrated in Fig. 1. In health-related research, its application is particularly valuable, as it ensures process traceability and reinforces the validity of the generated predictive models. On the other hand, Fig. 2 describes a data analysis system for cardiac diseases driven by ML. The core process involves an ETL for enriched data ingestion, creating an interoperable database with clinical variables (such as blood pressure and BMI). The Data Assets manage the reuse of these resources through APIs. Data is stored in a Cardiac Disease Repository (with fragments and enterprise caching) and is subsequently processed and modeled using tools such as Python and R. The result is a trained ML catalog ready to be applied in various business scenarios.

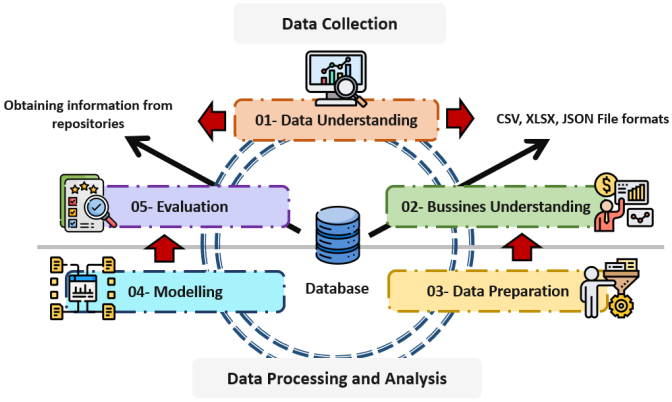


Fig. 1. CRISP-DM methodology.

1) Business understanding: The initialization stage in model development consists of understanding the project's requirements based on the analysis of the problem to be addressed. It focuses on a deep comprehension of the situation and the clear formulation of objectives that will guide the subsequent development of the model. In this regard, Table III presents the Business Understanding phase of the study [66],[67],[68]. The main problem is identified as the high prevalence and late diagnosis of coronary heart disease. The business objective is to support early diagnosis through ML tools, while the project objective is to compare LR, RF, and XGBoost to determine the most efficient model. Success criteria encompass clinical, technical, and social aspects, also considering constraints such as data availability and risks like overfitting. Finally, expected benefits are highlighted, including support for early diagnosis, resource optimization, and reduction of cardiovascular mortality.

2) Data understanding: In this second stage, corresponding to the Data Understanding phase, a systematic process was carried out aimed at the initial exploration and analysis of the available data. First, the dataset was collected in its raw form, providing the necessary clinical and demographic information for the study. On one hand, Table IV defines the activities of the Data Understanding phase of CRISP-DM, which include dataset collection and description, exploration through statistical analyses and visualizations, quality verification to detect inconsistencies, analysis of relationships between variables, and generation of initial hypotheses, applied in this study to the use of clinical and demographic data to

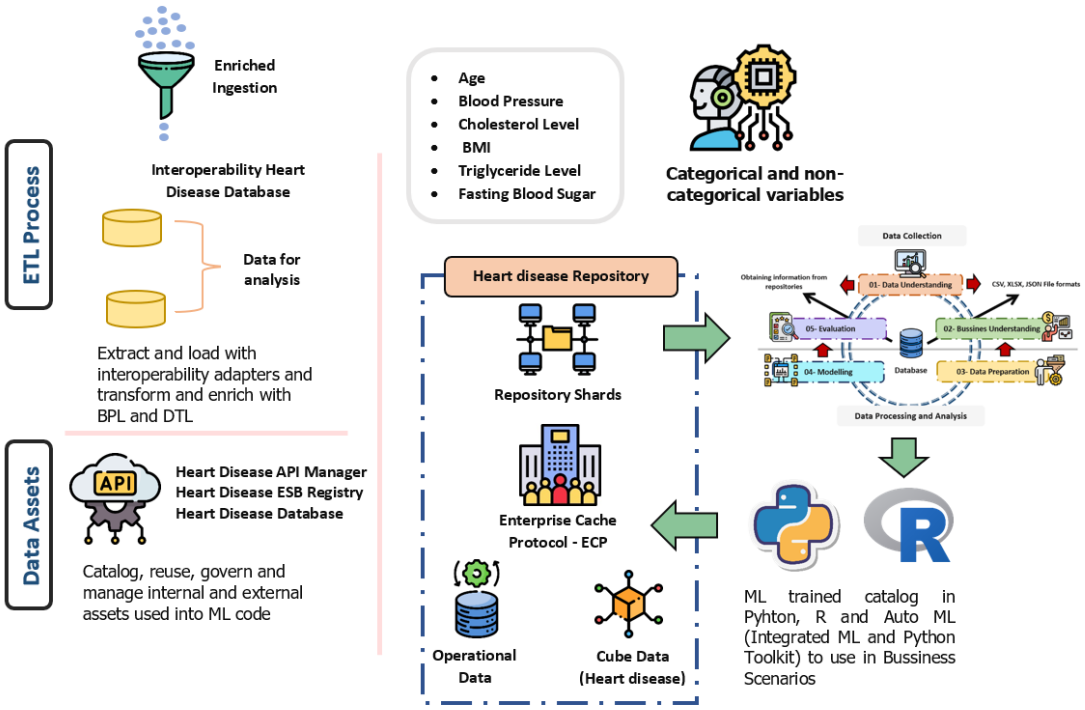


Fig. 2. ML architecture.

TABLE III. BUSINESS UNDERSTANDING PHASE ACCORDING TO CRISP-DM

Element	Description
Business / Clinical Problem	Coronary heart disease is one of the leading causes of mortality worldwide. Its late diagnosis increases the risk of severe complications and places a significant burden on healthcare systems.
Business Objective	To contribute to the early diagnosis of coronary heart disease through decision-support tools based on ML, enabling optimization of healthcare resources and reduction of cardiovascular mortality.
Project Objective	To evaluate and compare the performance of three ML algorithms (LR, RF, and XGBoost) in predicting coronary heart disease, in order to identify the most efficient and clinically applicable model.
Success Criteria	- **Clinical**: support early identification of at-risk patients. - **Technical**: achieve satisfactory performance metrics (accuracy, sensitivity, specificity, AUC). - **Social**: provide evidence applicable to local contexts, with potential impact on reducing cardiovascular mortality.
Constraints	Availability and quality of clinical data, potential imbalance in class distribution, and computational limitations for training complex models.
Risks	Risk of model overfitting, bias in clinical data, and lack of generalization to other populations or healthcare settings.
Expected Benefits	Support for early diagnosis, optimization of hospital resources, reduction of complications and mortality, and contribution of evidence to the field of medical informatics.

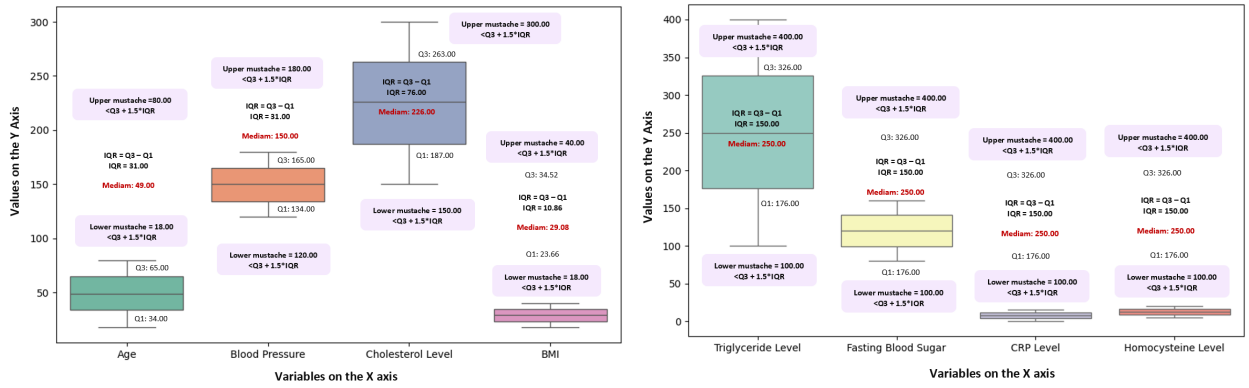
identify risk factors associated with heart disease [69],[70]. On the other hand, Fig. 3 shows box plots summarizing the distribution of various clinical and laboratory variables associated with heart disease. In Fig. 3a, direct risk factors such as age, blood pressure, cholesterol level, and body mass index (BMI) are presented, showing medians, quartiles, and dispersion ranges, with cholesterol and blood pressure exhibiting higher variability. In Fig. 3b, laboratory variables such as triglycerides, fasting glucose, C-reactive protein (CRP), and homocysteine are included, also showing outliers and greater dispersion in triglycerides, while fasting glucose exhibits a more concentrated range. These results allow the identification of central patterns and potential extreme values in the main factors associated with cardiovascular risk.

3) Data preparation: Table V presents the descriptive results of the variables included in the study: Table V(A) includes sociodemographic variables such as age and gender, which allow characterization of the population; Table V(B) covers relevant clinical variables such as blood pressure, cholesterol level, presence of diabetes, and body mass index, which are key factors in predicting heart disease; Table V(C) addresses habits and lifestyle factors, including alcohol and sugar consumption, stress level, and hours of sleep, which directly influence cardiovascular health; and Table V(D) presents the variable of interest, Heart Disease Status [71],[72], along with standardized measures of central tendency and dispersion, facilitating fair comparison across variables.

On the other hand, Table VI is divided into two sections: Table VI(A) General Information of the Dataset and Table VI(B) Missing Values per Column. The table confirms that the dataset contains a total of 21 attributes (20 predictor variables plus the target variable, Heart Disease

TABLE IV. ACTIVITIES OF THE DATA UNDERSTANDING PHASE IN CRISP-DM

Activity	Description	Application in the Study
Initial Data Collection	Obtain the dataset from the selected source and verify its structure.	Download of the dataset from Kaggle with clinical, demographic, and lifestyle variables.
Data Description	Identify the number of records, variables, and data types.	Review of 20 variables (Age, Gender, Blood Pressure, Cholesterol, Diabetes, etc.).
Data Exploration	Analyze relevant variables statistically and through visualizations.	Histograms of age, blood pressure, and cholesterol distribution; correlation plots.
Quality Verification	Detect missing, duplicate, outlier, or inconsistent values.	Identification of null values in clinical variables such as BMI and blood pressure.
Relationship Analysis	Examine links between attributes and the target variable.	Study of correlations between risk factors (cholesterol, diabetes, smoking) and *Heart Disease Status*.
Hypothesis Generation	Formulate initial assumptions to guide modeling.	Hypothesis: elevated cholesterol levels and hypertension are associated with a higher risk of coronary heart disease.



(a) Direct risk factors.

(b) Laboratory variables.

Fig. 3. Measures of central tendency according to dimensions.

Status), including both numerical (float64) and categorical (object) data types. However, the absence of the Statistical Summary section in the image prevents verification of claims regarding the mean age of 49 years, the average blood pressure of 149 mmHg, and cholesterol of 225 mg/dl, as well as the purported lack of calculated statistics for the BMI and Triglycerides variables.

4) Modelling: In this stage, predictive models were built using the data prepared in previous phases. For this purpose, different ML algorithms were selected and trained using the training and validation datasets. Fig. 4 presents the distribution of prediction errors for three classification models. In Fig. 4a, the performance of *RF* is shown, where errors are more concentrated around zero, indicating good predictive capability. In Fig. 4b, corresponding to *LR*, a greater dispersion of errors is observed, reflecting lower accuracy compared to the other methods [73]. Finally, Fig. 4c illustrates the distribution for *XGBoost*, which, like *RF*, keeps most errors close to zero, although with slight asymmetry. In all cases, the shaded areas highlight the most critical errors: false negatives on the left and false positives on the right, making it evident that the tree-based models (a) and (c) provide better control of these errors compared to the linear model (b).

B. Mathematical Formulas in ML Algorithms

The following formulas are commonly used in classical ML algorithms, such as LR, RF, and XGBoost. They formally describe the underlying mechanisms, optimization, and evaluation metrics of these models [74][75].

1) Logistic regression: Probability function: The logistic function models the probability of the positive class $Y = 1$ given predictors X [see Eq. (1)]. It transforms a linear combination of input features into a value between 0 and 1, representing probability.

$$P(Y = 1 \mid X) = \frac{1}{1 + e^{-(\beta_0 + \sum_{i=1}^{n} \beta_i X_i)}} \tag{1}$$

Transforms a linear combination of predictors into a probability between 0 and 1.

2) Logistic regression: Log-loss function: The log-loss function is minimized to estimate the coefficients β [see Eq. (2)]. It penalizes predictions that are confident but incorrect, guiding the optimization of model parameters.

$$\mathcal{L}(\beta) = -\sum_{i=1}^{m} \Big[y_i \log(P(Y_i = 1|X_i)) + (1 - y_i) \log(1 - P(Y_i = 1|X_i)) \Big] \tag{2}$$

Penalizes incorrect predictions according to their probability, guiding parameter estimation.

TABLE V. STATISTICAL SUMMARY OF PROCESSED VARIABLES

(A) DIRECT RISK FACTORS

Variable	count	mean	std	min	25%	50%	75%	max
Age	9971	-0.00	1.00	-1.72	-0.84	-0.02	0.86	1.69
Gender	10000	0.50	0.50	0.00	0.00	1.00	1.00	2.00
Blood Pressure	9981	-0.00	1.00	-1.69	-0.90	0.01	0.87	1.72
Cholesterol Level	9970	0.00	1.00	-1.71	-0.88	0.01	0.86	1.69
Diabetes	10000	0.50	0.51	0.00	0.00	0.00	1.00	2.00
BMI	10000	0.00	1.00	-1.72	-0.86	0.00	0.86	1.69
Smoking	10000	0.50	0.51	0.00	0.00	0.00	1.00	2.00
Family Heart Disease	10000	0.50	0.51	0.00	0.00	0.00	1.00	2.00

(B) LABORATORY VARIABLES

Variable	count	mean	std	min	25%	50%	75%	max
Triglyceride Level	9974	0.00	1.00	-1.72	-0.86	0.00	0.86	1.69
Fasting Blood Sugar	9978	0.00	1.00	-1.68	-0.88	0.01	0.88	1.68
CRP Level	9974	0.00	1.00	-1.72	-0.86	0.00	0.86	1.69
Homocysteine Level	9980	0.00	1.00	-1.72	-0.86	0.00	0.86	1.69

(C) LIFESTYLE VARIABLES

Variable	count	mean	std	min	25%	50%	75%	max
Alcohol Consumption	10000	1.49	1.13	0.00	0.00	1.00	2.25	3.00
Stress Level	10000	1.02	0.82	0.00	0.00	1.00	2.00	3.00
Sleep Hours	9975	0.00	1.00	-1.68	-0.88	0.01	0.88	1.68
Sugar Consumption	10000	1.00	0.82	0.00	0.00	1.00	2.00	3.00

(D) TARGET VARIABLE

Variable	count	mean	std	min	25%	50%	75%	max
Heart Disease Status	10000	0.20	0.40	0.00	0.00	0.00	0.00	1.00

3) Logistic regression: Odds ratio: The odds ratio expresses the effect of each predictor on the likelihood of $Y = 1$ [see Eq. (3)]. It quantifies how a unit change in a predictor multiplies the odds of the positive outcome, providing interpretability of model coefficients.

$$\text{Odds} = \frac{P(Y=1|X)}{1-P(Y=1|X)} = e^{\beta_0+\sum_{i=1}^{n}\beta_i X_i} \tag{3}$$

Represents the multiplicative change in odds for a unit change in each predictor.

4) Logistic regression: Regularized loss function (ridge): Regularization prevents overfitting by penalizing large coefficients [see Eq. (4)]. It adds an $L2$ penalty to the log-loss, encouraging smaller coefficients and improving the generalization ability of the model.

$$\mathcal{L}_{reg}(\beta) = \mathcal{L}(\beta) + \lambda \sum_{i=1}^{n} \beta_i^2 \tag{4}$$

Adds an $L2$ penalty to the log-loss, controlling model complexity.

5) Random forest: Classification prediction: For classification, the final prediction $\hat{y}$ is obtained by majority voting of B trees [see Eq. (5)]. Each tree casts a vote, and the class with the most votes is selected as the final prediction, improving robustness against individual tree errors.

$$\hat{y} = \text{mode}\{h_1(X), h_2(X), \ldots, h_B(X)\} \tag{5}$$

Each tree votes, and the most frequent class is chosen as the final prediction.

6) Random forest: Regression prediction: For regression problems, the prediction is the average of all B trees [see Eq. (6)]. Averaging reduces variance and improves the stability and generalization of the model, making it less sensitive to individual tree fluctuations.

$$\hat{y} = \frac{1}{B} \sum_{b=1}^{B} h_b(X) \tag{6}$$

Reduces variance and improves generalization by averaging multiple models.

7) Random forest: Gini impurity: The Gini index measures the impurity of a node and is used for splitting [see Eq. (7)]. Lower Gini values indicate more homogeneous nodes, which helps the tree create purer child nodes and improve classification accuracy.

$$Gini = 1 - \sum_{k=1}^{K} p_k^2 \tag{7}$$

TABLE VI. SUMMARY AND CHARACTERISTICS OF THE DATASET

(A) GENERAL INFORMATION OF THE DATASET

Column	Non-null values	Data type
Age	9971	float64
Gender	9981	object
Blood Pressure	9981	float64
Cholesterol Level	9970	float64
Exercise Habits	9975	object
Smoking	9975	object
Family Heart Disease	9979	object
Diabetes	9970	object
BMI	9978	float64
High Blood Pressure	9974	object
Low HDL Cholesterol	9975	object
High LDL Cholesterol	9974	object
Alcohol Consumption	7414	object
Stress Level	9978	object
Sleep Hours	9975	float64
Sugar Consumption	9970	object
Triglyceride Level	9974	float64
Fasting Blood Sugar	9978	float64
CRP Level	9974	float64
Homocysteine Level	9980	float64
Heart Disease Status	10000	object

(B) MISSING VALUES PER COLUMN

Column	Missing values
Age	29
Gender	19
Blood Pressure	19
Cholesterol Level	30
Exercise Habits	25
Smoking	25
Family Heart Disease	21
Diabetes	30
BMI	22
High Blood Pressure	26
Low HDL Cholesterol	25
High LDL Cholesterol	26
Alcohol Consumption	2586
Stress Level	22
Sleep Hours	25
Sugar Consumption	30
Triglyceride Level	26
Fasting Blood Sugar	22
CRP Level	26
Homocysteine Level	20
Heart Disease Status	0

Minimized to create more homogeneous child nodes during tree construction.

8) Random forest: Feature importance: Feature importance is calculated based on the total reduction in impurity [see Eq. (8)]. It quantifies how much each feature contributes to improving prediction accuracy, allowing interpretability and identification of key variables.

$$FI_j = \sum_{t=1}^{B} \sum_{\text{splits } s \text{ in tree } t} \Delta i(s) \cdot \mathbf{1}_{\{j \text{ used in } s\}} \tag{8}$$

Summarizes how much each feature contributes to prediction accuracy.

9) XGBoost: Prediction function: The final prediction is the sum of contributions from K sequential trees [see Eq. (9)]. Each tree iteratively corrects the errors of previous trees, allowing the model to capture complex patterns and improve overall predictive performance.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(X_i), \quad f_k \in \mathcal{F} \tag{9}$$

Each tree f_k corrects the errors of previous trees.

10) XGBoost: Regularized objective: XGBoost minimizes a regularized objective combining loss and complexity of trees [see Eq. (10)]. This approach balances predictive accuracy with model complexity, preventing overfitting while ensuring robust performance on unseen data.

$$\mathcal{L}(\phi) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{10}$$

Prevents overfitting while minimizing prediction error.

11) XGBoost: Gradient update: Each iteration updates predictions using the gradient and learning rate η [see Eq. (11)]. This sequential adjustment allows the model to correct previous errors gradually, improving convergence and prediction accuracy while controlling the step size.

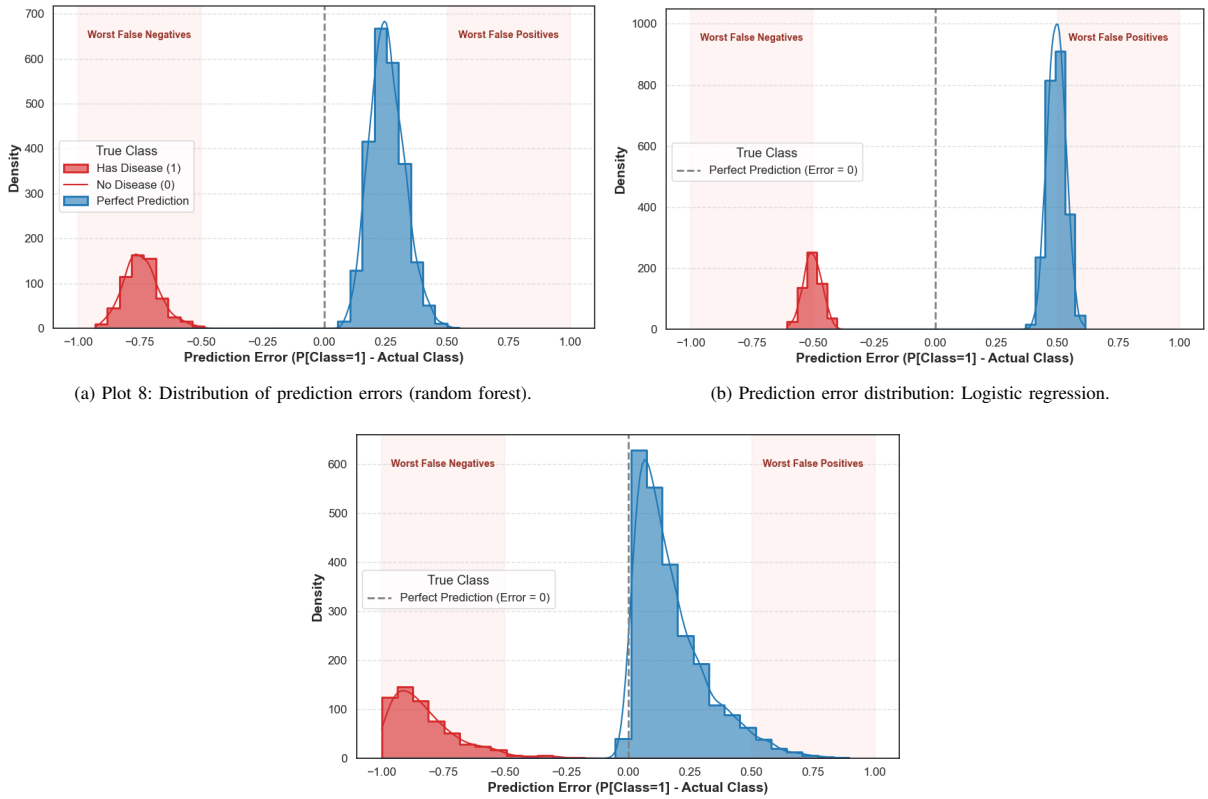
$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta f_t(X_i) \tag{11}$$

Sequentially corrects previous errors, controlling step size via η.

12) XGBoost: Second-order approximation: The second-order Taylor expansion allows efficient tree optimization [see Eq. (12)]. By using both first and second derivatives of the loss function, it provides precise gradient-based updates that accelerate learning and improve model accuracy.

$$\mathcal{L}^{(t)} \approx \sum_{i=1}^{n} \left[g_i f_t(X_i) + \frac{1}{2} h_i f_t^2(X_i) \right] + \Omega(f_t) \tag{12}$$

Uses first (g_i) and second (h_i) derivatives of the loss function for precise updates.



(a) Plot 8: Distribution of prediction errors (random forest).

(b) Prediction error distribution: Logistic regression.

(c) Prediction error distribution: XGBoost.

Fig. 4. Comparative analysis of the impact of class balancing and model diagnostics.

13) XGBoost: Leaf weight optimization: Optimal leaf weights are computed to minimize the regularized loss [see Eq. (13)]. This calculation ensures that each leaf contributes optimally to reducing the objective function while accounting for regularization, enhancing model stability and predictive performance.

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{13}$$

Assigns optimal output values to leaves of the tree considering regularization.

14) XGBoost: Split gain: The gain of a split measures its improvement in the objective function [see Eq. (14)]. Higher gain values indicate more effective splits, guiding the tree to partition data in a way that maximizes predictive accuracy and model efficiency.

$$Gain = \frac{1}{2}\left[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda}\right] - \gamma \tag{14}$$

Quantifies how much a split improves the objective, guiding tree growth.

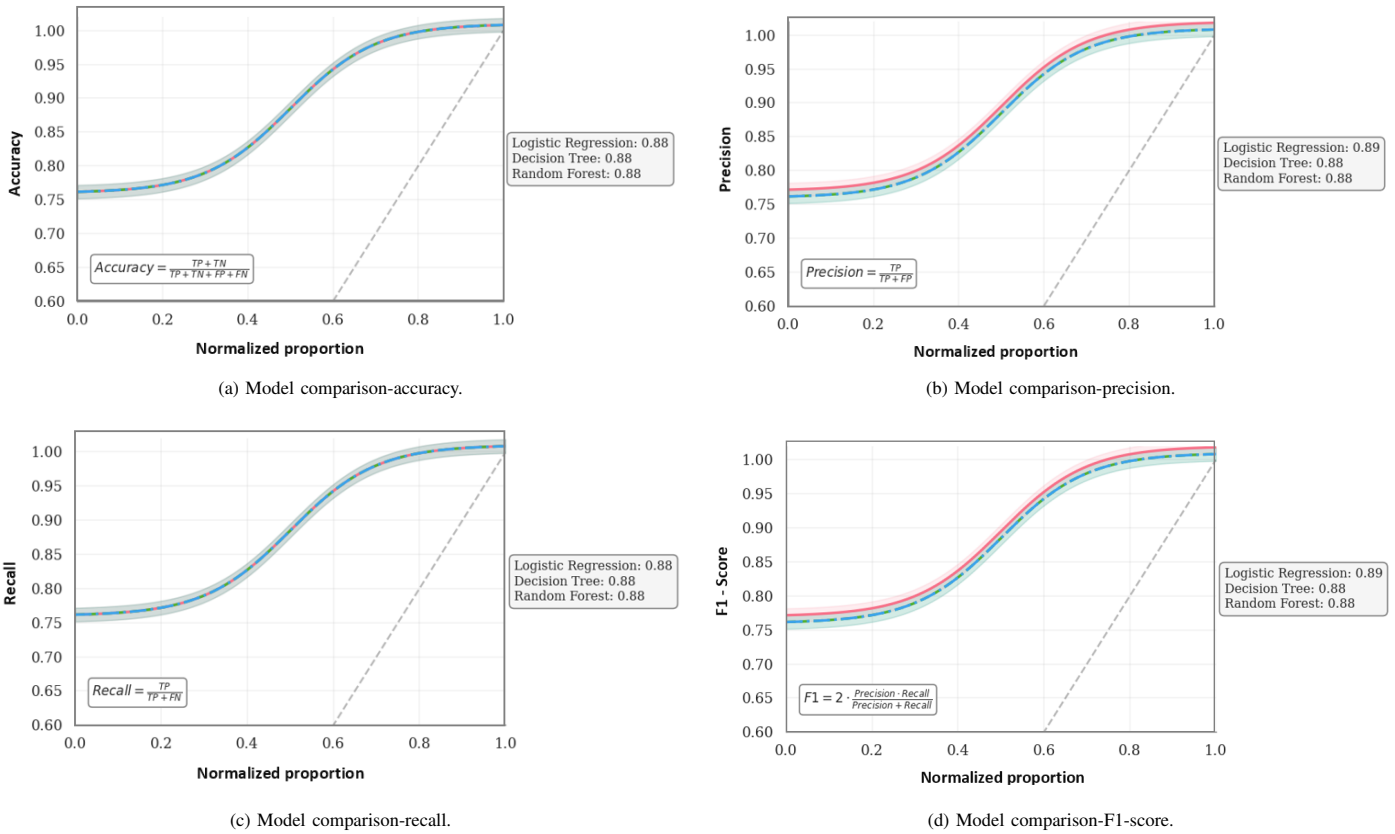
IV. Results

This section presents the results obtained from the comparison of metrics for the algorithms used, aiming to evaluate the performance of each model and select the one that offers the best results. Additionally, a comparison between different ML methodologies is carried out, highlighting their main characteristics and evaluation parameters. From this analysis, it was determined that the most suitable methodology for the development of the present study is CRISP-DM, as it provides a structured and flexible approach that facilitates both the construction and interpretation of the models.

A. Evaluation of Results

This section presents a comparative analysis of the predictive performance of three machine learning architectures: LR, RF, and XGBoost. The models were rigorously evaluated using standard binary classification metrics (Accuracy, Precision, Sensitivity (Recall), and F1-score), with the objective of predicting the presence of coronary artery disease (CAD) based on a standardized set of clinical and health habit features.

The XGBoost model consistently demonstrated superior performance across all evaluation metrics (Fig. 5). Its inherent ability to model nonlinear relationships and complex interactions translated into the best discrimination of the positive class, achieving the highest Accuracy ($\approx 91.2\%$) and the highest F1-score (F1 $\approx$ 90.8%) in Fig. 5d among the



(a) Model comparison-accuracy.

(b) Model comparison-precision.

(c) Model comparison-recall.

(d) Model comparison-F1-score.

Fig. 5. Comparison of ML models by performance metrics.

models. This result underscores the robustness of XGBoost and positions it as the reference model in this study. The RF algorithm exhibited notably competitive performance, maintaining a balanced trade-off between Precision (Fig. 5b) and Sensitivity (Fig. 5c). Its metrics remained very close to the top-performing model (Acc $\approx 90.5\%$, F1 $\approx 90.1\%$) (Fig. 5a), validating the effectiveness of bagging techniques in achieving strong generalization and mitigating the risk of overfitting without sacrificing predictive power. In contrast, the LR model, due to its inherently linear nature, yielded more modest values across all metrics (Acc $\approx 87.8\%$, F1 $\approx 86.9\%$). While LR provides a fundamental interpretative baseline, its performance was predictably outperformed by the tree ensemble models, reiterating its limitations in capturing the inherent complexity of the physiological phenomena underlying coronary disease.

B. Comparison of Methodologies

For the comparison of methodologies, Table VII presents a comparative evaluation of four widely recognized data mining methodologies: CRISP-DM, Knowledge Discovery in Databases (KDD), Sample, Explore, Modify, Model, Assess (SEMMA), and Analytics Solutions Unified Method for Data Mining (ASUM-DM). Each was analyzed under five evaluation criteria: clear structure and phases, practical applicability, integration with machine learning, reproducibility, and interpretability. The results, expressed as percentages, show that the CRISP-DM methodology achieves the highest performance across all evaluated aspects, with an approximate general average of 93%, standing out for its methodological clarity, ability to replicate results, and ease of communicating findings. In contrast, the KDD and SEMMA methodologies exhibit a more technical but less integrative approach, while ASUM-DM maintains a balanced performance albeit with less formal structuring. Overall, the results support the selection of CRISP-DM as the most robust and adaptable methodological framework for studies predicting cardiovascular diseases using machine learning algorithms.

V. Discussion

Previous research has identified multiple causes and characteristics associated with heart diseases within the medical field. In this context, the results obtained in the present study are consistent with trends described in recent literature, where ensemble-based algorithms, particularly XGBoost and Random Forest, have demonstrated superior performance in predicting cardiovascular diseases. In this study, the XGBoost model achieved the highest values across all evaluation metrics (*accuracy, precision, recall* and *F1-score*), confirming its ability to model nonlinear interactions between clinical variables and lifestyle-related factors. These findings align with those reported by [32], who demonstrated that XGBoost provides an optimal balance between sensitivity and specificity in detecting cardiac complications and identifying myocardial ischemia, reaching AUC values above 0.90. Likewise, the RF model showed competitive performance, particularly in the *precision* and *recall* metrics, suggesting adequate

TABLE VII. COMPARATIVE EVALUATION OF METHODOLOGIES BASED ON PERFORMANCE

Evaluation Criteria	Criterion Description	CRISP-DM	KDD	SEMMA	ASUM-DM
Clear Structure and Phases (%)	Assesses the clarity, sequentiality, and organization of methodological stages.	**95**	88	82	90
Practical Applicability (%)	Measures the ease of implementation in real-world data mining projects.	**93**	85	87	89
Integration with ML (%)	Evaluates compatibility with modeling and validation processes for machine learning algorithms.	**90**	80	86	88
Reproducibility (%)	Analyzes the ability to replicate results and maintain consistency across different runs.	**92**	83	80	86
Interpretability (%)	Assesses the transparency of results and ease of communicating findings to different audiences.	**94**	82	78	84
Approximate general average (%)	Average value of overall performance considering the previous criteria.	**93**	84	83	87

generalization capability and lower susceptibility to overfitting. This behavior is consistent with the observations highlighted by [30], [38], and [47], who emphasized the efficiency of Random Forest in classifying heart diseases and its superiority over traditional methods such as LR. These studies revealed that RF achieved precision above 92% and AUC values close to 0.99, results comparable to those obtained in the present work.

In contrast, LR exhibited moderate performance, which was expected given its linear nature and lower ability to capture complex relationships among predictor variables. Nevertheless, its utility lies in providing an interpretable and clinically transparent baseline, an aspect also emphasized by [14] and [53], who acknowledge that despite its limitations, this model retains value in scenarios where interpretability is prioritized over predictive complexity. Moreover, the superiority of XGBoost in this study can also be attributed to its ability to integrate regularization mechanisms and sequential optimization, characteristics that [43] and [20] noted allow maintaining high accuracy even in clinical datasets with certain imbalance or noise. This behavior reinforces the idea that boosting models are more suitable for complex biomedical contexts, where relationships between predictors and outcome variables are not strictly linear.

More broadly, the results of this work confirm observations reported by several authors [51], [52], who reported that combining optimization techniques (such as Grey Wolf Optimization or CoCoSo) with ensemble algorithms significantly increases diagnostic accuracy, outperforming traditional models. Although the present study did not implement metaheuristic strategies, the robustness of the XGBoost model suggests that its incorporation in later stages could further enhance the system's predictive capability. Consequently, the findings align with the general assertion in the literature that ML methods—particularly those based on boosting and bagging—offer a more favorable balance between accuracy, generalization, and stability for the diagnosis and prediction of CHD [39], [19]. Therefore, the XGBoost model consolidates as the most suitable alternative for future external validation and integration into medical decision support systems, aligning with the current trend of incorporating artificial intelligence into clinical practice to strengthen early diagnosis and personalized cardiovascular risk management.

VI. CONCLUSION

Faced with the significant challenges posed by machine learning in the application of algorithms aimed at the diagnosis and prediction of heart diseases, researchers have intensified efforts to develop increasingly accurate, robust, and interpretable models. These models, resulting from the interaction between clinical, genetic, and lifestyle factors, demand computational approaches capable of capturing nonlinear relationships and hidden patterns in medical data. In this context, the present study conducted a comparison of various data mining methodologies oriented toward the development of predictive models in the field of cardiopathy. The approaches analyzed included KDD, SEMMA, ASUM-DM, and CRISP-DM, considering technical, methodological, and applicability criteria. The evaluation results indicated that the CRISP-DM methodology stood out as the most efficient and adaptable to the study context, due to its iterative structure, systematic approach to data understanding, and alignment with the analytical objectives of ML.

In line with the above, the CRISP-DM methodology is structured into five fundamental phases that guide the entire knowledge discovery process from data. The first phase, Business Understanding, establishes the analytical objectives and expected outcomes of the predictive model, taking into account the specific clinical context. The second phase, Data Understanding, enables the characterization of variables through descriptive statistics, identifying their structure and dimensionality without the need to examine each individual observation. The third phase, Data Preparation, encompasses the cleaning, transformation, and normalization of information, correcting missing, inconsistent, or outlier values that could affect modeling quality. In the fourth phase, Modeling, machine learning algorithms are trained with the processed data, enabling the identification of predictive patterns and the reduction of prediction errors. Finally, the fifth phase, Evaluation, compares the generated models using performance metrics such as Accuracy, Precision, Recall, and F1-Score, selecting the one with the greatest generalization capacity and predictive efficacy. The results obtained in this study support the effectiveness of boosting and bagging-based models, such as XGBoost and RF, in predicting cardiovascular diseases. These techniques demonstrated a superior balance between accuracy and generalization capacity compared to LR, which is critical in clinical contexts where classification errors can have significant consequences. In particular, the

XGBoost model emerges as the most promising alternative for application in later stages of the study, including external validation and integration into medical decision support systems, thus contributing to more preventive, personalized, and evidence-based care.

Finally, understanding the implications of coronary heart disease is essential to visualize its causes and guide the implementation of effective technological solutions. Based on the findings obtained through our model, it is suggested to complement the evaluated algorithms with additional technologies, such as deep learning models, integration of medical imaging data, or time-series analysis of vital signs, in order to improve prediction accuracy and efficacy. Future studies could explore the combination of these techniques with clinical decision support systems, as well as the validation of the model across different population contexts, strengthening both medical informatics and the practical application of early diagnostic tools, with a potential impact on reducing cardiovascular mortality and optimizing healthcare resources.

ACKNOWLEDGMENT

Thanks to the University of Sciences and Humanities, which made this project a reality.

REFERENCES

[1] C. W. Tsao, A. W. Aday, Z. I. Almarzooq, Álvaro García. Alonso, A. Z. Beaton, M. S. Bittencourt, A. K. Boehme, A. E. Buxton, A. P. Carson, Y. Commodore-Mensah, M. S. V. Elkind, K. R. Evenson, C. Eze-Nliam, J. F. Ferguson, G. Generoso, J. E. Ho, R. Kalani, S. S. Khan, B. M. Kissela, K. L. Knutson, D. A. Levine, T. T. Lewis, J. Liu, M. S. Loop, J. Ma, M. E. Mussolino, S. D. Navaneethan, A. M. Perak, R. Poudel, M. Rezk-Hanna, G. A. Roth, E. B. Schroeder, S. H. Shah, E. L. Thacker, L. B. VanWagner, S. S. Virani, J. H. Voecks, N. Y. Wang, K. Yaffe, and S. S. Martin, "Heart disease and stroke statistics-2022 update: A report from the american heart association," *Circulation*, vol. 145, no. 8, pp. E153–E639, 2022, doi:10.1161/CIR.0000000000001052.

[2] M. Bhushan, A. Pandit, and A. Garg, "Machine learning and deep learning techniques for the analysis of heart disease: a systematic literature review, open challenges and future directions," *Artificial Intelligence Review*, vol. 56, no. 12, pp. 10 493–10 536, 2023, doi:10.1007/s10462-023-10493-5.

[3] K. Budholiya, S. K. Shrivastava, and V. Sharma, "An optimized xgboost based diagnostic system for effective prediction of heart disease," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 7, pp. 4514–4523, 2022, doi:10.1016/j.jksuci.2020.10.013.

[4] N. L. Fitriyani, M. Syafrudin, G. Alfian, and J. Rhee, "Hdpm: An effective heart disease prediction model for a clinical decision support system," *IEEE Access*, vol. 8, no. —, pp. 133 034–133 050, 2020, doi:10.1109/ACCESS.2020.3010511.

[5] D. Larassati, A. Zaidiah, and S. Afrizal, "Sistem prediksi penyakit jantung koroner menggunakan metode naive bayes," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 7, no. 2, pp. 294–302, 2022, doi:10.29100/jipi.v7i2.2842.

[6] V. Gujrati, S. Joshi, and S. Nagpal, "Logistic regression to predict heart disease," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 7, no. 10, 2023, doi:10.55041/ijsrem25957.

[7] H. E. Hamdaoui, S. Boujraf, N. E. H. Chaoui, B. Alami, and M. Maaroufi, "Improving heart disease prediction using random forest and adaboost algorithms," *International Journal of Online and Biomedical Engineering*, vol. 17, no. 11, 2021, doi:10.3991/ijoe.v17i11.24781.

[8] J. Yang and J. Guan, "A heart disease prediction model based on feature optimization and smote-xgboost algorithm," *Information (Switzerland)*, vol. 13, no. 10, p. 475, 2022, doi:10.3390/info13100475.

[9] M. Abubakar, A. H. Maidabara, Y. M. Malgwi, and A. Mohammed, "Web based heart disease prediction model using machine learning technique," *Computer Science & IT Research Journal*, vol. 5, no. 2, 2024, doi:10.51594/csitrj.v5i2.837.

[10] K. Brown, P. Roshanitabrizi, J. Rwebembera, E. Okello, A. Beaton, M. G. Linguraru, and C. A. Sable, "Using artificial intelligence for rheumatic heart disease detection by echocardiography: Focus on mitral regurgitation," *Journal of the American Heart Association*, vol. 13, no. 2, p. e031257, 2024, doi:10.1161/JAHA.123.031257.

[11] S. Agrawal, "Revolutionizing cardiovascular health: A machine learning approach for predictive analysis and personalized intervention in heart disease," *International Journal for Research in Applied Science and Engineering Technology*, vol. 12, no. 3, 2024, doi:10.22214/ijraset.2024.58797.

[12] H. E. S. Alsafi and O. N. Ocan, "Retraction note: A novel intelligent machine learning system for coronary heart disease diagnosis," *Applied Nanoscience*, vol. 14, no. 3, 2024, doi:10.1007/s13204-024-03012-7.

[13] B. M. R. International, "Retracted: An effective machine learning-based model for an early heart disease prediction," *BioMed research international*, vol. 2024, 2024, doi:10.1155/2024/9754362.

[14] C.-Y. Ma, Y.-M. Luo, T.-Y. Zhang, Y.-D. Hao, X.-Q. Xie, X.-W. Liu, X.-L. Ren, X.-L. He, Y.-M. Han, K.-J. Deng, D. Yan, H. Yang, H. Tang, and H. Lin, "Predicting coronary heart disease in chinese diabetics using machine learning," *Computers in Biology and Medicine*, vol. 169, no. 7, p. 107952, 2024, doi:10.1016/j.compbiomed.2024.107952, artículo en línea con identificador 107952; sin rango de páginas tradicional.

[15] M. M. Mijwil, A. K. Faieq, and M. Aljanabi, "Early detection of cardiovascular disease utilizing machine learning techniques: Evaluating the predictive capabilities of seven algorithms," *Iraqi Journal for Computer Science and Mathematics*, vol. 5, no. 1, pp. 18–32, 2024, doi:10.52866/ijcsm.2024.05.01.018.

[16] P. K. Mall, S. Srivastava, M. M. Patel, A. Kumar, V. Narayan, S. Kumar, P. K. Singh, and D. S. Singh, "Optimizing heart attack prediction through ohe2lm: A hybrid modelling strategy," *Journal of Electrical Systems*, vol. 20, no. 1, pp. 66–75, 2024, doi:10.52783/jes.665.

[17] G. P. Raju, "Multiple disease prediction system," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 8, no. 3, pp. 1–7, 2024, doi:10.55041/ijsrem28972.

[18] A. Ogunpola, F. Saeed, S. Basurra, A. M. Albarrak, and S. N. Qasem, "Machine learning-based predictive models for detection of cardiovascular diseases," *Diagnostics*, vol. 14, no. 2, p. 144, 2024, doi:10.3390/diagnostics14020144.

[19] A. J. Marelli, C. Li, A. Liu, H. Nguyen, H. Moroz, J. M. Brophy, L. Guo, D. L. Buckeridge, J. Tang, A. Y. Yang, and Y. Li, "Machine learning informed diagnosis for congenital heart disease in large claims data source," *JACC: Advances*, vol. 3, no. 2, p. 100801, 2024, doi:10.1016/j.jacadv.2023.100801.

[20] X. Li, Z. Wang, W. Zhao, R. Shi, Y. Zhu, H. Pan, and D. Wang, "Machine learning algorithm for predict the in-hospital mortality in critically ill patients with congestive heart failure combined with chronic kidney disease," *Renal Failure*, vol. 46, no. 1, p. 2315298, 2024, doi:10.1080/0886022X.2024.2315298.

[21] M. Duan, Y. Zhang, Y. Liu, B. Mao, G. Li, D. Han, and X. Zhang, "Machine learning aided non-invasive diagnosis of coronary heart disease based on tongue features fusion," *Technology and Health Care*, vol. 32, no. 1, pp. 441–457, 2024, doi:10.3233/THC-230590.

[22] F. Türk, "Investigation of machine learning algorithms on heart disease through dominant feature detection and feature selection," *Signal, Image and Video Processing*, vol. 18, no. 4, pp. 3943–3955, 2024, doi:10.1007/s11760-024-03060-0.

[23] B. Duraisamy, R. Sunku, K. Selvaraj, V. V. R. Pilla, and M. Sanikala, "Heart disease prediction using support vector machine," *Multidisciplinary Science Journal*, vol. 6, no. 1, pp. 104–112, 2024, doi:10.31893/multiscience.2024ss0104.

[24] A. Kumar, A. K. Singh, and A. Garg, "Evaluation of machine learning techniques for heart disease prediction using multi-criteria decision making," *Journal of Intelligent and Fuzzy Systems*, vol. 46, no. 1, pp. 1–15, 2024, doi:10.3233/JIFS-233443.

[25] G. Saranya and A. Pravin, "Grid search based optimum feature selection by tuning hyperparameters for heart disease diagnosis in machine learning," *The Open Biomedical Engineering Journal*, vol. 17, no. 1, pp. 4371–4385, 2024, doi:10.2174/18741207-v17-e230510-2022-ht28-4371-8.

[26] L. J. V. N and A. Das, "Forecasting the risk of coronary heart diseases using machine learning algorithms," *Computology: Journal of Applied Computer Science and Intelligent Technologies*, vol. 3, no. 2, pp. 45–56, 2024, doi:10.17492/computology.v3i2.2307.

[27] D. D. Solomon, Sonia, K. Kumar, K. Kanwar, S. Iyer, and M. Kumar, "Extensive review on the role of machine learning for multifactorial genetic disorders prediction," *Archives of Computational Methods in Engineering*, vol. 31, no. 2, pp. 789–810, 2024, doi:10.1007/s11831-023-09996-9.

[28] A. A. Almazroi, E. A. Aldhahri, S. Bashir, and S. Ashfaq, "A clinical decision support system for heart disease prediction using deep learning," *IEEE Access*, vol. 11, no. 1, pp. 1–15, 2023, doi:10.1109/ACCESS.2023.3285247.

[29] M. Shokouhifar, M. Hasanvand, E. Moharamkhani, and F. Werner, "Ensemble heuristic–metaheuristic feature fusion learning for heart disease diagnosis using tabular data," *Algorithms*, vol. 17, no. 1, p. 34, 2024, doi:10.3390/a17010034.

[30] N. S. Gupta, S. K. Rout, S. Barik, R. R. Kalangi, and B. Swapna, "Enhancing heart disease prediction accuracy through hybrid machine learning methods," *EAI Endorsed Transactions on Internet of Things*, vol. 10, no. 1, pp. 1–12, 2024, doi:10.4108/eetiot.5367.

[31] S. H. Mulyani, N. Wijaya, and F. Trinidya, "Enhancing heart disease detection using convolutional neural networks and classic machine learning methods," *Journal of Computer, Electronic, and Telecommunication*, vol. 4, no. 2, pp. 394–405, 2024, doi:10.52435/complete.v4i2.394.

[32] S. Srinivasan, S. Gunasekaran, S. K. Mathivanan, B. A. Malar, P. Jayagopal, and G. T. Dalu, "An active learning machine technique based prediction of cardiovascular heart disease from uci-repository database," *Scientific Reports*, vol. 13, no. 1, pp. 1–15, 2023, doi:10.1038/s41598-023-40717-1.

[33] A. Hammoud, A. Karaki, R. Tafreshi, S. Abdulla, and M. Wahid, "Coronary heart disease prediction: A comparative study of machine learning algorithms," *Journal of Advances in Information Technology*, vol. 15, no. 1, pp. 27–32, 2024, doi:10.12720/jait.15.1.27-32.

[34] C. Bernand, E. Mirand, and M. Aryun, "Coronary heart disease prediction models using machine learning and deep learning algorithms," *AIP Conference Proceedings*, vol. 2838, no. 1, p. 020012, 2024, doi:10.1063/5.0179929.

[35] A. A. Maulani, S. Winarno, J. Zeniarja, R. T. E. Putri, and A. N. Cahyani, "Comparison of hyperparameter optimization techniques in hybrid cnn-lstm model for heart disease classification," *Sinkron*, vol. 9, no. 1, p. 13219, 2024, doi:10.33395/sinkron.v9i1.13219.

[36] G. Manikandan, B. Pragadeesh, V. Manojkumar, A. L. Karthikeyan, R. Manikandan, and A. H. Gandomi, "Classification models combined with boruta feature selection for heart disease prediction," *Informatics in Medicine Unlocked*, vol. 44, p. 101442, 2024, doi:10.1016/j.imu.2023.101442.

[37] A. H. Najim and N. Nasri, "Artificial intelligence for heart disease prediction and imputation of missing data in cardiovascular datasets," *Cogent Engineering*, vol. 11, no. 1, pp. 1–18, 2024, doi:10.1080/23311916.2024.2325635.

[38] B. S.P. and D. M., "An optimized deep auto encoder with enhanced extreme learning machine model for heart disease prediction and classification," *International Journal of System of Systems Engineering*, vol. 16, no. 2, pp. 145–160, 2026, doi:10.1504/ijsse.2026.10062277.

[39] P. Rani, R. Kumar, A. Jain, R. Lamba, R. K. Sachdeva, K. Kumar, and M. Kumar, "An extensive review of machine learning and deep learning techniques on heart disease classification and prediction," *Archives of Computational Methods in Engineering*, vol. 31, no. 6, pp. 3871–3895, 2024, doi:10.1007/s11831-024-10075-w.

[40] S. Mondal, R. Maity, Y. Omo, S. Ghosh, and A. Nag, "An efficient computational risk prediction model

of heart diseases based on dual-stage stacked machine learning approaches," *IEEE Access*, vol. 12, no. 1, pp. 3 350 996–3 351 010, 2024, doi:10.1109/ACCESS.2024.3350996.

[41] B. A. Majeed, A. Y. Hardan, B. Y. Hardan, and D. F. Munaf, "Accurate ai-based chatbot to diagnose heart diseases pre-human doctor consultation," *Revue d'Intelligence Artificielle*, vol. 38, no. 1, pp. 121–135, 2024, doi:10.18280/ria.380121.

[42] J. S. Lee, E.-S. Choi, Y. Hwang, K.-S. Lee, and K. H. Ahn, "Preterm birth and maternal heart disease: A machine learning analysis using the korean national health insurance database," *PLoS ONE*, vol. 18, no. 3, p. e0283959, 2023, doi:10.1371/journal.pone.0283959.

[43] N. R. Kolukula, P. N. Pothineni, V. M. K. Chinta, V. G. Boppana, R. P. Kalapala, and S. Duvvi, "Predictive analytics of heart disease presence with feature importance based on machine learning algorithms," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 32, no. 2, pp. 1070–1077, 2023, doi:10.11591/ijeecs.v32.i2.pp1070-1077.

[44] S. M. Muhammed, G. Abdul-Majeed, and M. S. Mahmoud, "Prediction of heart diseases by using supervised machine learning algorithms," *Wasit Journal for Pure Sciences*, vol. 2, no. 1, pp. 231–243, 2023, doi:10.31185/wjps.125.

[45] G. Murugesan, C. Kavitha, G. Jabakumar, and E. Swarnalatha, "Prediction of heart disease using machine learning algorithms with feature selection techniques," *CARDIOMETRY*, vol. 26, no. 26, pp. 778–786, 2023, doi:10.18137/cardiometry.2023.26.778786.

[46] S. Mishra, N. K. Tiwari, K. Kumari, and V. Kumawat, "Prediction of heart disease using machine learning," 2023, doi:10.1109/ICAAIC56838.2023.10140478.

[47] A. A. Ahmad and H. Polat, "Prediction of heart disease based on machine learning using jellyfish optimization algorithm," *Diagnostics*, vol. 13, no. 14, 2023, doi:10.3390/diagnostics13142392.

[48] P. Yu, M. Skinner, I. Esangbedo, J. J. Lasa, X. Li, S. Natarajan, and L. Raman, "Predicting cardiac arrest in children with heart disease: A novel machine learning algorithm," *Journal of Clinical Medicine*, vol. 12, no. 7, 2023, doi:10.3390/jcm12072728.

[49] G. A. Ansari, S. S. Bhat, M. D. Ansari, S. Ahmad, J. Nazeer, and A. E. Eljialy, "Performance evaluation of machine learning techniques (mlt) for heart disease prediction," *Computational and Mathematical Methods in Medicine*, vol. 2023, 2023, doi:10.1155/2023/8191261.

[50] S. Tuba, "Optimization heart disease prediction using machine learning models," *Fidelity : Jurnal Teknik Elektro*, vol. 5, no. 1, p. 142, 2023, doi:10.52005/fidelity.v5i1.142.

[51] K. M. M. Uddin, R. Ripa, N. Yeasmin, N. Biswas, and S. K. Dey, "Machine learning-based approach to the diagnosis of cardiovascular vascular disease using a combined dataset," *Intelligence-Based Medicine*, vol. 7, p. 100100, 2023.

[52] Y. M. Ayano, F. Schwenker, B. D. Dufera, and T. G. Debelee, "Interpretable machine learning techniques in ecg-based heart disease classification: A systematic review," *Diagnostics*, vol. 13, no. 1, p. 111, 2023.

[53] L. M. Paladino, A. Hughes, A. Perera, O. Topsakal, and T. C. Akinci, "Evaluating the performance of automated machine learning (automl) tools for heart disease diagnosis and prediction," *AI (Switzerland)*, vol. 4, no. 4, pp. 53–68, 2023, doi:10.3390/ai4040053.

[54] A. J. Albert, R. Murugan, and T. Sripriya, "Diagnosis of heart disease using oversampling methods and decision tree classifier in cardiology," *Research on Biomedical Engineering*, vol. 39, no. 1, pp. 1–12, 2023, doi:10.1007/s42600-022-00253-9.

[55] L. V. Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, J. Pfrommer, A. Pick, R. Ramamurthy, M. Walczak, J. Garcke, C. Bauckhage, and J. Schuecker, "Informed machine learning - a taxonomy and survey of integrating prior knowledge into learning systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 1–20, 2023, doi:10.1109/TKDE.2021.3079836.

[56] I. H. Sarker, "Machine learning: Algorithms, real-world applications and research directions," *SN Computer Science*, vol. 2, no. 3, p. 160, 2021, doi:10.1007/s42979-021-00592-x.

[57] Y. Xu, Y. Zhou, P. Sekula, and L. Ding, "Machine learning in construction: From shallow to deep learning," *Developments in the Built Environment*, vol. 6, p. 100045, 2021, doi:10.1016/j.dibe.2021.100045.

[58] S. Raschka, J. Patterson, and C. Nolet, "Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence," *Information (Switzerland)*, vol. 11, no. 4, p. 193, 2020, doi:10.3390/info11040193.

[59] R. M. Radke, T. Frenzel, H. Baumgartner, and G. P. Diller, "Adult congenital heart disease and the covid-19 pandemic," *Heart*, vol. 106, no. 17, 2020, doi: 10.1136/heartjnl-2020-317258.

[60] P. Severino, A. D'Amato, M. Pucci, F. Infusino, L. I. Birtolo, M. V. Mariani, C. Lavalle, V. Maestrini, M. Mancone, and F. Fedele, "Ischemic heart disease and heart failure: Role of coronary ion channels," p. 3167, 2020, doi:10.3390/ijms21093167.

[61] R. H. Ritchie and E. D. Abel, "Basic mechanisms of diabetic heart disease," *Circulation Research*, vol. 126, no. 11, pp. 1501–1525, 2020, doi:10.1161/CIRCRESAHA.120.315913.

[62] Instituto Nacional de Estadística e Informática (INEI), "Perú: Enfermedades no transmisibles y transmisibles," 2025, [Accedido: 14-mayo-2025].

[63] N. Chandrasekhar and S. Peddakrishna, "Enhancing heart disease prediction accuracy through machine learning techniques and optimization," *Processes*, vol. 11, no. 4, pp. 1210–1221, 2023, doi:10.3390/pr11041210.

[64] V. Venkat, H. Abdelhalim, W. DeGroat, S. Zeeshan, and Z. Ahmed, "Investigating genes associated with heart failure, atrial fibrillation, and other cardiovascular diseases, and predicting disease using machine learning techniques for translational research and precision medicine," *Genomics*, vol. 115, no. 2, p. 110584, 2023, doi:10.1016/j.ygeno.2023.110584.

[65] N. Nissa, S. Jamwal, and M. Neshat, "A technical comparative heart disease prediction framework using boosting ensemble techniques," *Computation*, vol. 12, no. 1, p. 15, 2024, doi:10.3390/computation12010015.

[66] R. Canelón, C. Carrasco, and F. Rivera, "Design of a remote assistance model for truck maintenance in the mining industry," *Journal of Quality in Maintenance Engineering*, vol. 30, no. 1, pp. 1–337, 2024, doi:10.1108/JQME-02-2023-0024.

[67] S. Mirfakhraei, N. Abdolvand, and S. R. Harandi, "The rfmrv model for customer segmentation based on the referral value," *Interdisciplinary Journal of Management Studies*, vol. 17, no. 2, pp. 455–473, 2024, doi:10.22059/ijms.2023.329229.674722.

[68] C. Wang, A. Stupina, and S. Bezhitskiy, "Online sales prediction approach using methodology of crisp-dm," *ITM Web of Conferences*, vol. 59, p. 01006, 2024, doi:10.1051/itmconf/20245901006.

[69] S. Kanimozhi and N. Sivanandan, "Machine learning based heart disease prediction system," p. 03013, 2024, doi:10.1051/e3sconf/202449103013.

[70] P. K. Rajani, K. Patil, B. Marathe, P. Mhaisane, and A. Tundalwar, "Heart disease prediction using different machine learning algorithms," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 9S, pp. 7430–7442, 2023.

[71] S. Roobini, M. S. Kavitha, and S. Karthik, "A systematic review on machine learning and neural network based models for disease prediction," *Journal of Integrated Science and Technology*, vol. 12, no. 4, pp. 787–787, 2024, doi:10.62110/sciencein.jist.2024.v12.787.

[72] P. Pachiyannan, M. Alsulami, D. Alsadie, A. K. J. Saudagar, M. AlKhathami, and R. C. Poonia, "A novel machine learning-based prediction method for early detection and diagnosis of congenital heart disease using ecg signal processing," *Technologies*, vol. 12, no. 1, pp. 1–15, 2024.

[73] S. Thangavel, S. Selvaraj, V. G. Karthikeyan, and K. Keerthika, "Analyzing machine learning classifiers for the diagnosis of heart disease," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 10, no. 1, pp. 1–12, 2024, doi:10.4108/eetpht.10.5244.

[74] S. V. Babu, P. Ramya, and J. Gracewell, "Revolutionizing heart disease prediction with quantum-enhanced machine learning," *Scientific Reports*, vol. 14, no. 1, p. 7453, 2024, doi:10.1038/s41598-024-55991-w.

[75] A. M. Qadri, A. Raza, K. Munir, and M. S. Almutairi, "Effective feature engineering technique for heart disease prediction with machine learning," *IEEE Access*, vol. 11, no. 1, pp. 1–12, 2023, doi:10.1109/ACCESS.2023.3281484.

Integrating Processing-In-Memory into HW/SW Co-Design for Automotive Embedded Systems

Zineb El Kacimi, Safae Dahmani, Oussama Elissati, Mouhcine Chami
Institut National des Postes et Télécommunications, STRS Lab., Rabat, Morocco

***Abstract*—Continuous expansion of using intelligent learning workloads in modern vehicles, leads Electronic Control Units (ECUs) to manage massive volumes of data while dealing with hard real-time constraints. That said, ECUs must operate under strict power budget due to limited battery capacity and other safety and functional requirements. We study in this research the feasibility of integrating Processing-In-Memory (PIM) approach into the hardware/software co-design process for the emerging ECU architectures. The idea here is to allocate data-centric tasks to in-memory compute units so that computation occurs directly where data is stored. The proposed approach will allow avoiding expensive data traffic, which will improve processing performance and reduce energy consumption. Our work introduces a conceptual framework that reconciles the PIM strengths with automotive requirements and introduces techniques from dynamic voltage scaling to smarter memory management, and is limited to a conceptual architectural proposal without experimental validation or quantitative evaluation. We end up discussing some crucial opportunities and open challenges arising from the implementation of PIM in next-generation automotive systems.**

Keywords*—Electronic Control Units; energy consumption; hardware/software co-design; Processing-In-Memory; real-time constraints

I. Introduction

Nowadays, automotive industry is growing very quickly towards smart electric vehicles, which bring together assisted and autonomous driving, high connectivity, and a wide range of intelligent features. With the increase in complexity and performance demands, These developments marked a shift in embedded systems, especially Electronic Control Units (ECUs), the main nervous system of modern cars. It is important to highlight that ECUs are responsible for managing features from real-time sensor operations to battery management and various advanced driver assistance systems (ADAS) among many other functions. With the increase of data-centric features, we are driving classical design approaches towards increased computational power and real-time accuracy.

A focus on hardware/software co-design approaches is a key point to deal with new embedded systems. It works on optimizing both hardware and software together to achieve better performance and cost-effectiveness across the whole system. This technique has already demonstrated results in ECU subsystems: engine control [1], security firewalls [2], and reconfigurable driver designs [3]. Yet, the major remaining limitations are performance bottleneck related to data movement, scaling across different ECU types, and low latency under strict power constraints.

In such a context, Processing-In-Memory (PIM) [4] offers a promising computing paradigm, which can transform how current data processing in embedded devices is conducted. By processing computation close to the memory components, PIM immediately relieves the typical processor-memory bottleneck, leading to a significant reduction in the amount of data transfer and energy consumption. This computing paradigm has been reported successfully in accelerating data-dense applications such as Artificial Intelligence (AI) inference, graph processing, and real-time signal analysis [5], [6].

The importance of the growing use of AI and deep learning in modern vehicles, as reviewed in [7], [8], also highlights the demand for new, disruptive architectural approaches to handle complex, data-driven tasks. Having shown capabilities in parallel data flows, latency reduction, and low power costs, PIM can provide a great value to automotive applications because they are performance and power-constrained applications in addition to data-intensive, especially with the increasing use of AI.

Integrating PIM into hardware/software co-design moves data-intensive work to where the data already resides, reducing transfer overhead and latency. In practice, during the hardware/software co-design partitioning step, workloads with memory-bound tasks run on PIM while the control-heavy tasks stay on the CPU. That leads to balanced resources utilization and helps improve real-time responsiveness. Existing literature shows limited exploration of PIM integration in the automotive domain. The systematic integration of PIM into ECU architectures is highly dependent on its compatibility with automotive standardized hardware/software co-design techniques, and the challenges that can be faced to meet the requirements of automotive safety-critical systems.

This study aims to introduce a conceptual framework that combines hardware/software co-design principles with emerging PIM architectures to support the development of next-generation ECUs.

This study is organized as follows: Section II reviews the state-of-the-art of the hardware/software co-design in automotive embedded systems, focusing on ECUs and identifying bottlenecks that PIM can alleviate. Section III presents an analysis of recent advancements in PIM architectures across different memory technologies, in addition to a proposal for PIM-enabled ECU architecture and techniques for its integration. Section IV details the motivation and pathways for integrating PIM into automotive ECU architectures. Section V discusses challenges and opportunities for integrating PIM methodology into automotive ECUs. Finally, Section VI concludes the study along-with a declaration on the use of generative AI in Section VII.

Although no experimental implementation is provided,

this work lays out the foundation for future development, prototyping, and evaluation, and we are currently exploring comparative benchmarking using existing gem5-based simulation tools.

II. Hardware-Software Co-Design in Automotive Embedded Systems

A. The Need for Hardware/Software Co-design in Automotive Embedded Systems

Hardware/software co-design is a methodology in which both the hardware and the software are considered interdependent components of a system, aiming to optimize them concurrently in the design process. This collaborative process helps in the early evaluation of trade-offs between performance, power, cost, area, safety, and flexibility to form integrated systems that are optimized to meet specific requirements at minimal cost. Hardware/software co-design, as outlined by Shaout, El-Mousa and Mattar in their work on the heterogeneous embedded systems [9], is an approach that consist of several activities including system partitioning, in which functions are strategically allocated between hardware and software according to performance, cost, and power limitations. Then this division is further optimized in task allocation and scheduling, matching tasks to meet timing constraints and reduce system latency. In addition, hardware/software interface definition helps in establishing clear communication flows for building and ensuring the compatibility of the components, while performance modeling and simulation help to identify the potential bottlenecks early on the design phase. Also, design space exploration is further an important step in which various levels of analysis can be made that allows for the trade-offs between competing needs (performance versus power consumption). Co-synthesis and co-verification guarantee accurate joint behavior, so hardware and software are developed and tested in parallel. Finally, integration and validation are the last steps, enabling the testing of the entire system to determine whether it meets its design specifications.

In sectors, where the rate of technological developments is accelerating, such as automotive, it becomes even more pressing to adopt such a method. Given the increasing adoption of embedded computing systems used in cars, hardware/software co-design is becoming an apparent need. A case study of how this method can expedite computationally intensive tasks, and be used to accelerate computation has been reported in embedded security, with the Elliptic Curve Cryptography accelerator on a Zynq SoC producing impressive efficiency gains based on an efficient hardware/software partitioning [10]. Real-time performance, safety, power efficiency, and cybersecurity are now fundamental requirements in modern vehicles, and hardware/software co-design ensures that computing architectures are optimized to meet these constraints while supporting the integration of evolving components and secure connectivity, as highlighted by Abdallah et al. [11].

Integrating PIM into hardware/software co-design can create new opportunities to improve the balance of performance and power efficiency. Thus, PIM also reduces overall overheads and latencies caused by data transfer by directly conducting data-heavy computations in memory, which helps to optimize system partitioning and resource allocation. Inherently, hardware/software co-design frameworks need to consider the computational capacity of the PIM-enabled memory and allocate tasks such as sensor-centric data handling to memory, whereas other high-level control functions should be retained by the CPU. Task scheduling can take further advantage of the parallel data flows generated by PIM and reduce latency and energy usage.

Given the increasing complexity of embedded computing systems in modern vehicles, we are seeing a rise in the number of ECUs, each packed with multiple software features, which leads to an obvious need for updating hardware/software co-design process to meet the new design requirements and ensure both reliability and performance. In [12], Axelsson points out that to address these challenges, hardware/software co-design is needed across the overall system architecture, not only at the ECU level.

Hardware/software co-design has moved from being a "nice-to-have" best practice to a core engineering discipline, therefore, adopting advanced co-design methods is essential to balance today's competing priorities including performance, cost, safety, and power.

Current hardware/software co-design strategies provide different solutions for varying ECU types, each with its own trade-offs. We will review these approaches to point out current weaknesses, particularly for handling real-time performance of data extensive tasks, scalability and power efficiency. This analysis provides the foundation for evaluating the architectural shift towards integrating PIM to address these core limitations.

B. ECU: The Heart of Automotive Electronics

An ECU is a specialized embedded system, which controls various functions in a larger system, mainly through sensor inputs, control algorithms, and actuator operation. Modern automotive systems have very high use cases of ECUs that make them perform from basic tasks to high-end processes. Automobile ECUs were originally single-unit components: a microcontroller, sensors, and actuators connected via point-to-point wiring with other vehicle components. This simple architecture was acceptable for early vehicle designs with minimal electronic requirements (see Fig. 1). For instance, a Powertrain Control Unit (PCU) manages engine and transmission functions, ensuring optimal fuel efficiency and emissions compliance. Meanwhile, a Body Control Module (BCM) handles auxiliary functions such as lighting and climate management. For the driver and passengers, infotainment ECUs consolidate all multimedia and connectivity functions. However, the most computationally intensive tasks are computed by ADAS systems ECUs, which receive real-time input from cameras, radar, and LiDAR to perform features like adaptive cruise control and collision avoidance. These cases demonstrate that ECUs support everything in the modern automobile from its most basic operations to its most advanced autonomous functions.

It's clear that today's vehicles come up with significantly more complex needs compared to initial automotive requirements. The current automotive environment has set new standards for ECU design [13]. In fact, modern architectures go beyond standalone microcontroller units and evolve toward more

integration of domain controllers and centralized computing platforms [14], [15].

The most obvious challenge in these architectures is the incorporation of workloads which demand considerable computational power to process large volumes of sensor data in real-time. Traditional ECUs were designed primarily for the purpose of relatively simple control applications, but for these demanding applications often lack both the processing capacity and architectural flexibility needed to support such demanding applications. Thus, to handle the computationally intensive tasks of applications like sensor fusion and real-time control, current designs integrate dedicated hardware accelerators. This hybrid architectural evolution reflects the industry's adaptation to growing functional complexity and performance demands.

In the next subsection, we will explore the existing hardware-software co-design methodologies to highlight opportunities for further optimization, which leads us to novel approaches such as PIM to address the challenges of next-generation ECU design.

C. Existing Hardware/Software Co-design Methodologies for ECUs

Various hardware-software co-design strategies have been explored in the area of ECUs, one of the most indispensable components of modern automotive systems. The goal of these hardware/software co-design methods is to find the optimal split of tasks between hardware and software, incorporating the best of both worlds to satisfy the tight performance, power, and security specifications. For instance, hardware/software partitioning of crankshaft functionality on an FPGA delivered a 58% performance improvement [1]. This demonstrates co-design's ability to optimize critical automotive functions. Yet, a significant bottleneck is scalability; extending this approach to other ECUs is indeed a significant challenge. Coupled with the high cost of FPGA-based solutions that may lead to a lack of widespread industry adoption.

In the context of security domain, Pesé et al. [2] introduced an embedded firewall combined of hardware and software to secure in-vehicle networks, such as Automotive Ethernet, this firewall has been shown to minimize latency and jitter while maintaining constant throughput under dynamic conditions. Nevertheless, integration issues and resource limitations restrict this system's applicability across broader automotive systems. Another subsequent study [3] proposes a reconfigurable ECU architecture that co-integrates security and dependability primitives to address the critical need for security without sacrificing performance, reporting a 47.9×speedup and 2.4×better energy efficiency compared to a multi-core automotive processor. This work provides quantifiable evidence that hardware/software co-design can successfully overcome the traditional performance versus security trade-off.

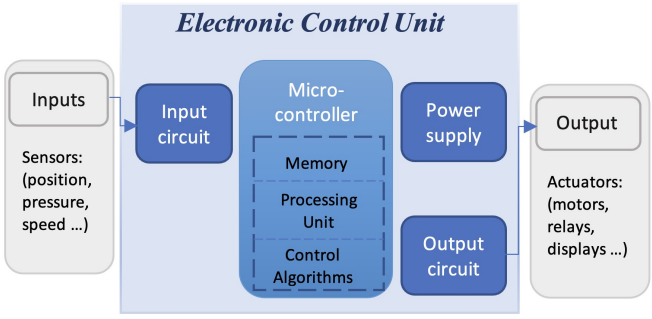


Fig. 1. Simplified architecture of early ECUs with basic microcontroller-based designs.

This principle of embedding efficient, transparent security is further demonstrated by a lightweight FPGA-based IDS-ECU for the CAN bus [16]. The architecture introduces a hardware-accelerated intrusion detection system that operates transparently to the primary ECU software. Compared to GPU-based implementations, it achieved a 2.3×reduction in per-message latency and a 15×reduction in power consumption. This shows that integrating built-in operating capabilities, such as security monitoring, into ECUs not only doesn't compromise its performance but can even improve it. FPGA-based remote reconfigurable ECUs suggested in [17] offer flexibility and adaptability but struggle in ensuring secure remote updates and managing implementation costs. Another hardware-software co-design approach was developed in [18] for an automotive application, a Multiple Injection Driver Automotive Subsystem. This approach consists of a configurable system-on-chip (CSoC), which consists of a microprocessor and embedded FPGA. Implementing this technique enabled rapid system design cycles, promoting component re-use, and working within tight timelines for complex embedded system. However, the authors note that a key limitation of their specific flow was that the interface synthesis was not fully automatic, requiring some minimal manual steps, which presents a challenge for complete design automation. While these studies have shown considerable promise in optimizing ECU performance, power consumption and security, modern automotive systems are incorporating increasingly data-intensive applications that further introduce the bottleneck of data movement, which leads to higher power consumption and limits real-time processing. To address these challenges, PIM can complement hardware/software co-design methodologies by offloading data-intensive tasks from traditional processors, and this can also ensure robust protection against dynamic threats by accelerating intrusion detection algorithms and firewall operations, in the context of automotive security applications.

The subsequent section will discuss current research on PIM technology, including its underlying mechanisms, and examine its potential to tackle challenges in modern automotive systems.

III. Overview of PIM and Related Works

PIM is a modern computing paradigm that integrates computational capabilities in or near memory systems. This approach was introduced to mitigate the performance limitations of the traditional Von Neumann architecture. In the latter, processors are physically separated from memory, causing heavy traffic between them. Significant latency and energy overhead are caused by memory-processor bottlenecks. This is particularly true for intensive data workloads which is the case for intelligent learning, graph processing or sensor fusion computations in automotive. By putting processing close to

where the data is stored, PIM minimizes data movement, reduces network related latency, in addition to important energy savings.

The processor-memory bottleneck is a big challenge for modern computing systems. For instance, [19] states that traditional architectures allocate up to 99.75% of their energy consumption in floating-point operations on memory access rather than computation itself. PIM solves this problem by fitting processing units directly into existing memory modules to enable operations, like matrix multiplications, filtering, and pattern recognition, to be done in place. This makes PIM a good match for AI-driven tasks.

A. Overview of PIM Architectures

The performance of PIM architectures heavily depends on the developments in memory technologies, design approaches, and application fields. These factors provide the foundation for integrating computational capabilities directly within memory systems. In the following, we describe the important characteristics of PIM by design type, application field, and memory technology, according to Asifuzzaman et al. in [5].

1) PIM by design type: PIM architecture can be classified according to its computational features integrated into or near memory system. Table I provides descriptions and features of the design types.

2) PIM by application field: There are several application domains where PIM designs help to achieve maximum efficiency. Table II illustrates the major application fields and their use cases.

3) PIM by memory technology: Memory technologies are a major factor for effective PIM implementation. Table III presents the common memory technologies and their suitability for PIM.

The presented classifications highlight diverse PIM-based approaches, addressing conventional memory limitations considering emerging applications' demands. In the next subsection, we investigate how developments in PIM bring transformational performance and energy efficiency within different sectors.

B. Advancements in PIM Architectures

In this section, we present a wide range of examples of PIM-based architectures. Associative In-Memory Processors (AiMP), which are discussed in [20], utilize resistive memory for direct access to memory arrays, improving throughput and energy consumption in clustering and matrix operations. In [21], the authors demonstrate that hybrid schemes like Hetraph, are able to combine analog and digital processors, and this combination tradeoff for performance and energy efficiency for irregular workloads such as graph processing. Hetraph provides up to 7.54×speedup over CPUs, and 1.56×over GPUs, resulting in significant energy savings of up to 57.58×vs CPUs, 19.93×vs GPUs.

Beyond recent hardware innovations, software approaches have been developed to fully utilize the potential of PIM. In this regard, for example, the work in [19] presents a hardware/software co-design methodology, which specifically targets associative in-memory processors and employs dynamic voltage scaling techniques and selective instruction scheduling. On average, the proposed method achieves a 7% decrease in memory area and an 18% reduction in overall energy consumption.

Moreover, the PIM architectures programming frameworks focus on task division between control-intensive (CPU) and data-intensive (PIM) components, avoiding unnecessary information exchanges between memory and processors. Similar paradigms have been applied to IoT systems, as demonstrated in [22], where they improve data processing time and power consumption, leading to enhanced utilization efficiency. In addition, J. Joo et al. [23] examine the utilization of PIM-based systems for regular expression matching and report an up to 96% reduction in execution time in comparison to legacy multithreaded implementation methods. Another interesting work [6] introduced the TransPIM that integrates processing capabilities in its memory for transformer workloads. It demonstrates performance gains of 22.1×to 114.9×compared to GPUs. Its token-based processing approach minimizes data movement and enhances scalability for long sequences, making it suitable for real-time applications like sensor fusion. Similarly, the iMTransformer, presented in [24], utilizes PIM technologies to accelerate attention mechanisms in transformer networks with 13.71×speedup and 12.57×better energy efficiency compared to GPUs. Another approach presented in [25] advances in memory Big Data processing by combining efficient indexing (e.g., B trees, R trees), parallelism (scale up/scale out in systems like Apache Spark), and memory optimization (e.g., RowClone). These techniques reduce latency and improve efficiency by executing bulk operations near memory.

Due to its previously discussed capacity to improve energy consumption, latency and throughput, PIM technology offers an effective solution to address the growing data-centric challenges in automotive systems, a domain that is rapidly evolving toward electric, connected and autonomous vehicles.

IV. Motivation and Pathways for Integrating PIM into Automotive ECU Architectures

The ECUs have been heavily demanded by the rapid technological change in automotive systems, including autonomous driving, ADAS, and electric vehicles (EVs). Given the need for high-throughput and real-time sensor data such as LiDAR, radar, cameras, etc., embedded systems must maintain specific constraints including latency, energy consumption, safety, and cost. At the same time, traditional Von Neumann architectures introduce bottlenecks caused by intensive processor-memory data movement, which is associated with high energy consumption, latency, and poor scalability, and this is limiting the ability of ECUs to meet the growing complexity of modern automotive systems.

A. Unlocking the Potential of PIM for Automotive Applications

Below, we provide a table (see Table IV) for the key advantages of PIM for automotive systems. The table also gives some examples of ECU applications best optimizing the given advantages.

TABLE I. TYPES OF PIM DESIGNS

Design Type	Description	Examples/Key Features
Cell-level PIM	Simple operations performed at the cell level of the memory array.	Basic arithmetic or logical operations within memory cells.
Sense amplifier / row-buffer PIM	More complex operations performed at the sense amplifier or row-buffer level.	Suitable for operations requiring faster access to larger memory segments.
Near-memory PIM	Processing cores placed near memory banks to execute a subset of CPU instructions.	Enables larger-scale computations and supports diverse workloads.
Hybrid PIM	Combines multiple PIM approaches (e.g., analog and digital processing) to optimize performance.	Supports irregular workloads and balances efficiency for diverse applications.

TABLE II. PIM APPLICATION FIELDS AND THEIR KEY USE CASES

Application Field	Description	Examples/Use Cases
Deep Neural Networks (DNNs)	Optimized for compute patterns of DNNs using specialized memory technologies like ReRAM.	Matrix multiplications, convolution operations, and attention mechanisms in transformers.
Graph Processing	Designed for irregular workloads and high data movement requirements.	Algorithms like BFS, PageRank, and graph clustering.
AI and Machine Learning	Accelerates AI workloads with in-memory computations to reduce latency and energy consumption.	Transformer workloads, pattern recognition, and filtering tasks.
IoT and Edge Computing	Enhances data processing efficiency for real-time applications in IoT systems.	Sensor fusion, real-time analytics, and distributed systems.

TABLE III. COMPARISON OF MEMORY TECHNOLOGIES SUITABLE FOR PIM

Memory Technology	Description	Suitability for PIM
DRAM	Widely used commodity memory technology with affordable costs.	Limited scalability but suitable for general-purpose PIM designs.
HBM (High-Bandwidth Memory)	3D-stacked DRAM variant offering high bandwidth and efficiency.	Ideal for data-intensive applications requiring fast memory access.
HMC (Hybrid Memory Cube)	3D-stacked memory integrating DRAM layers with a logic base layer.	Reduces latency and increases bandwidth for high-performance computing.
STT-MRAM	Non-volatile memory using magnetic properties to store data.	High endurance and fast access speeds make it suitable for frequent data access.
ReRAM	Non-volatile memory using resistive switching for data storage.	Performs computations directly within its unique data array.
PCM (Phase Change Memory)	Relies on material phase changes for data storage, offering high density and scalability.	Suitable for large-scale data storage and processing.

TABLE IV. WHY PIM IS ESSENTIAL FOR AUTOMOTIVE SYSTEMS

Key Benefit	Brief Description	Example ECU/System
Reduction in Data Movement	By performing computations directly in memory, PIM minimizes data transfer between processors and memory, reducing latency and energy consumption.	ADAS ECUs: Real-time sensor fusion for lane detection and collision avoidance.
Improved Energy Efficiency	PIM reduces energy usage by eliminating unnecessary data movement and optimizing memory operations, making it ideal for energy-constrained systems.	Battery Management Systems: Efficient monitoring and optimization of EV battery performance.
Scalability for Complex Workloads	PIM architectures can handle the growing complexity of workloads, such as neural network inference and multi-sensor fusion, without performance bottlenecks.	Autonomous Driving Systems: Path planning and obstacle detection using data from cameras, radar, and LiDAR.
Real-Time Performance	PIM accelerates time-sensitive computations, ensuring ECUs can meet stringent latency requirements in critical systems.	Braking and Steering ECUs: Fast decision-making for ABS and electronic steering control.

B. Techniques for Integrating PIM into ECU Architectures

1) Dynamic Voltage Scaling (DVS): As shown by Yantir et al. [19], can be effectively integrated into hardware/software co-design to optimize energy efficiency and responsiveness in PIM-enhanced subsystems like Powertrain or Braking Control. Dynamic Voltage Scaling is a co-design methodology realized by combining the need for tight coupling of hardware configurability with software control. Multi-VDD designs enable the dynamic adjustment of memory bank voltage in response to computational load at the hardware level. Simultaneously at the software level, a scheduler is used to align the execution of the task with these power states. To illustrate, during steady cruising voltage may be decreased for saving power, rise for acceleration or scale with braking intensity to maintain safety and fulfil real-time computational demands. Through the coupling of hardware voltage scaling with software task scheduling, DVS guarantees an effective operation without affecting system responses in mission-essential automotive applications.

2) Hierarchical memory management: Divides data into memory tiers; mission-critical tasks can be performed within the rapid SRAM, while less urgent tasks within slow DRAM

or STT-RAM to balance latency and efficiency. PIM, along with an optimized hardware/software co-design, improves this by allowing computations to run within the right memory tier with reduced distance to data. For instance, brake and steering ECUs run on PIM-based SRAM for real-time, low-latency processing, and infotainment programs to perform PIM tasks in slow-tier memory to free faster memory. Research on STT-RAM caches [26] shows that heterogeneous memory designs can optimize retention time and energy, and hierarchical PIM therefore is well-suited to different automotive workloads. Complementing this, the study on Big Data architecture, presented in [27], highlights the importance of a distinct memory management layer, which automatically caches frequently used data and optimizes data locality. This concept aligns with hierarchical memory management in its goal of improving performance by minimizing data movement and ensuring that critical tasks are stored and executed in the most appropriate memory tier.

3) Sensor-specific processing: Uses token-based dataflows that allow individual sensors, including cameras, radar, and LiDAR, to process their data independently in PIM-enabled memory and in compliance with hardware/software co-design principles. This technique, inspired by the TransPIM architecture [6], exploits a token-based processing protocol to manage sensor workloads through local processing of data tokens in memory. Every token, representing a unit of information based on the sensor data including for instance, lane detection from cameras or speed from radar, is processed directly in memory, thus reducing data movement and latency. Building on this work, hardware/software co-design frameworks facilitate system partitioning to distribute computation-intensive operations to PIM near the sensor, while software algorithms dynamically coordinate job scheduling and data aggregation among sensors on-the-fly. The main advantage of this approach is parallelizing sensor data processing in memory through concurrent scheduling of the workload and optimal distribution of tasks. Such an integration can then, minimize latency and improve performance for real-time applications such as ADAS and autonomous driving.

4) Task Partitioning between CPU and PIM: Aligns with hardware/software co-design principles, ensuring efficient resource utilization by prioritizing workloads according to their nature. On one side, you have control-intensive tasks, such as temperature sampling and adjusting speeds, are expected to be done on the CPU. On the other hand, data-driven tasks like data consolidation for sensor data shall be computed in memory for faster processing. Hardware/software co-design frameworks allow this partitioning by ensuring that the right tasks can be dynamically assigned to the right resource. The latter structure works well for ADAS and allows PIM accelerators to perform real-time image recognition, while safety-critical control operations such as braking and steering modifications are run simultaneously on the CPU. Just like battery management systems in EVs can use PIM to analyze sensor data while control functions, like adjusting the charging rate, remain on the CPU.

C. Proposed PIM-Enabled ECU Architecture

A main approach for integrating PIM within automotive systems is to optimize a single ECU by embedding processing capabilities directly into its memory subsystem. As an example, we choose a collision avoidance ECU (see Fig. 2); the system integrates a CPU and a PIM-enabled memory module to handle critical data processing tasks. The PIM units can perform early-stage computations on sensor inputs, such as radar and camera data, and execute functions like object detection, distance estimation, as well as threat classification directly in memory, while the CPU will take care of high-level decisions and control logic. Hence, more complex systems can be built using this architecture.

Following this, the architecture can be expanded to a more complex system, which is derived from the Domain Controller ECU paradigm presented in [28]. It promotes the integration of multiple distributed ECUs into a centralized framework. It allows for real-time processing, simplified maintenance, and over-the-air (OTA) software updates. In that concept, diverse domain-specific tasks are performed in a single central ECU to effectively serve as the system's command center. Taking this paradigm and applying it to the ECU system, our proposed architecture works towards implementing PIM capabilities to offload and accelerate data-intensive tasks.

Fig. 3 illustrates the proposed conceptual integration of PIM into a centralized automotive ECU architecture. The system is organized around a central ECU platform responsible for high-level control, task scheduling, and coordination. Based on intelligent workload partitioning, the highly parallel tasks are offloaded from the central processor to a shared PIM-enabled memory subsystem.

However, certain sensor data requiring immediate high processing is sent directly to the central CPU before being forwarded to the PIM subsystem for data-heavy computations.

This subsystem consists of specialized PIM blocks, each associated with a different sensor domain, such as camera, radar, or LiDAR, and capable of performing early-stage, in-memory processing. Tasks like AI-based image filtering, Doppler speed estimation, and neural network inference are executed directly within the memory units. The output from these PIM units is stored in a hierarchical memory structure that uses fast SRAM for time critical control loops and non-volatile STT-RAM or DRAM for storing historical or less critical data. The information collected is preprocessed and then sent to the central CPU, where high-level decisions are made based on this refined data. Then, the CPU generates control commands that are forwarded directly to the actuators, such as those responsible for advanced driver-assistance systems (ADAS). In

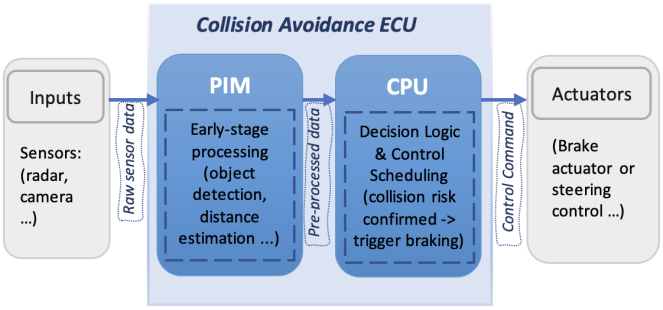


Fig. 2. Architecture of PIM integrated in collision avoidance ECU.

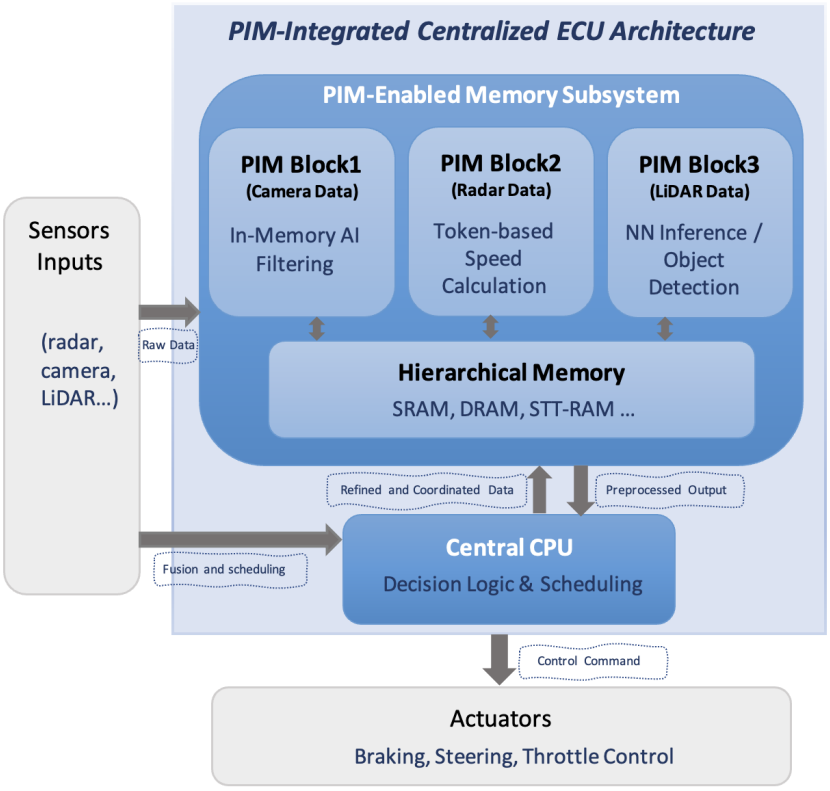


Fig. 3. PIM centralized ECU architecture.

PIM architectures, intelligent processing is distributed across different blocks of the memory subsystem and the CPU. Real-time responsiveness is then enhanced thanks to the reduction of memory traffic and redundant computations.

To conclude, the PIM-based hybrid architecture is showing promising performance and energy efficiency improvements in the context of emerging complex applications.

V. Challenges and Opportunities

Integrating PIM into automotive ECU architectures is not an easy task. Several critical challenges are identified to create successful PIM-based heterogeneous systems. Automotive ECUs from multiple generations communicate over deterministic networks, such as CAN, LIN, and FlexRay. Hence, PIM components, as new nodes, must remain compatible with these legacy buses and existing communication stacks. Second, the absence of standardized programming paradigms and compiler support for PIM limits its accessibility for embedded developers. Here are several aspects of potential solutions that still need development: high-level APIs, task schedulers, hardware/software interfaces that facilitate the adoption of PIM in automotive software stacks, particularly under AUTOSAR or embedded Linux environments. Additionally, automotive applications require deterministic timing behavior to satisfy real-time constraints, and PIM will introduce new execution patterns that will further complicate latency and worst-case execution time (WCET) analysis, while these analysis are crucial for real-time safety-critical tasks. From a security perspective, PIM-based systems need to maintain robust security measures to prevent system level threats such as malicious data injection or memory-side channel attacks. Moreover, fault isolation between PIM and CPU elements is necessary to avoid failures spreading across critical functions.

VI. Conclusion

A conceptual framework for implementing a PIM model within automotive ECU architectures is proposed in response to some of the core challenges in embedded systems, such as power consumption reduction, data movement avoidance, and real-time processing support needed by advanced automotive applications like ADAS, autonomous driving, and intelligent battery management. Building on existing hardware/software co-design methodologies and the latest developments in PIM architectures, the new processing framework that integrates PIM through a refined hardware/software co-design methodology to effectively deploy PIM within the strict constraints of safety-critical and resource-limited automotive environments. This contribution remains conceptual and does not include experimental validation or modeling.

Future work will focus on validating the concept through detailed simulation using cycle-accurate architectural simulators such as gem5, DRAMSim3, or PIMEval. The evaluation process will include metrics such as latency, energy consumption, memory traffic reduction, and worst-case execution time (WCET). Representative automotive workloads, such as ADAS perception pipelines, radar signal processing, and battery management algorithms, will be ported to PIM-based execution models.

To sum up, through this work we established a foundation

for real-time and safety-critical use of PIM technologies in the context of hardware/software co-design of data-intensive automotive systems to improve performance and power efficiency.

VII. Declaration On Generative AI

During the preparation of this manuscript, the authors utilized Gemini and ChatGPT to support drafting and language refinement, but the development of scientific ideas, arguments, results, and conclusions was carried out only by human authors. The authors take full responsibility for the final version of the manuscript.

References

[1] Y. K. Chein, M. Fariq, F. Rahman, N. Kamsani, and F. Rokhani, "Hardware software partitioning of crankshaft function in engine control units using fpga-based testing," in *Proceedings of the IEEE 15th Student Conference on Research and Development (SCOReD)*, 2017, pp. 131–136.

[2] M. D. Pesé, K. Schmidt, and H. Zweck, "Hardware/software co-design of an automotive embedded firewall," *SAE Technical Paper*, 2017.

[3] B. Poudel and A. Munir, "Design and evaluation of a reconfigurable ecu architecture for secure and dependable automotive cps," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 1, pp. 235–252, 2018.

[4] X. Zou, S. Xu, X. Chen, L. Yan, and Y. Han, "Breaking the von neumann bottleneck: architecture-level processing-in-memory technology," *Science China Information Sciences*, vol. 64, no. 6, p. 160404, 2021.

[5] K. Asifuzzaman, N. R. Miniskar, A. R. Young, F. Liu, and J. S. Vetter, "A survey on processing-in-memory techniques: Advances and challenges," *Memories-Materials, Devices, Circuits and Systems*, vol. 4, p. 100022, 2023.

[6] M. Zhou, W. Xu, J. Kang, and T. Rosing, "TransPIM: A memory-based acceleration via software-hardware co-design for transformer," in *Proceedings of the IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, 2022, pp. 1071–1085.

[7] M. A. Jabu, A. A. Alugongo, and N. Z. Nkomo, "A critical review on the use of artificial intelligence in the automotive industry," *International Journal of Engineering Trends and Technology*, vol. 73, no. 6, pp. 450–456, 2025.

[8] S. O. Laghzal and A. EL OUADI, "Mapping artificial intelligence research in automotive manufacturing: A bibliometric study," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 16, no. 11, pp. 923–934, 2025.

[9] A. Shaout, A. H. El-Mousa, and K. Mattar, "Specification and modeling of hw/sw co-design for heterogeneous embedded systems," in *Proceedings of the World Congress on Engineering*, vol. 1, 2009.

[10] K. V. Patel, M. V. Shah, P. P. Prajapati, and A. J. Kshatriya, "Design and implementation of effective elliptic curve cryptography accelerator using hardware/software co-design on zynq board," *International Journal of Engineering Trends and Technology*, vol. 70, no. 8, pp. 327–335, 2022.

[11] A. Abdallah, E. M. Feron, G. Hellestrand, P. Koopman, and M. Wolf, "Hardware/software codesign of aerospace and automotive systems," *Proceedings of the IEEE*, vol. 98, no. 4, pp. 584–602, 2010.

[12] J. Axelsson, "Hw/sw codesign for automotive applications: Challenges on the architecture level," in *Proceedings of the Fourth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC)*, 2001, pp. 123–126.

[13] H. Askaripoor, M. Hashemi Farzaneh, and A. Knoll, "E/e architecture synthesis: Challenges and technologies," *Electronics*, vol. 11, no. 4, p. 518, 2022.

[14] S. Sinha, "Towards a centralized multicore automotive system," Ph.D. dissertation, Boston University, Boston, MA, USA, 2022.

[15] S. Sinha, A. Farrukh, and R. West, "ModelMap: A model-based multi-domain application framework for centralized automotive systems," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, 2022, pp. 1–9.

[16] S. Khandelwal and S. Shreejith, "A lightweight fpga-based ids-ecu architecture for automotive can," in *Proceedings of the IEEE International Conference on Field-Programmable Technology (ICFPT)*, 2022, pp. 1–9.

[17] K. Cho, J. Kim, D. Y. Choi, Y. H. Yoon, J. H. Oh, and S. E. Lee, "An fpga-based ecu for remote reconfiguration in automotive systems," *Micromachines*, vol. 12, no. 11, p. 1309, 2021.

[18] M. Baleani, M. Conti, A. Ferrari, and A. Sangiovanni-Vincentelli, "Hw/sw codesign of a multiple injection driver automotive subsystem using a configurable system-on-chip," in *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2002, pp. 87–92.

[19] H. E. Yantır, A. M. Eltawil, and K. N. Salama, "A hardware/software co-design methodology for in-memory processors," *Journal of Parallel and Distributed Computing*, vol. 161, pp. 63–71, 2022.

[20] L. Yavits, "Will computing in memory become a new dawn of associative processors?" *Elsevier*, 2023.

[21] Y. Huang, L. Zheng, P. Yao, J. Zhao, X. Liao, H. Jin, and J. Xue, "A heterogeneous pim hardware-software co-design for energy-efficient graph processing," in *Proceedings of the IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2020, pp. 684–695.

[22] X. Yang, Y. Hou, and H. He, "A processing-in-memory architecture programming paradigm for wireless internet-of-things applications," *Sensors*, vol. 19, no. 1, p. 140, 2019.

[23] J. Joo, H. Kim, H. Han, E. G. Im, and S. Kang, "Highly parallel regular expression matching using a real processing-in-memory system," *IEEE Access*, vol. 13, pp. 18 937–18 951, 2025.

[24] A. F. Laguna, M. M. Sharifi, A. Kazemi, X. Yin, M. Niemier, and X. S. Hu, "Hardware-software co-design of an in-memory transformer network accelerator," *Frontiers in Electronics*, vol. 3, p. 847069, 2022.

[25] B. H. Malik, M. Maryam, M. Khalid, J. Khlaid, N. U. Rehman, S. I. Sajjad, and et al., "Fast and efficient in-memory big data processing," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 10, no. 5, pp. 517–522, 2019.

[26] D. Gajaria, K. A. Gomez, and T. Adegbija, "STT-RAM-based hierarchical in-memory computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 35, no. 9, pp. 1615–1629, 2024.

[27] M. Dessokey, S. M. Saif, H. Eldeeb, S. Salem, and E. Saad, "Importance of memory management layer in big data architecture," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 5, pp. 460–466, 2022.

[28] D. Wang and S. Ganesan, "Automotive domain controller," in *Proceedings of the International Conference on Computing and Information Technology (ICCIT)*, 2020, pp. 1–5.

A Feasibility Study of Explainable Machine Learning on Small-Scale Postoperative Voice Data

Noura Haddou[1], Najlae Idrissi[2], Sofia Ben Jebara[3]
Faculty of Science and Technology-Intelligence Data and Computing Team,
Sultan Moulay Slimane University, Beni Mellal, Morocco[1,2]
Research Laboratory COSIM-Higher School of Communications of Tunis, University of Carthage, Ariana, 2088, Tunisia[3]

***Abstract*—Voice dysfunction is a common complication following thyroid surgery. However, the application of explainable machine learning for predicting postoperative voice recovery remains largely unexplored. Therefore, an investigation was done to examine voice recovery based on acoustic, objective, and glottal features. Voice recordings were collected from female patients before surgery and one month after surgery. Acoustic and glottal parameters, including Quasi Open Quotient, Speed Quotient, age, and others, were automatically extracted from the recordings. Random Forest, Support Vector Machines, and Logistic Regression with Sequential Feature Selection were applied to examine model behavior and identify feature importance. Model stability and interpretability were evaluated across cross-validation folds. Performance metrics varied over folds, highlighting the exploratory and statistically fragile nature of predictions in small datasets. SHAP (SHapley Additive exPlanations) analysis revealed variability in feature contributions, emphasizing the need for cautious interpretation and detailed methodological reporting. Our findings provide preliminary guidance for applying explainable machine learning to small biomedical datasets. They demonstrate the importance of careful methodological design.**

***Keywords*—XAI; explainable AI; SHAP; glottal features; SVM; thyroidectomy; voice recovery**

I. Introduction

Thyroidectomy is a surgical intervention performed to remove a portion or the entire thyroid gland. It can be performed to treat malignant or benign thyroid diseases [1]. This surgery is mostly followed by a change in voice quality, reported in approximately 30% to 80% of patients [2], [3]. This change is described as a disorder characterized by an impaired pitch, volume, vocal effort, or voice quality, which alters communication or voice-related quality of life [4].

To detect pathological voices, both subjective and objective assessment methods are commonly used [5], [6]. The subjective voice assessment method includes an auditory perceptual assessment and visual examination of the vocal cords in the hospital [5], [7]. However, the diagnosis based on this type of evaluation is difficult because it relies on the experience of doctors, and there is also a high possibility of clinical errors [8]. In contrast, the objective assessment is based on non-invasive computer analysis of acoustic signals to identify pathological voice, which may not even be audible to a human being [5].

Despite these advances, speech pathology remains an underexplored area, and the field has not yet received significant attention from researchers. This creates a significant need for further research and improved diagnostic tools for voice-related disorders [9]. Thyroid cancer is one of these diseases. It is the second most common cancer for both men and women in the 15 to 34 age group and the most common cancer in women [2], [10].

Regrettably, most existing works have focused on voice analysis or classification tasks. In particular, research has aimed to identify acoustic parameters associated with voice changes post-surgery and to develop computer-based diagnostic systems using a combination of these features. These classification techniques have been widely used to build automated detection systems. However, they are not commonly applied to predict voice quality or recovery following thyroidectomy.

In this pilot research, we propose a ML-based system that uses voice recordings from two stages: pre-operative and one month postoperative. Objective, acoustic and glottal features are used to train multiple classifiers. We also apply Sequential feature selection (SFS) for feature selection to improve prediction performance. Finally, we integrate SHapley Additive exPlanations (SHAP) to interpret the decision-making process of the best-performing model. The aim was to address a binary classification problem to differentiate between recovery and non-recovery voice. The main contributions of this research are as follows:

- We implement a leakage-safe, group-stratified machine learning pipeline suitable for extremely small biomedical voice datasets.
- We demonstrate the use of group-stratified nested cross-validation combined with sequential feature selection to reduce evaluation bias in longitudinal voice data.
- We provide an empirical analysis of model instability and performance variance under small-sample regimes using bootstrapped confidence intervals and learning curves.
- We investigate the limitations and reliability of SHAP-based explainability when applied to small-scale tabular datasets.

II. Related Work

Recently, scientists focused on building several systems for detecting pathological voices using machine learning algorithms. These approaches can be classified into two primary techniques: traditional pipeline methods and end-to-end techniques [11]. End-to-end approaches utilize the original raw

voice or transformed speech, such as the spectrogram in its 2D form as input for the classification model, and employ deep learning techniques to generate target labels [12].

On the other hand, the traditional pipeline approaches involve two stages: firstly, the extraction of relevant parameters from a speech signal, and secondly, using these extracted features to train the model and classify the labeled output according to the intended purpose (normal/pathology voice, type of pathology, etc.) [12]. Most studies have used a set of objective and subjective parameters to realize these automatic systems. These measurements contain jitter, glottal parameters [12], etc.

To address this, some researchers have proposed new detection frameworks using ML or deep learning algorithms to forecast voice recovery after thyroid surgery. For example, Lee *et al.* [13] published a study to predict patient voice recovery after three months of postoperative spectrograms with a deep neural network algorithm. They initially proposed an approach that employed pre-operative and first postoperative patients' voice recordings and GRBAS scores to predict the patient's voice GRBAS scores after three months post-surgery. The utilization of GRBAS scores primarily aimed to characterize vocal quality by evaluating the degree of voice impairment during each session. In their study, 114 patients were followed. All patients who have undergone surgery for thyroid cancer. The participants were taken before surgery, two weeks, and three months after the operation. The proposed method has shown that long-term voice disorders can be predicted using pre- and postoperative spectrograms [13].

Another study by Kurt *et al.* [14], focused on developing a detection system using ML techniques to determine patients at risk of developing vocal cord palsy before thyroidectomy [14]. They included a sample of 1039 patients who underwent thyroidectomy in their research. The authors used a set of variables based on clinician reports, which can divided into two categories: continuous variables such as age, tumor size, white blood cell count, etc., and categorical variables including sex of patient, type of resection, presence of comorbidity, and so on. They achieved an accuracy of 100%. However, although Lee *et al.* [13] used spectrograms and GRBAS scores (Grade, Roughness, Breathiness, Asthenia, Strain) in their deep learning approach, this method has not yet been evaluated with traditional machine learning techniques or direct acoustic and glottal parameters. Most existing studies focus on evaluating voice quality before and after surgery using statistical tests [15], [16], [17].

In terms of classification, the majority aim to develop systems that distinguish between pathological and normal voices. To our knowledge, no study has yet explored ML-based prediction of post-thyroidectomy voice recovery using glottal, objective, and acoustic features with SHAP explainability.

III. Materials and Methods

In this section, we outline the methodology and materials used in the present study. Fig. 1 presents the proposed framework for predicting voice recovery after thyroidectomy. The system comprises different steps, including data collection (longitudinal audio recordings), pre-processing, feature extraction, classification, and interpretability analysis using the SHAP method. Classification was performed using the SFS algorithm.

A. Data Collection

Twenty-five females (aged 17-66 years, SD = 14.44) were included in this study. They underwent either partial or total thyroidectomy for malignant or benign thyroid cancer. Among them, 12 underwent partial thyroidectomy, and 13 underwent total thyroidectomy. All participants underwent thyroidectomy for benign thyroid, except for five who underwent surgery for malignant thyroid. The subjects did not have any vocal cord damage or history of alcohol before thyroid surgery. Moreover, no treatments were administered to the patients between the first day and one month after surgery, except for those who experienced vocal fold paralysis on the first day. Voice therapy is recommended in cases of vocal cord paralysis. Thyroidectomy was performed by various surgeons using a conventional cervical method, without the assistance of intraoperative neuromonitoring (IONM) for the identification of either the external branch of the superior laryngeal nerve (EBSLN) or the recurrent laryngeal nerve (RLN). Each patient underwent a laryngeal endoscopic examination conducted by a senior physician or a trained resident during quiet phonation and breathing. Voice recordings were acquired using a microphone integrated into the endoscopy column, positioned approximately 20 cm from the patient's lips. Patients sustained the French vowel /i/ at pre-surgery, and 30 days post-surgery.

B. Data Pre-processing

The duration of audio recordings varied among participants because they were instructed to pronounce the vowel for as long as possible at a comfortable volume and pitch. To ensure consistency across samples, a 3-second segment was extracted from the middle of each recording. This segment length was chosen after initially extracting 3-second segments from the beginning, middle, and end of each recording. Statistical analysis using ANOVA showed no significant effect of segment position on the acoustic parameters. Therefore, the middle 3-second segment was selected, as it represents the most stable portion of the sustained vowel. Acoustic and glottal parameters were then extracted from this segment for each participant.

C. Feature Extraction

In this section, we present the features used in the current study. They are classified into two categories: acoustic features and glottal features. In this work, we used only the parameters extracted on the day before the operation and excluded the other features. Additionally, the VHI-10 parameter was excluded from the training process to avoid any data leakage.

1) Acoustic parameters: From the vocal samples, we extracted the mean values of the fundamental frequency (F0). F0 presents the vocal fold vibration frequency, measured in Hertz, which is defined by the duration of each period T_i. The study also includes information on age, type of operation (partial or total), the nature of the thyroid (malignant or benign), and the Voice Handicap Index (VHI-10). The VHI-10 is a metric employed to record the patient's perception of handicap or impairment due to a speech disease, where scores above 11 are defined as pathology and indicate vocal disability. To further

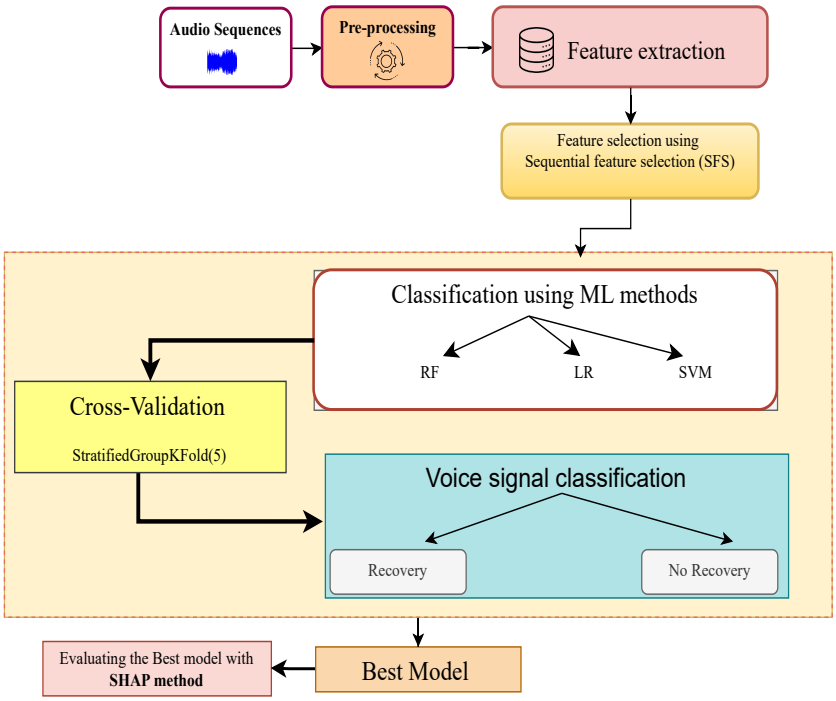


Fig. 1. Proposed method for predicting voice recovery stages following thyroidectomy.

improve the analysis, glottal flow features were used due to their demonstrated effectiveness in capturing the vocal effects of thyroidectomy [18].

2) Glottal parameters: In the literature, many techniques have been developed to obtain the glottal signal features. Among these methods, we find the Iterative Adaptive Inverse Filtering (IAIF) method [19], which estimates the glottal signal from the input speech signal through a two-step iterative process. Initially, the method estimates the glottal waveform and the vocal tract transfer function [18], [20]. It then refines these estimates to improve the performance of glottal signal estimation [18], [20].

In our work, we use the Pitch-Synchronous Iterative Adaptive Inverse Filtering (PSIAIF) method [21] because of its ease of implementation. PSIAIF uses the IAIF method to extract glottal flow from the speech signal. Its main goal is to eliminate the influence of glottal excitation from the speech spectrum of the original signal [22]. The method proceeds by first estimating the vocal fold model using Linear Prediction Coefficients (LPC), and then obtaining the glottal flow through inverse filtering [22]. PSIAIF incorporates pitch-synchronous markers on the glottal waveform, which reduces the effect of formant ripple in comparison to IAIF [22]. This method has been implemented in the MATLAB toolbox Aparat [23]. Below, we provide a summary of the features that were extracted:

- The open quotient (OQ): is the ratio of the duration of the vocal fold open phase to the total duration of a glottal cycle [24].
- OQ1: Open quotient calculated from the primary opening of the glottal flow.
- OQ2: Open quotient calculated from the secondary opening of the glottal flow.
- The Speed Quotient (SQ): is the ratio of the duration of the opening phase to the duration of the closing phase [24]. **SQ1** and **SQ2** represent the speed quotients calculated from the primary and secondary openings, respectively.
- The normalized amplitude quotient (NAQ): is defined as the ratio of the amplitude quotient to the total duration of the glottal pulse [24].

To extract these features we used the methodology described in Alku (1992) [21]. The LPC analysis orders for the vocal tract and glottal source were set based on the sampling frequency, approximately $2 \times \text{round}(f_s/2000) + 4$ and $2 \times \text{round}(f_s/4000)$, respectively.

A leaky integration coefficient of 0.99 was used to model lip radiation. A single pass of a high-pass FIR filter (cutoff frequencies at 40 Hz stopband and 70 Hz passband) was applied prior to inverse filtering to remove low-frequency noise. The analysis was performed on frames windowed with a Hanning window. The audio signal was processed using its original sampling rate.

D. Classification Techniques

In the current study, a change in VHI-10 (ΔVHI_{10}) was calculated for each patient as:

$$\Delta \text{VHI}_{10} = \text{VHI}_{10,1\text{ month}} - \text{VHI}_{10,\text{pre-op}}$$

A negative ΔVHI_{10} (< 0) indicated improvement and was classified as *recovered*, while $\Delta \text{VHI}_{10} \geq 0$ was classified as *not recovered*.

Given the limited dataset size, we employed cross-validation with stratification to evaluate the models. Specifically, we used StratifiedGroupKFold with k = 5. This value was selected after testing k values from 2 to 7, with the best performance achieved at k = 5. To avoid data leakage, groups were defined by patient ID, ensuring that recordings from the same patient did not appear in both training and testing sets. The models were evaluated using a pipeline that applied MinMax scaling to the input features, which normalizes each feature by subtracting the minimum value and dividing by the range (max - min). The models were evaluated using the Sequential Feature Selection (SFS) method. To assess model stability and avoid data leakage, a nested cross-validation procedure was applied inside SFS during the feature selection phase. In addition, a bootstrapped validation with 10000 resamples was performed to calculate the corresponding confidence intervals (CIs) for model performance metrics.

For classification, various machine learning techniques, including SVM, RF, and Logistic Regression (LR) were used. The SVM was chosen as the representative algorithm due to its stability with limited samples. RF and LR were evaluated for comparison and to verify that performance differences were consistent; results are reported for completeness.

E. Hyperparameters

For hyperparameter tuning, GridSearchCV method was applied to each algorithm using the same StratifiedGroupKFold cross-validation process described above. For the SVM model, parameters such as the regularization parameter C (ranging from 1 to 50), kernel type (linear, RBF, polynomial, sigmoid), and gamma (auto or scale) were optimized. RF tuning included the number of estimators (1 to 40), maximum tree depth (1 to 51), and the criterion (Gini or entropy). The selected parameters were then employed in this study. The best hyperparameters were then used for each model. The final hyperparameters for each model were:

- **SVM:** $C = 1$, gamma = `scale`, kernel = `rbf`
- **LR:** max_iter = 1000
- **RF:** max_depth = 2, criterion=`gini`, n_estimators = 3

All analysis were conducted in Python with scikit-learn version 1.6.1, pandas, and matplotlib. The experiments were conducted on an HP laptop with an Intel Core i7-1165G7 processor (2.80 GHz) and 32 GB of RAM.

F. Shapley Additive Explanations (SHAP)

Recently, many studies have applied Explainable Artificial Intelligence (XAI) techniques to understand better the decision-making processes of machine learning models [25]. In this field, researchers have introduced numerous methods to explain how model features, inputs, or components influence model outputs [26]. In particular, these include Gradient-weighted Class Activation Mapping (Grad-CAM), Local Interpretable Model-Agnostic Explanations (LIME), and SHAP [26]. However, LIME and SHAP are the most widely used because they can be applied to both tabular data and image data. SHAP, introduced by Lundberg and Lee [27], explains model results using concepts from cooperative game theory. SHAP can be applied for both local and global explanations [27]. Local explanations provide insight into individual predictions, while global explanations summarize feature influence across the entire dataset. Shapley values identify the contribution of each feature to the prediction produced by the model.

IV. Results

In this section, we present an analysis of the features and show the results of the classification experiments. In addition, it includes an interpretability analysis of the best-performing model using the SHAP framework.

A. Feature Distribution Analysis

A violin plot is a powerful visualization tool used to display the distribution of data and its probability density function. It summarizes key statistical values, including the median and quartiles, while also showing the full distribution of the dataset. In this study, we present violin plots for the numeric acoustic parameters, grouped by recovery status (Recovery vs. No Recovery).

In Fig. 2, the median value of MeanF0 in the No-Recovery group (171.644 Hz), with data points largely concentrated between 150 Hz and 250 Hz, is slightly higher than that of the Recovery group (168.300 Hz). Similarly, for all other parameters, the No-Recovery group consistently shows higher values compared to the Recovery group. This may suggest that higher values are associated with poorer voice recovery. However, this observation remains preliminary and requires further confirmation.

B. Classifier Performance

Table I summarizes the classification performances for the SVM, LR, and RF algorithms. The experiments reveal that SVM outperforms the other models in terms of accuracy. It reaches an accuracy of 76% with a standard deviation (SD) of 0.09, a recall of 87%, an F1-score of 80%, a precision of 77%, and an AUC score of 70%. This suggests their ability to identify the positive class. However, the SD value indicates that the model's performance is unstable between folds. In contrast, the RF model reaches the highest precision of 77%. However, its lower recall (73%) and accuracy (73%) indicate a lower ability to detect all true positives. Also, its performance is more variable. The lower results were obtained using the LR model, except for AUC (80%), which was higher than that of the SVM and RF methods.

In addition, the CI interval for SVM (accuracy: [0.68, 0.80]) shows moderate stability, indicating relatively consistent performance across folds. However, the wide confidence intervals for all models show that their performance can change significantly with different data.

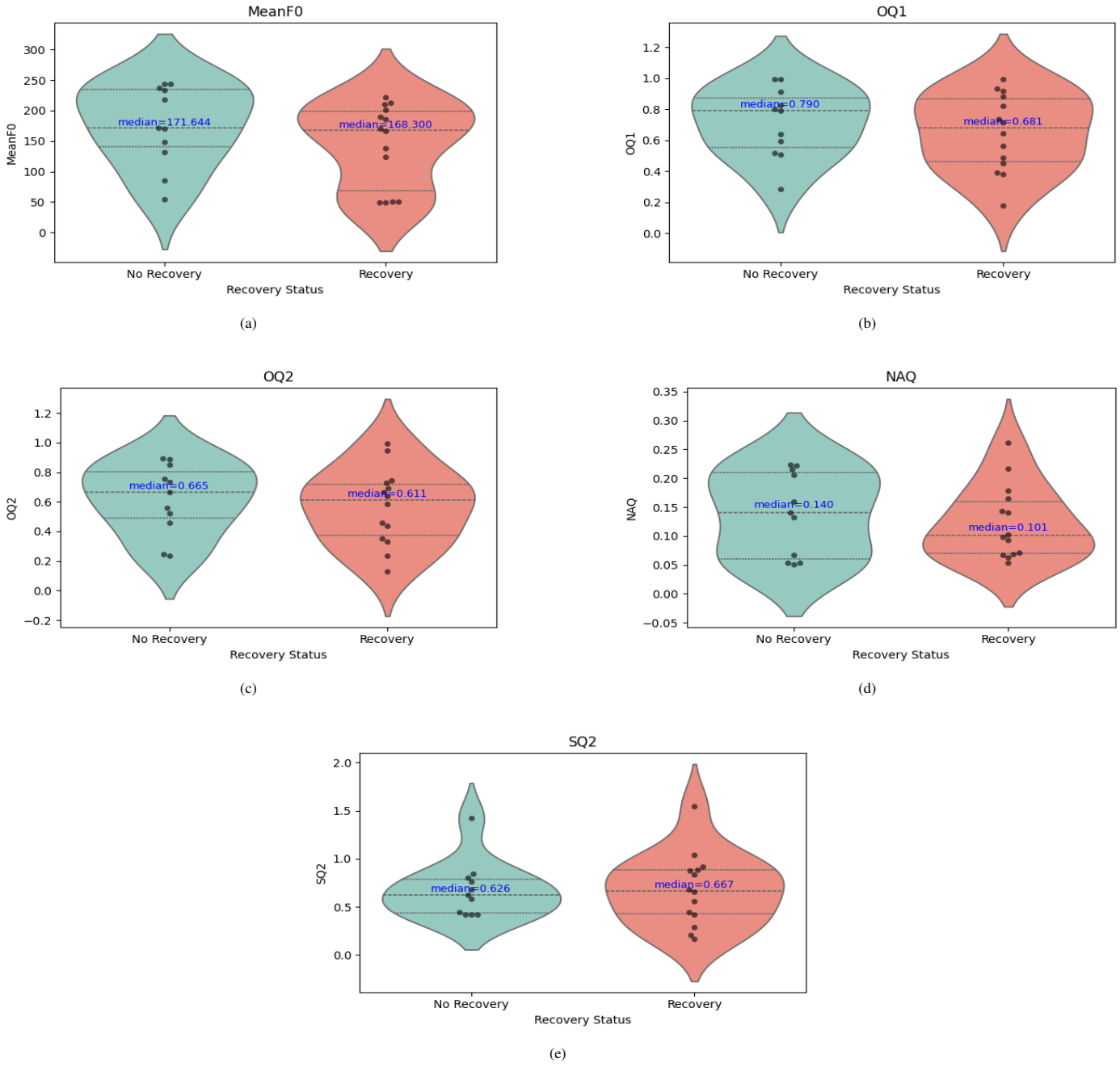


Fig. 2. Violin plots of acoustic and glottal features in the Recovery and No-Recovery groups: a) Mean fundamental frequency (MeanF0), b) Primary Open Quotient (OQ1), c) Secondary Open Quotient (OQ2), d) Normalized Amplitude Quotient (NAQ), and e) Secondary Speed Quotient (SQ2).

TABLE I. CROSS-VALIDATED PERFORMANCE METRICS USING SFS SELECTION METHOD (MEAN ± SD) [95% CI]

Algorithms	Accuracy	Precision	Recall	F1	AUC	Features
SVM	**0.76** ± 0.09 [0.68, 0.80]	**0.77** ± 0.14 [0.68, 0.88]	**0.87** ± 0.18 [0.73, 1.00]	**0.80** ± 0.08 [0.73, 0.85]	0.70 ± 0.15 [0.58, 0.82]	Type_CN, Type
RF	0.72 ± 0.23 [0.56, 0.88]	0.77 ± 0.22 [0.60, 0.93]	0.73 ± 0.28 [0.53, 0.93]	0.73 ± 0.22 [0.56, 0.89]	0.78 ± 0.23 [0.60, 0.95]	Age, Type_CN, Type, NAQ
LR	0.64 ± 0.17 [0.52, 0.76]	0.69 ± 0.19 [0.57, 0.85]	0.73 ± 0.28 [0.53, 0.93]	0.68 ± 0.17 [0.53, 0.79]	**0.80** ± 0.24 [0.62, 0.98]	Age, Type_CN, Type

C. Statistical Comparison

To verify whether the differences were statistically significant among the three classifiers, we conducted a Wilcoxon signed-rank test based on the bootstrapped F1-scores. The obtained results indicated that SVM consistently outperformed both RF and LR, with mean differences of 0.063 (95% CI [0.061, 0.065]) for SVM vs RF and 0.113 (95% CI [0.111, 0.114]) for SVM vs LR. In addition, RF achieved higher F1-scores than LR, with a mean difference of 0.050 (95% CI [0.049, 0.051]). Effect sizes were moderate to large (r = -0.55 for SVM vs RF, $r = -0.87$ for SVM vs. LR, $r = -0.87$ for RF vs. LR), and all comparisons were statistically significant ($p < 0.001$). These findings suggest that SVM demonstrates superior performance in distinguishing between classes on the dataset analyzed.

D. Model Interpretation and Error Analysis

Fig. 3, Fig. 4, and Fig. 5 present the confusion matrix, learning curve, and ROC curve, respectively, for the SVM classifier. Overall, the model demonstrated a strong ability to differentiate between the classes. It performed particularly well in identifying recovery recordings, correctly classifying 12 out of 14 samples. The no-recovery class was classified moderately well, with 7 correct predictions, though some samples were misclassified as recovery.

To better understand the model's behavior, we analyzed the misclassified samples from the SVM model predictions in Table II. The analysis shows that most false positives (FP) (patients predicted as recovered but who did not actually recover) occurred in three patients who underwent partial surgeries for benign conditions. It generally occurs in younger or middle-aged adults (ages 35–53). A false positive was also observed in a patient who had a total thyroidectomy for a malignant thyroid. In contrast, false negatives (patients predicted as not recovered but who actually recovered) were more common among younger patients (ages 31–42) who underwent total surgeries for benign thyroid conditions. These findings suggest that the model's performance is significantly influenced by the type of surgery and the underlying pathology. In addition, ROC analysis supports these findings, with an AUC of 65% achieved for the recovery class. However, due to the small number of misclassified cases, we did not perform statistical tests, as such analyses would not provide reliable or meaningful results.

Dealing with the SVM learning curve (see Fig. 5) displays initial overfitting, with high training accuracy but low validation accuracy. As the training size increases, validation accuracy rises and converges toward the training curve, indicating improved generalization and reduced overfitting.

E. SHAP Results

Fig. 6 presents the SHAP summary plot (mean absolute SHAP values) by class. These values represent the average contribution of each feature to the predictions of the model. In the No recovery class, the most influential variables were Type (partial/total) and Age (patient age), with SHAP values of 0.09 and 0.05, respectively. Other relevant features included Type_CN (benign/malign), OQ1, MeanF0, NAQ, SQ2, and OQ2. In contrast, the Recovery class is mainly associated with

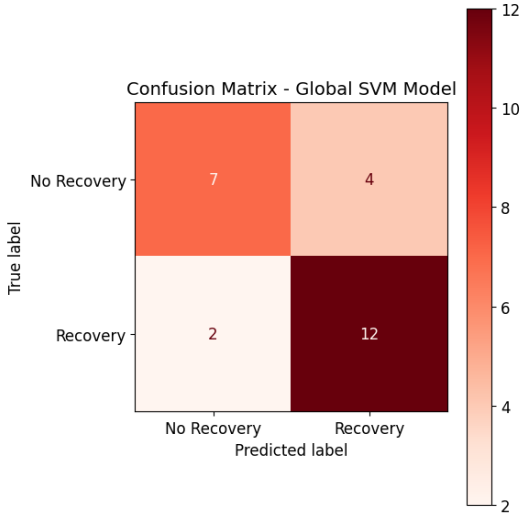


Fig. 3. Confusion matrix for the SVM model predicting post-thyroidectomy voice recovery. The rows represent actual classes and the columns predicted classes.

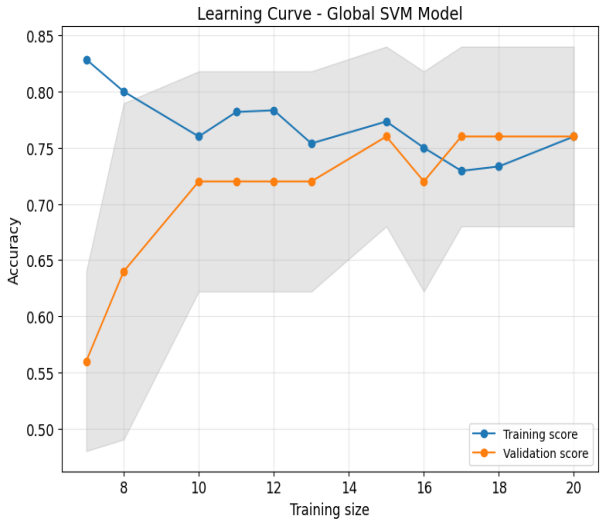


Fig. 4. Learning curve for the SVM method predicting post-thyroidectomy voice recovery. The x-axis represents the number of training samples used, and each point shows the mean accuracy over five cross-validation folds. Shaded regions indicate ±1 standard deviation across folds.

Type (0.08), Type_CN (0.03), and Age (0.03), which have the highest impact.

Fig. 7 presents SHAP-based waterfall plots for three participants selected at random. These plots illustrate the contribution of individual variables to the prediction of the model for each patient. The horizontal axis represents the value predicted by the model, denoted as while each bar illustrates the marginal effect of a specific variable on this prediction. In each plot, red bars indicate positive contributions to the decision, while the blue bars represent a negative contribution. In other words, red means features that increase the model's predicted probability of class, whereas blue bars correspond to features that reduce it.

TABLE II. ERROR ANALYSIS OF MISCLASSIFIED PATIENTS FOR SVM MODEL

Patient_ID	ErrorType	Age	Type	Type_CN	TrueLabel	PredictedLabel
1	False Positive	35	Partial	Benign	0	1
2	False Negative	42	Total	Benign	1	0
3	False Positive	52	Partial	Benign	0	1
4	False Positive	53	Partial	Benign	0	1
5	False Positive	31	Total	Malign	0	1
6	False Negative	31	Total	Benign	1	0

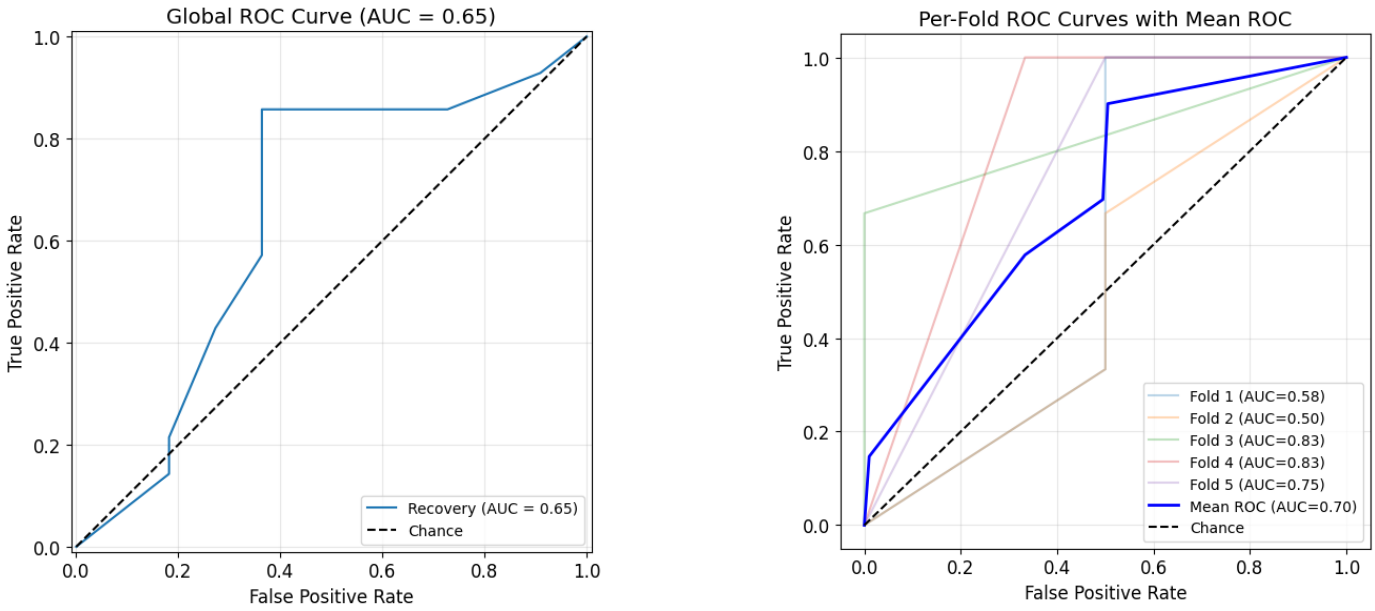


Fig. 5. ROC curve for the SVM method: a) Global ROC curve showing the overall discriminative performance (AUC calculated on all test predictions across folds). b) Per-fold ROC curves with the mean ROC curve (bold line) to illustrate variability across cross-validation folds.

1) Participant 1: is a 35-year-old woman who underwent partial thyroidectomy for a benign condition. For this woman, the predicted score is -0.129 which is below the mean value (-0.004). Type, OQ1, and Age show negative contributions (-0.06, -0.01, and -0.07, respectively), while NAQ contributed positively. Other features had negligible effects.

2) Participant 6: is a 35-year-old woman who underwent total thyroidectomy for a malignant condition. The model predicted a score of 0.016, higher than the baseline value of 0.004. Most features contributed negatively to the prediction, except for Type_CN, MeanF0, and Age, which had positive effects. Among these positive contributors, Type_CN had the largest value (+0.07), in contrast to what was observed for Participant 1.

3) Participant 15: is a 64-year-old woman who underwent a partial thyroidectomy for a benign condition. Most parameters had negligible effects.

V. DISCUSSION

Vocal changes are one of the common problems for subjects undergoing thyroid surgery. In the current work, we predict vocal recovery following thyroidectomy. For this purpose, machine learning techniques (ML) and a set of features are used. In addition, we use various ML algorithms to classify voices into recovery and non-recovery classes. All participants were evaluated before surgery and a month postoperative. This work was designed as a pilot feasibility study rather than a fully powered trial.

The SVM technique achieved the highest performance, with an F1-score of 80%. The SHAP analysis revealed that the most important features contributing to no-recovery predictions were the type of performed operation (partial/total), followed by age, and then the nature of the thyroid (malignant/benign). Various works have also found that speech can be affected by the extent of thyroid surgery [28], [29], [30]. Yılmaz et al. [31] reported that the type of surgical procedure (subtotal thyroidectomy or total), age, histopathologic diagnosis, and sex did not significantly impact vocal quality after thyroidectomy.

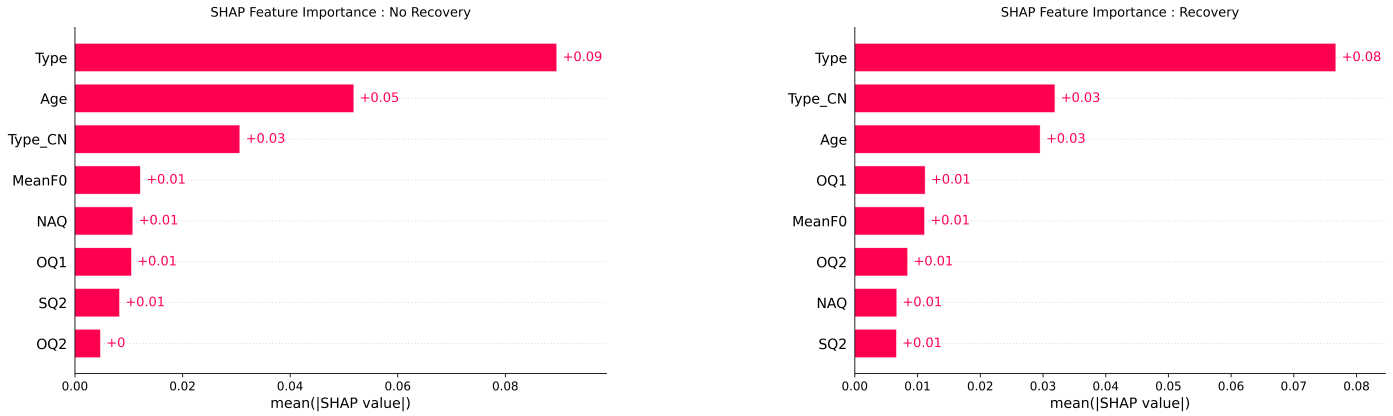
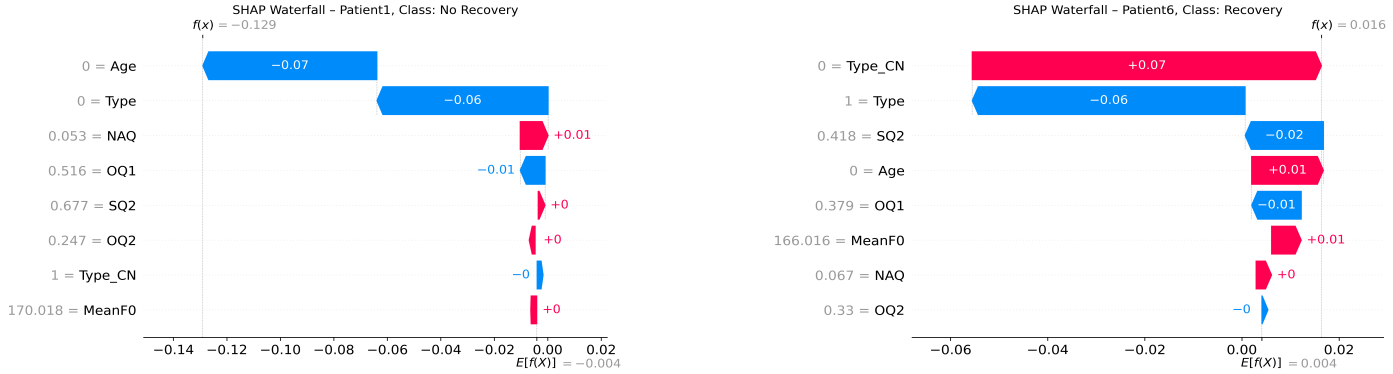
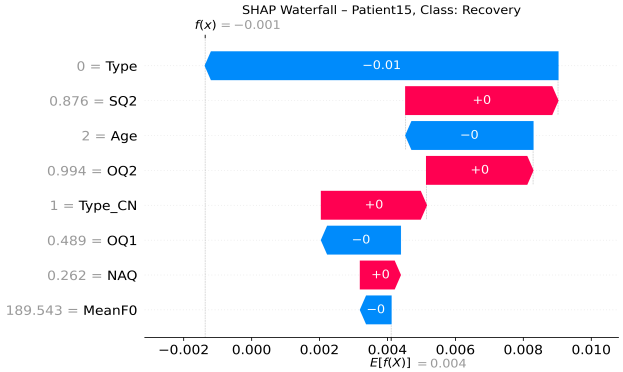


Fig. 6. SHAP summary plot (mean absolute SHAP values) for recovery and no recovery classes.



(a) Participant 1

(b) Participant 6



(c) Participant 15

Fig. 7. Example of waterfall plots for three participants selected randomly from the dataset.

However, they observed that voice changes were associated with tumor type. This may be due to various factors, such as injuries to the Recurrent Laryngeal Nerve (RLN) [32]. In addition, patients' emotions, experiences, and professional occupations may significantly impact these outcomes. Some existing studies have also identified female gender and the volume of the resected gland as risk factors for post-operative speech alterations after total thyroidectomy [28]. Also, a set of glottal features corresponding to different phases of the glottal cycle also contributed to the model's performance. Among these, OQ, and SQ are commonly used time-based parameters, while NAQ represents the shape and timing of the glottal waveform [18].

In our SHAP analysis, the glottal features had a similar impact in predicting both classes. Existing studies showed that in healthy individuals, higher OQ1 and OQ2 values are associated with better voice quality. In contrast, lower values are generally associated with pathological conditions such as nodules and vocal fold paralysis [18]. Moreover, OQ is

inversely correlated with vocal intensity, meaning that lower values correspond to higher vocal intensity [24].

For the NAQ parameter, higher values are indicative of a more symmetrical glottal waveform and are associated with breathy phonation type [33], [34]. OQ and NAQ are also inversely related to vocal intensity [24]. A decrease in NAQ may reflect increased vocal intensity due to a prolonged glottal closure phase [35].

Previous studies have revealed that the SQ2 parameter is lower in healthy voices than in pathological ones. In Addition, decreased values of these parameters have been reported in female patients with heart failure, suggesting their sensitivity to physiological alterations [24]. The glottal parameters are also applied to distinguish between patients with healthy voices and those with vocal fold paralysis [20] or neurological disorders such as Parkinson's disease [36]. These outcomes underline the clinical value of using glottal source features in postoperative voice assessment.

Some recent studies related to this current study have utilized ML techniques to predict thyroid cancer in patients who underwent thyroidectomy. They also achieved promising results in predicting thyroid cancer [14], [13], [37]. Lee *et al.* [13] developed a deep neural network using preoperative and two-week postoperative spectrograms to predict three-month postoperative vocal recovery in 114 patients. Their model achieved high predictive performance, with AUC values of 0.918 for breathiness, 0.735 for asthenia, and 0.894 for grade. In contrast, the present study achieved an AUC of 0.65 for the recovery class using acoustic and glottal features with SVM model on a smaller dataset of 25 patients. The lower performance may be attributed to the smaller sample size and differences in feature representation. Also, our study did not include deep learning models for classification. However, our method provides interpretable predictions using SHAP analysis to identify parameters that most contribute to voice recovery.

In terms of predictive performance, our results show lower accuracy compared with Kurt *et al.* [14] in identifying patients at risk of developing vocal cord palsy before thyroidectomy using ML techniques. However, their study addressed a different clinical problem, focusing on preoperative risk prediction of vocal cord palsy, while the present study predicts postoperative voice recovery. Therefore, direct numerical comparison of accuracy is not appropriate. Although deep learning approaches have demonstrated promising results and the potential to enhance model performance, they were not employed in this study. This decision was primarily due to the limited size of our dataset, which is significantly smaller than those used in comparable studies. Furthermore, this research does not primarily aim to optimize performance measures, but to investigate glottal and acoustic parameters relevant to thyroidectomy. These parameters have previously demonstrated their ability to detect changes in voice after thyroid surgery.

A. Limitations of the Study

Several limitations must be acknowledged. In our study, we employ the PSIAIF method to estimate glottal flow without feature-sensitivity analysis. This method, however, is sensitive to parameter configuration, which can affect the performance of derived features such as OQ and NAQ. In addition, the dataset was relatively small wich may introduce statistical fragility, as model performance and feature importance estimates may be sensitive to minor variations in the data. Although cross-validation was employed to mitigate overfitting, small datasets inherently increase the risk of variability and reduced generalizability. Also, calibration analysis was not performed. Therefore, the agreement between predicted probabilities and actual outcomes remains unassessed. Another limitation is that all participants were female. This approach has reduced the variability of vocal parameters related to physiological differences between the sexes, but it also limits the generalizability of the results to male patients. In fact, male and female voices differ in fundamental frequency and glottal configuration, which can influence these parameters. It is therefore essential to create predictive models that reflect actual clinical diversity. In addition, we did not include external validation because the existing datasets in the literature are private and not publicly accessible. More precisely, no publicly available longitudinal datasets containing preoperative and postoperative voice recordings were identified. As a result, the generalizability of the proposed models to external populations remains unconfirmed. Finally, we used traditional ML models due to their interpretability and suitability for small datasets. However, the deep learning approaches were not explored in this work due to the limited dataset size.

VI. Conclusion

This pilot feasibility study applied machine learning techniques based on different features to predict voice recovery after thyroidectomy. The dataset was collected on the day before surgery and a month after surgery. Preliminary performance varied between models, with the SVM classifier achieving an F1-score of 80%. These findings suggest that machine learning-based prediction is feasible, but not yet conclusive. Future research is encouraged to address the study's limitations through several directions. It is necessary to use a larger, mixed-gender cohort to improve the generalization of the model. In addition, model performance could be improved by applying advanced inverse filtering methods, such as IAIF+, or closed-phase covariance approaches. Another direction is to employ deep learning architectures, which may further improve predictive accuracy. Taken together, these preliminary findings suggest that machine learning may provide a feasible approach for predicting voice recovery after thyroidectomy. However, external validation is required before clinical application.

Declaration of Interest

No competing interests.

Availability of Data

The dataset used in this study cannot be publicly released due to patient privacy regulations and institutional restrictions. However, to support reproducibility, we provide a complete description of the feature extraction process, model architectures, hyperparameters, validation strategy, and evaluation protocol. All experiments were conducted using deterministic random seeds and standard machine learning libraries.

ACKNOWLEDGMENT

The present research has been produced under the Moroccan-Tunisian project titled "Multimodal Detection and Classification of Pathological Voice Disorders". The Ministry of Higher Education and Scientific Research in Tunisia and the Ministry of Higher Education, Scientific Research and Innovation in Morocco sponsor it. We extend our gratitude to Ilhem Charfeddine, Malek Mnejja and, Mariam Ben Ayed from the Department of Otorhinolaryngology-Head and Neck Surgery at Habib Bourguiba University Hospital, Sfax, Tunisia, for their valuable contributions to data collection and for providing essential information on the study participants.

ETHICAL APPROVAL

We conducted our study after approval from the Ethical Committee of the Medical Institution. All patients were informed about the study protocol. Signed consent was obtained from each participant to confirm their agreement to participate. They were informed that their anonymous recording data would be used for acoustic and video analysis.

REFERENCES

[1] K. Van Lierde, E. D'haeseleer, F. L. Wuyts, N. Baudonck, L. Bernaert, and H. Vermeersch, "Impact of thyroidectomy without laryngeal nerve injury on vocal quality characteristics: an objective multiparameter approach," *The Laryngoscope*, vol. 120, no. 2, pp. 338–345, 2010.

[2] C. H. Ryu, S. J. Lee, J.-G. Cho, I. J. Choi, Y. S. Choi, Y. T. Hong, S. Y. Jung, J. W. Kim, D. Y. Lee, D. K. Lee *et al.*, "Care and management of voice change in thyroid surgery: Korean society of laryngology, phoniatrics and logopedics clinical practice guideline," *Clinical and Experimental Otorhinolaryngology*, vol. 15, no. 1, pp. 24–48, 2022.

[3] I. de Pedro Netto, A. Fae, J. G. Vartanian, A. P. B. Barros, L. M. Correia, R. N. Toledo, J. R. G. Testa, I. N. Nishimoto, L. P. Kowalski, and E. C.-d. Angelis, "Voice and vocal self-assessment after thyroidectomy," *Head & neck*, vol. 28, no. 12, pp. 1106–1114, 2006.

[4] S. R. Schwartz, S. M. Cohen, S. H. Dailey, R. M. Rosenfeld, E. S. Deutsch, M. B. Gillespie, E. Granieri, E. R. Hapner, C. E. Kimball, H. J. Krouse *et al.*, "Clinical practice guideline: hoarseness (dysphonia)," *Otolaryngology–Head and Neck Surgery*, vol. 141, no. 1_suppl, pp. 1–31, 2009.

[5] P. Harar, Z. Galaz, J. B. Alonso-Hernandez, J. Mekyska, R. Burget, and Z. Smekal, "Towards robust voice pathology detection: Investigation of supervised deep learning, gradient boosting, and anomaly detection approaches across four databases," *Neural Computing and Applications*, vol. 32, pp. 15 747–15 757, 2020.

[6] D. D. Mehta and R. E. Hillman, "Voice assessment: updates on perceptual, acoustic, aerodynamic, and endoscopic imaging methods," *Current opinion in otolaryngology & head and neck surgery*, vol. 16, no. 3, p. 211, 2008.

[7] J. Oates, "Auditory-perceptual evaluation of disordered voice quality: pros, cons and future directions," *Folia Phoniatrica et Logopaedica*, vol. 61, no. 1, pp. 49–56, 2009.

[8] M. Mizuta, C. Abe, E. Taguchi, T. Takeue, H. Tamaki, and T. Haji, "Validation of cepstral acoustic analysis for normal and pathological voice in the japanese language," *Journal of Voice*, vol. 36, no. 6, pp. 770–776, 2022.

[9] F. T. Al-Dhief, N. M. A. Latiff, N. N. N. A. Malik, N. S. Salim, M. M. Baki, M. A. A. Albadr, and M. A. Mohammed, "A survey of voice pathology surveillance systems based on internet of things and machine learning algorithms," *IEEE Access*, vol. 8, pp. 64 514–64 533, 2020.

[10] K.-W. Jung, Y.-J. Won, H.-J. Kong, and E. S. Lee, "Cancer statistics in korea: incidence, mortality, survival, and prevalence in 2016," *Cancer research and treatment: official journal of Korean Cancer Association*, vol. 51, no. 2, pp. 417–430, 2019.

[11] M. K. Reddy and P. Alku, "A comparison of cepstral features in the detection of pathological voices by varying the input and filterbank of the cepstrum computation," *IEEE Access*, vol. 9, pp. 135 953–135 963, 2021.

[12] N. Narendra and P. Alku, "Glottal source information for pathological voice detection," *IEEE Access*, vol. 8, pp. 67 745–67 755, 2020.

[13] J. H. Lee, C. Y. Lee, J. S. Eom, M. Pak, H. S. Jeong, and H. Y. Son, "Predictions for three-month postoperative vocal recovery after thyroid surgery from spectrograms with deep neural network," *Sensors*, vol. 22, no. 17, p. 6387, 2022.

[14] B. Kurt, İ. B. Kırkbir, T. Kurt, A. Güner, and M. Uluşahin, "A novel computer based risk prediction model for vocal cord palsy before thyroidectomy," *Computer Methods and Programs in Biomedicine*, vol. 236, p. 107563, 2023.

[15] L. Soylu, S. Ozbas, H. Y. Uslu, and S. Kocak, "The evaluation of the causes of subjective voice disturbances after thyroid surgery," *The American journal of surgery*, vol. 194, no. 3, pp. 317–322, 2007.

[16] F. Debruyne, F. Ostyn, P. Delaere, and W. Wellens, "Acoustic analysis of the speaking voice after thyroidectomy," *Journal of Voice*, vol. 11, no. 4, pp. 479–482, 1997.

[17] A. Stojadinovic, L. R. Henry, R. S. Howard, J. Gurevich-Uvena, M. J. Makashay, G. L. Coppit, C. D. Shriver, and N. P. Solomon, "Prospective trial of voice outcomes after thyroidectomy: evaluation of patient-reported and clinician-determined voice assessments in identifying postthyroidectomy dysphonia," *Surgery*, vol. 143, no. 6, pp. 732–742, 2008.

[18] M. Mnejja, S. B. Jebara, M. B. Ayed, S. Ayadi, O. Walha, B. Hammami, and I. Charfeddine, "Glottal features in vocal assessment following thyroidectomy," *Journal of Voice*, 2024.

[19] G. P. Kafentzis, Y. Stylianou, and P. Alku, "Glottal inverse filtering using stabilised weighted linear prediction," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011, pp. 5408–5411.

[20] A. Bedoui and S. B. Jebara, "On the use of opening phase slopes of the glottal signal to characterize unilateral vocal folds paralysis," in *2016 International Symposium on Signal, Image, Video and Communications (ISIVC)*, 2016, pp. 41–46.

[21] P. Alku, "Glottal wave analysis with pitch synchronous iterative adaptive inverse filtering," *Speech communication*, vol. 11, no. 2-3, pp. 109–118, 1992.

[22] X. Yao, W. Bai, Y. Ren, X. Liu, and Z. Hui, "Exploration of glottal characteristics and the vocal folds behavior for the speech under emotion," *Neurocomputing*, vol. 410, pp. 328–341, 2020.

[23] "Software aparat," November 15, 2022. [Online]. Available: http://aparat.sourceforge.net/index.php/

[24] M. Kohler, M. M. Vellasco, E. Cataldo *et al.*, "Analysis and classification of voice pathologies using glottal signal parameters," *Journal of Voice*, vol. 30, no. 5, pp. 549–556, 2016.

[25] R. Jegan and R. Jayagowri, "Voice pathology detection using optimized convolutional neural networks and explainable artificial intelligence-based analysis," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 27, no. 14, pp. 2041–2057, 2024.

[26] A. M. Salih, I. B. Galazzo, P. Gkontra, E. Rauseo, A. M. Lee, K. Lekadir, P. Radeva, S. E. Petersen, and G. Menegaz, "A review of evaluation approaches for explainable ai with applications in cardiology," *Artificial Intelligence Review*, vol. 57, no. 9, p. 240, 2024.

[27] R. K. Makumbura, L. Mampitiya, N. Rathnayake, D. Meddage, S. Henna, T. L. Dang, Y. Hoshino, and U. Rathnayake, "Advancing water quality assessment and prediction using machine learning models, coupled with explainable artificial intelligence (xai) techniques like shapley additive explanations (shap) for interpreting the black-box nature," *Results in Engineering*, vol. 23, p. 102831, 2024.

[28] J. Ryu, Y. M. Ryu, Y.-S. Jung, S.-j. Kim, Y. J. Lee, E.-K. Lee, S.-K. Kim, T.-S. Kim, T. H. Kim, C. Y. Lee *et al.*, "Extent of thyroidectomy affects vocal and throat functions: a prospective observational study of lobectomy versus total thyroidectomy," *Surgery*, vol. 154, no. 3, pp. 611–620, 2013.

[29] D. A. Vicente, N. P. Solomon, I. Avital, L. R. Henry, R. S. Howard, L. B. Helou, G. L. Coppit, C. D. Shriver, C. C. Buckenmaier, S. K. Libutti *et al.*, "Voice outcomes after total thyroidectomy, partial thyroidectomy, or non-neck surgery using a prospective multifactorial assessment," *Journal of the American College of Surgeons*, vol. 219, no. 1, pp. 152–163, 2014.

[30] A. Stojadinovic, A. R. Shaha, R. F. Orlikoff, A. Nissan, M.-F. Kornak, B. Singh, J. O. Boyle, J. P. Shah, M. F. Brennan, and D. H. Kraus, "Prospective functional voice assessment in patients undergoing thyroid surgery," *Annals of surgery*, vol. 236, no. 6, pp. 823–832, 2002.

[31] B. Yılmaz, S. Bakır, E. E. Yılmaz, E. Şengül, Ö. Uslukaya, A. Gül, F. E. Özkurt, and İ. Topçu, "An analysis on aerodynamic and acoustic changes after thyroidectomy," *International Surgery*, vol. 101, no. 5-6, pp. 233–240, 2016.

[32] P. Aluffi, M. Policarpo, C. Cherovac, M. Olina, R. Dosdegani, and F. Pia, "Post-thyroidectomy superior laryngeal nerve injury," *European Archives of Oto-Rhino-Laryngology*, vol. 258, no. 9, pp. 451–454, 2001.

[33] A. Palaparthi and I. R. Titze, "Analysis of glottal inverse filtering in the presence of source-filter interaction," *Speech communication*, vol. 123, pp. 98–108, 2020.

[34] K. R. Mittapalle, H. Pohjalainen, P. Helkkula, K. Kaitue, M. Minkkinen, H. Tolppanen, T. Nieminen, and P. Alku, "Glottal flow characteristics in vowels produced by speakers with heart failure," *Speech Communication*, vol. 137, pp. 35–43, 2022.

[35] P. Alku, M. Airas, E. Björkner, and J. Sundberg, "An amplitude quotient based method to analyze changes in the shape of the glottal pulse in the regulation of vocal intensity," *The Journal of the Acoustical Society of America*, vol. 120, no. 2, pp. 1052–1062, 2006.

[36] M. Novotnỳ, P. Dušek, I. Daly, E. Růžička, and J. Rusz, "Glottal source analysis of voice deficits in newly diagnosed drug-naïve patients with parkinson's disease: correlation between acoustic speech characteristics and non-speech motor performance," *Biomedical Signal Processing and Control*, vol. 57, p. 101818, 2020.

[37] Y. Xiao, Z. Wu, S. Ruan, Y. Xiong, and T. Huang, "Development and validation of the nomogram for predicting preoperative vocal cord palsy in thyroid cancer patients," *Gland Surgery*, vol. 10, no. 2, p. 541, 2021.

A Dual-Chain and Differential Privacy-Based Solution for Medical Data Privacy Protection and Access Control

Cen Gu[1], Luping Wang[2]*, Hongjie Wu[3]*
Suzhou University of Science and Technology, 99 Xuefu Road, SuZhou, 215004, China[1,2,3]
Jiangsu Province Key Laboratory of Intelligent Building, Energy Efficiency, 99 Xuefu Road, SuZhou, 215004, China[3]

***Abstract*—As living standards rise, people are paying increasing attention to health. Vast quantities of medical data are generated daily, yet each piece contains sensitive information such as patients' names, mobile numbers, email addresses, and places of employment. Should this information be compromised, the consequences would be irreversible, causing severe damage. Traditional solutions merely implement access control policies, permitting data access only to authorised personnel. While this approach offers some protection, even compliant users cannot be entirely trusted and may engage in malicious activities. Once data is accessed, patients' sensitive information becomes fully exposed to the user, posing a significant data security risk. Therefore, this study proposes a medical data sharing scheme based on Dual-Chain and differential privacy. It employs a hybrid approach combining private chains, consortium chains, and IPFS. Internal hospital personnel can access data after de-identification, while external parties can only access data that has been de-identified and subsequently augmented with noise. This significantly enhances security. The experimental section of this study also demonstrates that the proposed scheme effectively protects data, while the data shared with external users enables them to successfully complete downstream tasks.**

***Keywords*—Blockchain; differential privacy; IPFS; medical data**

I. Introduction

People's living standards are rising as a result of the economy's quick development. Health has become an increasingly prominent concern [1][2]. Consequently, in this era of information and data, vast amounts of medical data are generated daily [3]. Sharing and analyzing this data helps doctors make better decisions and provide effective patient care. However, this data often contains sensitive patient information, such as names, employers, phone numbers, and email addresses [4][5]. Moreover, during the sharing process, data users are not entirely trustworthy and may engage in malicious activities [6]. This significantly increases the risk of information leaks, resulting in extremely low security and reliability [7].

II. Related Works

Over time, with continuous technological advancement, significant achievements have been made in the privacy protection and access control of medical data. Akinyele et al. [8] introduced an attribute-based encryption access control scheme for personal health data on mobile devices, enabling data owners to exercise granular control over data access and sharing. Zhou et al. [9] designed a CP-ABE-based data exchange mechanism that allocates access permissions based on requester attributes and defines specific access policies, incorporating attribute timestamps to support attribute revocation. However, a common limitation of these ABE-based access control systems is their reliance on a single attribute authority to manage attribute permissions and key distribution. As a result, there is a major single point of breakdown: should the authority be compromised, it jeopardises system stability and overall security, undermining reliability.

To address potential single points of failure, Chase et al. [10] proposed a solution based on multi-attribute authorization authorities. However, it still relies on a fully trusted central authority, creating a significant vulnerability: should the central authority cease to be trustworthy, the entire system's security would be compromised, and the integrity of all encrypted data would be undermined, failing to achieve true decentralization. Liu et al. [11] subsequently suggested adopting a distributed multi-authority access control scheme, yet regulatory bodies continued to play a dominant role. The system remains only partially decentralized and remains susceptible to centralized power, which undermines its security and efficiency.

Subsequently, with the advent of blockchain technology [12][13], its inherent transparency and decentralization characteristics prompted integration efforts with existing systems. In 2016, Azaria et al. [14] proposed MedRec, an electronic health record management system. By employing distributed ledger technology to store patient records on a blockchain, it created a decentralized record management framework. This approach preserves data integrity and prevents tampering, while facilitating seamless sharing of medical records between distinct healthcare providers. Ownership, access rights, and data integrity of medical records are governed through smart contracts [15][16]. B. Chen et al. [17] and B.B. Gupta et al. [18] proposed integrating blockchain with CP-ABE to enhance security. In order to overcome centralized healthcare systems' shortcomings, Wang et al. [19] developed a trusted healthcare data management platform leveraging blockchain smart contracts and attribute-based encryption for flexible access control. Jin et al. [20] combined blockchain, smart contracts, and proxy re-encryption for secure medical information sharing. However, it lacks attribute integration, limiting fine-grained access control, and depends heavily on user activity, which may compromise security. Chen et al. [21] designed a blockchain-based access control scheme with hybrid on-chain/off-chain

*Corresponding authors.

storage to reduce expense of storage. However, the absence of hashing and signing the IPFS-generated content identifier (CID) on upload enables potential injection of malicious files, lowering system security.

However, the aforementioned approaches merely protect data at the access control layer by establishing an access policy, allowing only those who meet the policy to access the data. Yet in reality, even some visitors who comply with the access policy may not be entirely trustworthy. Over time, they could potentially engage in malicious activities, leading to data leaks and misuse. Therefore, Shao et al. [22] proposed a solution combining dynamic reputation with token-based allocation. Blockchain nodes distribute varying quantities of tokens to each visitor based on their reputation. After each access event, the reputation value is updated, and the token allocation is reassigned. While this approach offers some protection for data privacy, it relies on the assumption that visitors remain trustworthy throughout. If a visitor betrays trust during the period when they possess tokens for access, it could still result in data leakage. Cai et al. [23] advised a federated blockchain-based healthcare data sharing system with enhanced proxy re-encryption for improved data sharing security [24][25]. However, data privacy protection remains inadequate. Therefore, this study addresses the shortcomings of all previous approaches by proposing a novel solution with the following key contributions:

- This work introduces a role-oriented data classification framework that structurally separates internal raw-data access from external privacy-preserving data release. Instead of applying de-identification and differential privacy as isolated preprocessing steps, the proposed scheme embeds them into distinct operational domains governed by different access policies. Internal medical personnel access encrypted, but structurally intact datasets under strict attribute-based control, while external researchers interact only with differentially private representations. This separation reduces cross-domain inference risk and limits unnecessary exposure of raw medical records.
- A functionally decoupled dual-chain architecture is designed to enforce domain-specific governance. The private hospital chain manages identity authentication and fine-grained CP-ABE-based access control for raw data, whereas the consortium chain supervises differentially private data publication and external access requests. This architectural separation is not merely a deployment choice but a security-driven design that reduces trust concentration, isolates attack surfaces, and mitigates risks associated with single-chain or centralized systems.
- The scheme integrates blockchain integrity guarantees with IPFS-based encrypted storage in a policy-enforced workflow. By storing only metadata and cryptographic hashes on-chain, while maintaining encrypted bulk data off-chain, the framework ensures tamper-evident verification without compromising scalability. Unlike conventional blockchain-IPFS integrations that focus solely on storage optimization, this design tightly couples data integrity validation with access policy enforcement and privacy quantification, forming a unified secure data-sharing model.

III. Organization

This study's general organization is separated into four major sections: Section IV displays the preliminary knowledge used in the work. Section V analyzes the overall scheme, mainly from the overall model of the scheme, the description of the scheme and the specific flow of the scheme. Section VI analyzes the security of the entire solution. The performance analysis of the suggested plan is provided in Section VII. Lastly, we conclude the study in Section VIII.

IV. Background Knowledge

This section primarily introduces the technologies used in the study: blockchain technology, differential privacy technology, bilinear mapping, and linear secret sharing.

A. Blockchain

Blockchain is a fault-tolerant, append-only replicated state machine maintained by a distributed set of nodes over an unreliable network [26][27]. Transactions are batched into blocks that commit cryptographically to prior state, yielding tamper-evident immutability under standard hash-collision assumptions. A consensus layer establishes a globally consistent total order and finality; permissionless systems achieve Sybil resistance through resource-based mechanisms, whereas permissioned deployments typically rely on Byzantine fault-tolerant agreement. Public-key cryptography underpins identity, authentication, and authorization, and light-client protocols enable inclusion verification without full replication. Deterministic smart contracts specify application logic as state-transition functions executed by validators, enabling verifiable automation, while raising concerns about correctness, composability, and adversarial execution. Current research targets scalability, latency and throughput, privacy via advanced cryptography, and cryptoeconomic robustness. Despite open challenges, blockchains provide auditability and credible neutrality for applications in digital assets, decentralized finance, and provenance tracking [28][29].

B. Differential Privacy

Differential privacy [30][31], is a provable privacy framework for statistical release and machine learning that injects noise calibrated to the query's sensitivity, ensuring that the inclusion or exclusion of any single individual has only a controlled, quantifiable effect on the output distribution; formally, for neighboring datasets $D \sim D'$ that differ in one record and a randomized mechanism M, if the inequality below holds for every measurable event S, then M satisfies (ε, δ)-differential privacy [32][33], where ε bounds multiplicative leakage and δ allows a negligible failure probability, as shown in Eq. (1):

$$\forall\, D \sim D',\ \forall\, S:\ \Pr\big[M(D) \in S\big] \ \le\ e^{\varepsilon}\ \Pr\big[M(D') \in S\big] + \delta. \tag{1}$$

C. Bilinear Mapping

In elliptic curve cryptography, a bilinear mapping is a mapping e, in which elements of two elliptic curve groups are mapped to another group [34][35]. Suppose we have two groups G_1 and G_2, and a target group G_T [36]. We define a bilinear mapping as shown in Eq. (2):

$$e : G_1 \times G_2 \to G_T \tag{2}$$

This mapping has several important properties:

- Bilinear: For any $p \in G_1$, $q \in G_2$ and scalars a and b, the bilinear mapping is satisfied in Eq. (3):

$$e(p^a, q^b) = e(p, q)^{ab} \tag{3}$$

- Non-degradation: The bilinear mapping is not degenerate, there exists at least one $p \in G_1$ and $q \in G_2$, such that $e(p, q) \neq 1$ (unit element).
- Computability: For most implementations, computing bilinear mappings is efficient in practice.

D. Linear Secret Sharing Scheme

Linear Secret Sharing Scheme (LSSS) [37][38] first splits the secret into n distinct components and then sends these n parts to n distinct participants, and a set of authorized participants can recover the secret by merging their shares [39][40]. It is defined as follows:

The description of the access framework is (M, ρ), P stands for the group of characteristics, containing the attribute names and attribute values. M is a linear secret sharing matrix of size $l \times n$. ρ is a mapping function, and $\rho(i)$ refers to the mapping of the ith row of the matrix M to an attribute name. Map the secret s to a value on $\mathbb{Z}_p^*$ with p is a prime number. Construct a vector $\mathbf{v} = (s, r_2, r_3, \ldots, r_n)^T$ by randomly choosing $\{r_2, r_3, \ldots, r_n\} \in \mathbb{Z}_p^*$, and compute $\gamma_i = M_j \cdot \mathbf{v}$ where γ_i refers to the attribute named $\rho(i)$ — the secret share of all uses of the attribute [41][42].

Secret reconstruction: Let $B \in M$ be the collection of attributes that comply with the authorizations of the access structure, where $I = \{i \mid \rho(i) \in B\} \subseteq \{1, 2, \ldots, l\}$. Being able to find the coefficients $\{x_i \in \mathbb{Z}_p^*\}_{i \in I}$ in polynomial time such that $\sum_{i \in I} x_i M_i = (1, 0, \ldots, 0)$, and then compute $s = \sum_{i \in I} x_i \gamma_i$.

V. Scheme Structure

This proposal primarily consists of six entities: data owners, data users, attribute authorization agencies, authoritative certification agencies, IPFS, and Blockchain. Fig. 1 displays the complete model diagram.

1) Data Owner (DO): DO is primarily responsible for establishing access policies, authorizing visitors who meet these policies, and encrypting and uploading both original de-identified medical data and data with privacy-preserving noise added to IPFS. They then upload metadata to the blockchain, comprising the CID returned by IPFS, the CID's hash value, and the hash value of the encrypted ciphertext.

2) Data User (DU): DU here likely refers to internal medical personnel and external research scientists. Before accessing data, DU must submit an authorization request to DO. Only DU meeting the access policy are authorized to access the data.

3) Attribute Authorities (AA): Multiple Attribute Authorities work together to manage all attributes, authenticate the identity of the visitor and generate the visitor's attribute key.

4) Certification Authority (CA): All users must register with the CA, which issues certificates to users, generates their public and private keys, and assigns them a unique *id*.

5) Interplanetary File System (IPFS): IPFS primarily stores large-volume raw medical data to alleviate storage tension on the chain and returns the CID to the DO.

6) Blockchain (BC): This is mostly made up of a consortium blockchain created by outside research institutes and a private blockchain inside the hospital. The blockchain stores metadata and records all activities.

The following steps make up the overall scheme's basic process:

- Step 1: Each entity in the system first registers with CA and obtains its own public and private key, which it uploads to the BC.
- Step 2: DO encrypts both types of data using symmetric encryption algorithms and upload them to IPFS. Metadata composed of the CID of data that has undergone only de-identification processing is uploaded to the hospital's internal private chain. Metadata composed of the CID of data that has been added noise via DP is uploaded to the consortium chain. Separate access policies are established for internal and external personnel.
- Step 3: DU requests authorization from DO to access data. DO verifies compliance with access policies. If compliant, AA will assign DU an attribute private key. Blockchain nodes will send metadata to DU.
- Step 4: Upon receiving the metadata, DU first performs a hash operation on the CID within the metadata. If the hash matches, it uses the CID to request access to the original data from IPFS.
- Step 5: After obtaining the encrypted raw data from IPFS, DU first performs a hash operation on the ciphertext and verifies its consistency with the hash value stored in the metadata. If consistent, it uses its own attribute private key to decrypt the symmetric encryption key, thereby decrypting the raw data for access.

A. Scheme Definition

1) Initialization phase: During the initialization phase, all entities register with the CA and generate global parameters.

a) CA.setup: CA runs *CA.Setup* to initialize the system. Given the security parameter λ, it selects a prime p and instantiates multiplicative cyclic groups G and G_T of order p with generator g, together with the bilinear map $e : G \times G \to G_T$.

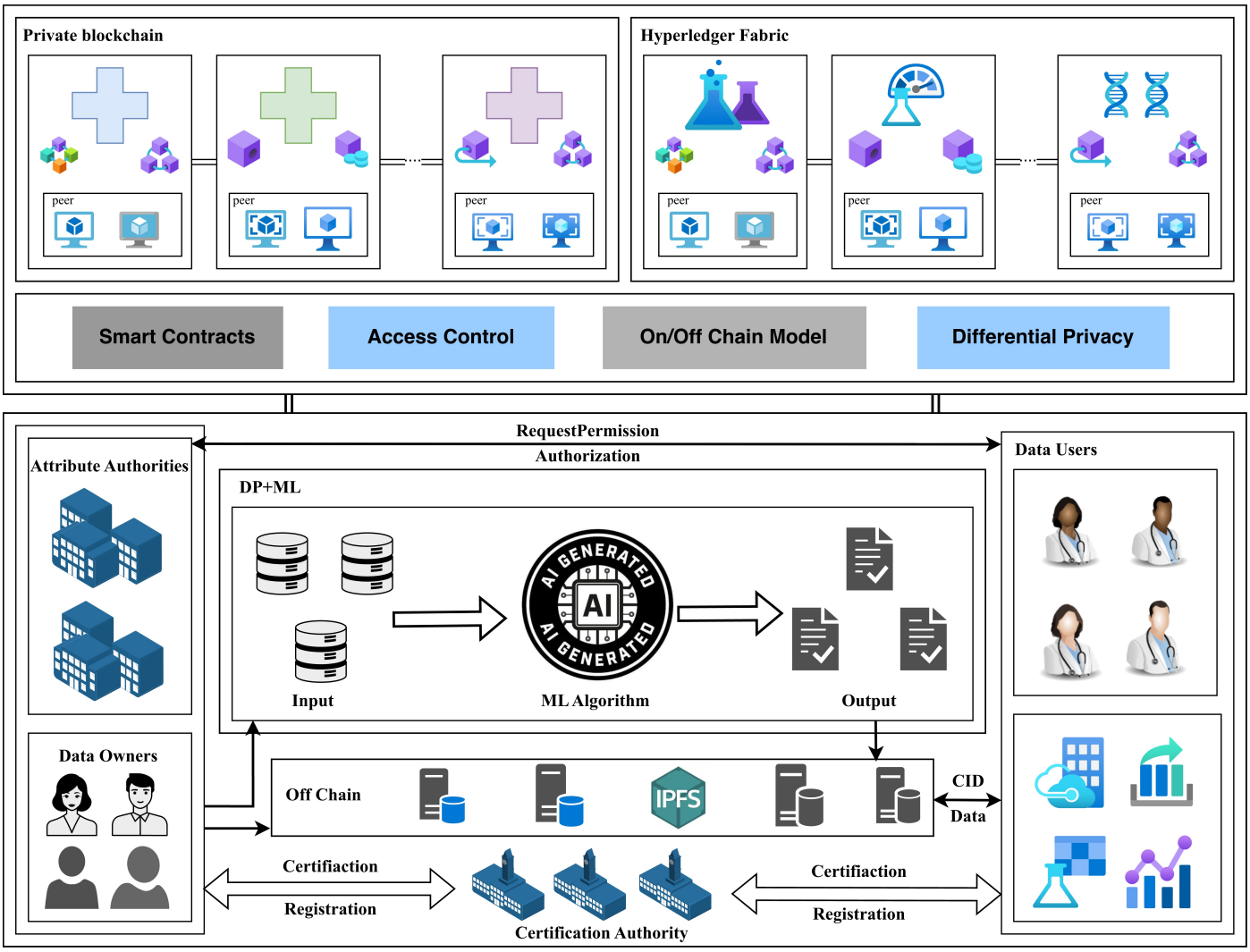


Fig. 1. System model.

It also chooses hash functions $H : \{0,1\}^* \to G$. The resulting global public parameters GPP are given in Eq. (4):

$$GPP = \{G, G_T, g, p, H, e\} \tag{4}$$

b) AA.register: Every AA submits a request together with its identifying information $info_a$, to register with the CA. The CA derives a unique identifier $aid \leftarrow H(info_a)$. The AA then samples $\alpha_{aid}, \beta_{aid} \in \mathbb{Z}_p$ to form its private key SK_{aid}, computes the public key $PK_{aid} = \left(g^{\beta_{aid}}, e(g,g)^{\alpha_{aid}}\right)$, and registers this public key with the CA.

c) DO.register: Each DO registers with the CA by submitting a request along with its identity information $info_o$. The CA computes a unique identifier $oid \leftarrow H(info_o)$. The DO then samples $\gamma \in \mathbb{Z}_p$ as its private key SK_{oid}, derives the corresponding public key $PK_{oid} = g^{\gamma}$, and registers this public key with the CA.

d) DU.register: By submitting a request and supplying its identifying information $info_u$, each DU registers with the CA. The CA derives a unique identifier $uid \leftarrow H(info_u)$. The DU samples $\mu \in \mathbb{Z}_p$ as its private key SK_{uid}, computes the corresponding public key $PK_{uid} = g^{\mu}$, and registers this public key with the CA.

2) Key generation phase:

a) KeyGen: The user requests a private key from each attribute authority AA_{aid} that governs their attributes. Let S be the user's attribute set, and assume $att \in S$ is administered by AA_{aid}. Before issuance, the authority verifies the user's identity uid and the validity of $Cert_{uid}$. If verification succeeds, AA_{aid} samples $r_{aid} \in \mathbb{Z}_p$ and constructs the user's attribute private-key component, as in Eq. (5):

$$SK_{att} = (g^{\alpha_{aid}} \cdot H(att)^{r_{aid}},\ g^{r_{aid}}) \tag{5}$$

Then the final private key of the user is the set of private keys produced by all relevant attribute authorizations, as shown in Eq. (6):

$$SK_{uid,att} = \bigcup_{aid} \{SK_{att} \mid att \in S \cap AA_{aid}\} \tag{6}$$

3) De-identification and DP phase: We utilized 500,000 outpatient medical records, first de-identifying sensitive fields such as patient names, workplaces, mobile numbers, and email addresses to protect privacy. This anonymized data was provided to internal personnel for viewing and use. Subsequently, we employed a rule-driven weakly supervised learning approach to match the de-identified data with a set of positive-class terms (infection-related) from the "initial diagnosis" field, generating binary labels as shown in Eq. (7). During feature construction, we exclusively utilized non-diagnostic text categories ("chief complaint + medical history"), while excluding diagnostic keywords used for annotation. This yields a bag-of-words count vector, as shown in Eq. (8). For each sample vector, we perform per-sample processing, as described in Eq. (9) and apply ℓ_1 regularization to limit its contribution. During the differential privacy phase, independent Laplace noise is added to each dimension of the pruned count vector, as shown in Eq. (10):

$$y = 1\{\, \exists\, k \in k_{\text{pos}} \mid k \subset \text{diagnosis} \,\} \tag{7}$$

$$x = \big[\, c(w_1), \ldots, c\big(w_{|v|}\big) \,\big] \tag{8}$$

$$x^{\text{clip}} = x \cdot \min\left(1, \frac{C}{\|x\|_1}\right) \tag{9}$$

$$\tilde{x} = x^{\text{clip}} + \eta,\ \eta_j \overset{\text{i.i.d.}}{\sim} \text{Laplace}(0, b),\ b = \frac{C}{\epsilon}. \tag{10}$$

4) Data encryption phase: DO chooses to employ symmetric encryption techniques to encrypt both the de-identified data and the data with additional noise via DP due to the effectiveness of encrypting massive amounts of raw data. DO selects different symmetric encryption key $k \in \mathbb{G}_T$ to encrypt each type of data separately to acquire the ciphertext C. Separate access policies (M, ρ) used to encrypt different symmetric key k. Here, M is an $l \times n$ matrix and ρ maps each row index i to an attribute. First sample a random vector $\mathbf{v} = (k, v_2, \ldots, v_n) \in \mathbb{Z}_p^n$. Compute the secret shares, as in Eq. (11); each x_i is associated with the attribute $\rho(i)$. For each row i, assume $\rho(i)$ is administered by authority AA_{aid}; choose $t_i \in \mathbb{Z}_p$ and derive the ciphertext component, as in Eq. (12). Next, compute the pairing value, as in Eq. (13). The final metadata is given in Eq. (14):

$$x_i = M_i \cdot \mathbf{v},\ \forall i \in \{1, \ldots, l\} \tag{11}$$

$$C_{i,1} = g^{\beta_{aid} x_i},\ C_{i,2} = H(\rho(i))^{x_i},\ C_{i,3} = g^{t_i} \tag{12}$$

$$D_i = e(g,g)^{\alpha_{aid} x_i} \cdot e(g,g)^{\beta_{aid} t_i} \tag{13}$$

$$metadata = \Big(C, hash(C),\ CID,\ hash(CID), \{C_{i,1},\, C_{i,2},\, C_{i,3},\, D_i\}_{i=1}^{\ell},\ k \cdot \prod_{aid} e(g,g)^{\alpha_{aid} k}\Big) \tag{14}$$

5) Data decryption phase: If DU meets the access policy, in addition to a reconstruction factor $\{w_i\}$ such that $\sum_{i \in I} w_i M_i = (1, 0, \ldots, 0)$. The pairwise value of each share is recovered through Eq. (15) and use the reconstruction coefficients w_i to decrypt the symmetric key k, according to Eq. (16) and Eq. (17). Eventually the DU decrypts the C via the obtained symmetric key k to get the data it wants to access.

$$\frac{e(C_{i,1}, g^{\alpha_{aid}} H(\rho(i))^{r_{aid}})}{e(C_{i,3}^{\beta_{aid}}, g^{r_{aid}})} = \frac{e(g^{x_i}, g^{\alpha_{aid}} H(\rho(i))^{r_{aid}})}{e(g^{\beta_{aid} t_i}, g^{r_{aid}})} \tag{15}$$

$$\prod_{i \in I} (e(g,g)^{\alpha_{aid} \lambda_i})^{w_i} = e(g,g)^{\sum_{i \in I} \alpha_{aid} \lambda_i w_i} = e(g,g)^{\sum_{aid} \alpha_{aid} k} \tag{16}$$

$$k = \frac{k \cdot \prod_{aid} e(g,g)^{\alpha_{aid} k}}{e(g,g)^{\sum_{aid} \alpha_{aid} k}} \tag{17}$$

VI. Security Analysis

This section primarily analyzes the proposed scheme from three perspectives: threat model, correctness, and resistance to attacks.

A. Threat Model

In this work, we consider a probabilistic polynomial-time adversary operating in a realistic medical data-sharing environment. The adversary is assumed to have full visibility of all public system parameters, blockchain transactions, on-chain metadata, content identifiers CID, and encrypted data stored on IPFS. In particular, the adversary may obtain complete read access to ciphertext corresponding to both internally shared de-identified datasets and externally shared differentially private datasets. The adversary may attempt active manipulation, including CID substitution, ciphertext tampering, replay of outdated metadata, or injection of malicious files into the IPFS storage layer. Multiple malicious data users may also collude by combining their attribute private keys in an attempt to satisfy an access policy not satisfied individually. Furthermore, the adversary may possess auxiliary background knowledge, such as demographic attributes or partial medical information, and attempt linkage attacks, membership inference, or statistical re-identification against externally released data. At the system level, metadata analysis and traffic pattern observation, including access frequency and timing information, are also considered possible attack vectors.

We assume that standard cryptographic hardness assumptions hold, including the discrete logarithm and bilinear Diffie–Hellman assumptions in pairing-based groups, as well as the semantic security of symmetric encryption. The Certification Authority is trusted to correctly initialize the system and protect master secrets, while Attribute Authorities follow the protocol for attribute key issuance and are not assumed to be fully colluding with the adversary. Blockchain nodes operate under Byzantine fault tolerance assumptions, and the adversary controls fewer nodes than the consensus threshold required to rewrite confirmed ledger history. IPFS is considered an untrusted storage layer and provides no

confidentiality guarantee. Under these assumptions, we define the security objectives of confidentiality, fine-grained access control, collusion resistance, data integrity, and differential privacy protection for external data release.

B. Proof of Correctness

First, consider the reconstruction properties of LSSS. Let B denote the set of attributes satisfying the access policy, and let the corresponding row index set be:

$$I = \{\, i \mid \rho(i) \in B \,\} \subseteq \{1, \ldots, l\}.$$

According to the LSSS definition, there exists a set of reconstruction coefficients $\{w_i\}_{i \in I} \subset \mathbb{Z}_p$, such that:

$$\textstyle\sum_{i \in I} w_i M_i = (1, 0, \ldots, 0).$$

From Eq. (11), it can be seen that the secret share for each row is:

$$x_i = M_i \cdot v \quad (v = (k, v_2, \ldots, v_n)^T).$$

By taking a linear combination of these shares, k is obtained by Eq. (18):

$$\begin{aligned} \sum_{i \in I} w_i x_i &= \sum_{i \in I} w_i (M_i \cdot v) \\ &= \left(\sum_{i \in I} w_i M_i \right) \cdot v \\ &= (1, 0, \ldots, 0) \cdot v \\ &= k \end{aligned} \tag{18}$$

In the decryption process description, the text uses λ_i to denote the secret share associated with the i-th row, equivalent to the aforementioned x_i. The above derivation demonstrates that provided the access structure is satisfied, the secret k can be linearly recovered using an appropriate reconstruction coefficient w_i. For each row $i \in I$, administered by AA_{aid}, the DU holds the attribute private-key component:

$$SK_{\rho(i)} = \left(g^{\alpha_{aid}} H(\rho(i))^{r_{aid}},\, g^{r_{aid}} \right).$$

Using the ciphertext components and the private key above, DU computes the pairing ratio in Eq. (19):

$$\begin{aligned} \frac{e(C_{i,1},\, g^{\alpha_{aid}} H(\rho(i))^{r_{aid}})}{e\left(C_{i,3}^{\beta_{aid}},\, g^{r_{aid}}\right)} &= \frac{e\left(g^{\beta_{aid} x_i},\, g^{\alpha_{aid}} H(\rho(i))^{r_{aid}}\right)}{e(g^{\beta_{aid} t_i},\, g^{r_{aid}})} \\ &= e(g, g)^{\alpha_{aid} \lambda_i} \end{aligned} \tag{19}$$

By bilinearity, the masking randomness t_i, r_{aid} cancels out, leaving a factor depending only on $\alpha_{aid}\lambda_i$. DU then uses the reconstruction coefficients w_i over all $i \in I$ to obtain $e(g,g)^{\sum_{aid} \alpha_{aid} k}$, where the last step uses $\sum_{i \in I} w_i \lambda_i = k$. This yields the masked exponent $\sum_{aid} \alpha_{aid} k$. Finally, combining this with the masked key term from the metadata, DU acquires k by Eq. (20):

$$k = \frac{k \cdot \prod_{aid} e(g,g)^{\alpha_{aid} k}}{e(g,g)^{\sum_{aid} \alpha_{aid} k}} \tag{20}$$

C. Resistance to Attacks

The proposed scheme leverages multi-authority CP-ABE, blockchain with IPFS, and differential privacy to jointly provide confidentiality, integrity, and privacy, thereby resisting several typical classes of attacks. For passive eavesdropping and key-guessing attacks, even a powerful adversary with full access to blockchain transactions and IPFS storage only observes the ciphertext C, the CID and their hashes, the ABE ciphertext components $\{c_{i,1}, c_{i,2}, c_{i,3}, D_i\}$, and the masked key term $k \cdot \prod_{aid} e(g,g)^{\alpha_{aid} k}$. Without any legitimate attribute private key SK_{att}, the adversary cannot form the pairing ratios, nor can they derive $e(g,g)^{\sum_{aid} \alpha_{aid} k}$. Attempting to guess k from $k \cdot \prod_{aid} e(g,g)^{\alpha_{aid} k}$ requires solving discrete logarithm or DBDH-type problems in bilinear groups, which is infeasible under appropriate parameters. The scheme is thus robust against eavesdropping and brute-force attacks.

For unauthorized access and collusion, access control is enforced by a multi-authority CP-ABE with an LSSS access structure. If a user's attribute set S does not satisfy the policy, there exist no coefficients w_i such that $\sum_{i \in I} w_i M_i = (1, 0, \ldots, 0)$, and hence $\sum_{i \in I} w_i \lambda_i = k$ cannot hold. Even when multiple unauthorized users collude and pool their attribute keys into a union set S_U, if this union still fails to satisfy the access matrix, LSSS theory guarantees that no valid reconstruction coefficients can be found, so the colluding users cannot elevate their collective access privileges or derive k beyond what their authorized attributes permit.

For data tampering and replay attacks, the scheme relies on blockchain-stored hashes and timestamps to guarantee integrity and auditability. Upon decryption, DU recomputes the hashes of the received CID and the ciphertext C retrieved from IPFS and compares them to on-chain values. Any modification of CID or ciphertext causes hash mismatches and will be detected and rejected. Replay attacks that attempt to resubmit outdated metadata or access requests are mitigated by the append-only, timestamped blockchain ledger, which records the latest state and access logs. The system can detect and prevent abnormal replays by checking against the most recent blockchain state, thus resisting tampering and replay attacks on IPFS and the blockchain.

Regarding inference attacks and privacy leakage, external researchers only see de-identified and differentially private feature vectors. De-identification removes explicit identifiers such as names and contact information from raw records, while the differential privacy mechanism bounds the impact of any single record on the output distribution within a factor of e^{ε}. This formally limits the advantage of any adversary, regardless of side information, in inferring sensitive attributes of any individual patient from $\tilde{x}$. Taken together, the cryptographic confidentiality of k, fine-grained access control, integrity protection, and rigorous differential privacy guarantees show that the proposed scheme can effectively withstand eavesdropping, tampering, collusion, replay, and inference attacks, while still supporting practical and privacy-preserving medical data sharing.

TABLE I. COMPARISON OF EXISTING SCHEMES

Aspect	Scheme					
	[17]	[18]	[19]	[20]	[23]	Ours
Patient-centered	✓	✓	✓	✗	✗	✓
Data Confidentiality	✓	✓	✓	✓	✓	✓
Identity Privacy Protection	✗	✗	✓	✗	✓	✓
Secure Distributed Storage	✗	✗	✓	✓	✓	✓
Fine-grained Access Control	✓	✗	✓	✗	✗	✓
Data Privacy Protection	✗	✗	✗	✗	✗	✓
Dual-Chain	✗	✗	✗	✗	✗	✓

TABLE II. GRID OF C/ϵ VALUES

	ϵ					
$C\backslash\epsilon$	6	8	12	20	30	40
3	0.500	0.375	0.250	0.150	0.100	0.075
4	0.667	0.500	0.333	0.200	0.133	0.100
5	0.833	0.625	0.417	0.250	0.167	0.125
6	1.000	0.750	0.500	0.300	0.200	0.150

VII. PERFORMANCE ANALYSIS

Every experiment in this work was carried out on a device equipped with an Apple M4 processor, 16GB RAM, and macOS Sequoia 15.5 operating system.

A. Functional Analysis

According to Table I, the proposed scheme in this study outperforms existing approaches in terms of patient-centricity, identity information protection, data confidentiality, secure distributed storage, granular access control, and data privacy protection. In [18] and [23], the authors neglect identity privacy protection in their approaches, while [17] relies on cloud storage for medical data, introducing potential data security risks. In [19], the authors employ trust-based organizational access control, limiting patients' ownership control over their medical data. In [20], the authors fail to achieve granular access control with its identity-based approach. All these solutions neglect privacy-preserving data processing. Consequently, the proposed solution in this study demonstrates significant advantages across these dimensions.

B. Performance Evaluation of DP

During the differential privacy phase, we add independent Laplace noise to each dimension of the truncated counts and employ a dimensionless noise intensity *scale*$= C/\varepsilon$ to uniformly compare the privacy–utility tradeoff under different truncation thresholds and privacy budgets. This study systematically scans combinations of $C \in \{3, 4, 5, 6\}$ and $\varepsilon \in \{6, 8, 12, 20, 30, 40\}$, as shown in Table II. We split the 500,000 data points into 95% for training, 5% for validation, and 5% for testing. During training, we feed the de-identified features—perturbed with differential privacy—into a continuous scoring function $s(x) = \sigma(w^\top x + b_0)$. On the validation set, we perform a grid search over the threshold $t \in [0, 1]$ and select the value that maximizes the F_1 score; this threshold is then fixed for the testing phase. Finally, we visualize the evaluation metrics of the experimental results, as shown in Fig. 2.

First, in Fig. 2(a), Fig. 2(b), and Fig. 2(d), the three metrics AUROC, AUPRC, and F_1 increase monotonically as ε grows, from a moderate regime at $\varepsilon = 6$ the performance is in a moderate regime with AUROC in [0.78,0.84], AUPRC in[0.45,0.62], and F_1 in [0.48,0.59]. At $\varepsilon = 12$ the metrics rise to AUROC in [0.88,0.91], AUPRC in [0.68,0.79], and F_1 in [0.64,0.71]. Near $\varepsilon = 20$, the method enters a high-performance regime with AUROC in [0.96,0.97], AUPRC in [0.91,0.94], and F_1 in [0.83,0.86]. Further increasing to $\varepsilon = 30$ and 40 approaches the no-DP ceiling both AUROC and AUPRC near 0.99, $F_1 > 0.95$. This pattern rapidly gains with larger ε followed by diminishing returns beyond $\varepsilon \approx 20$ indicates that our method preserves strong predictive signal under moderate privacy, while additional relaxation yields limited benefit.

Second, the effect of the clipping threshold C appears as a horizontal shift at fixed ε: smaller C (more aggressive clipping) lowers the curves, with larger gaps at small ε; when ε is sufficiently large, the curves converge, suggesting that the impact of clipping on utility is attenuated at high ε.

Third, and crucially, [see Fig. 2(c)] in the summary plot of *scale* $= C/\varepsilon$ versus AUPRC, the points from different C values almost lie on a single master curve, showing that the noise *scale* $= C/\varepsilon$ is the key dimensionless variable governing utility: as *scale* increases from 0.1 to 1.0, AUPRC decreases smoothly and monotonically, and results for different C align at the same *scale*. This "curve collapse" not only confirms the theoretical expectation of our clip-then-noise design, but also provides a practical tuning rule for deployment.

Overall, under the Laplace mechanism with rigorous differential privacy guarantees, high utility is achieved already at moderate ε. The metrics improve monotonically with ε and exhibit a clear "sweet spot" around $\varepsilon = 12 - 20$, and all three evaluation dimensions consistently support these conclusions, indicating that DP-processed shared data are useful and controllable for downstream binary classification.

C. Time Consumption

In this step, we performed encryption and decryption operations on 24 datasets of approximately 500MB each after adding noise via DP, recording the time consumed, as shown in Fig. 3. This figure illustrates the distribution of time overhead across 24 data files during the symmetric encryption

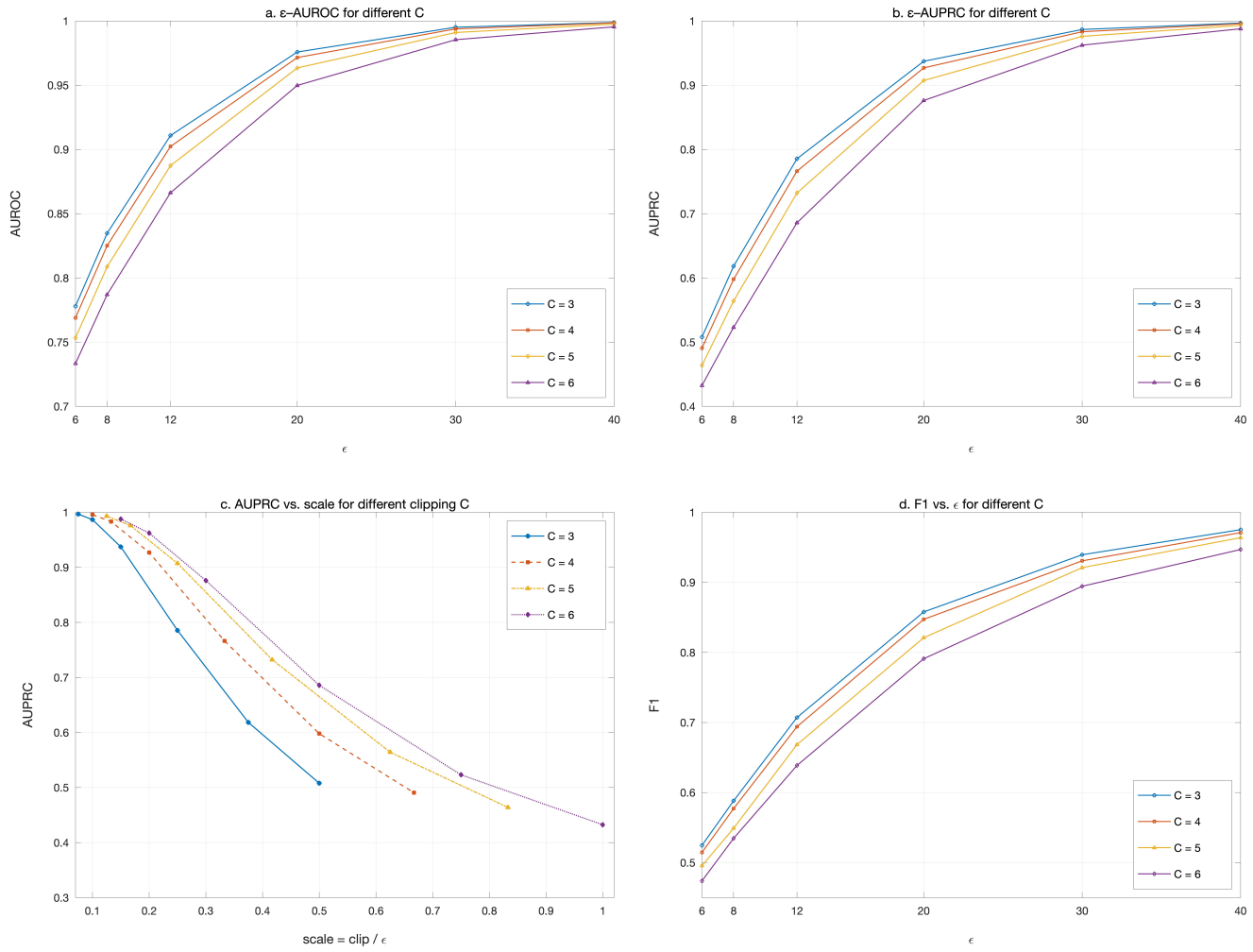


Fig. 2. Experimental results display.

TABLE III. HASH OPERATION TIME

Aspects	Time					
	dp_C3_e6	**dp_C3_e8**	**dp_C3_e12**	**dp_C3_e20**	**dp_C3_e30**	**dp_C3_e40**
enc	0.215 s	0.205 s	0.224 s	0.203 s	0.231 s	0.216 s
CID	0.008 ms	0.008 ms	0.005 ms	0.008 ms	0.007 ms	0.010 ms

and decryption phases. Overall, single-pass processing latency remains sub-second: the average encryption time is 0.294 s (range 0.255–0.372 s), while the average decryption time is 0.263 s (range 0.243–0.292 s). Compared with encryption, decryption achieves an average reduction of approximately 0.031 s, a 10.3% decrease in latency, reflecting the lighter key-recovery path. Both phases exhibit low variability: the standard deviation of encryption time is 0.029 s and that of decryption time is 0.014 s. This indicates stable and predictable operation across file/parameter combinations ($C = 3$–6, $\varepsilon = 6$–40). Notably, the most time-consuming samples occur under the C=3 configuration with smaller ε values, whereas other settings remain low and comparable. These results demonstrate the symmetric scheme's strong scalability and engineering viability—delivering consistent latency without sacrificing throughput, while further reducing processing overhead on the decryption side. After uploading the encrypted files to IPFS, the system obtain the CID and compile the metadata for upload to BC. We recorded the time consumed by the hash operations on the CID and ciphertext within the metadata, as shown in Table III. The hash operations for the ciphertext consistently took around 0.2s, while the CID hash times were in the millisecond range—demonstrating highly efficient processing. Finally, we recorded the time taken to upload each of these 24 files to IPFS and download them from IPFS. As shown in Fig. 4, both processes were completed within one second, demonstrating high efficiency.

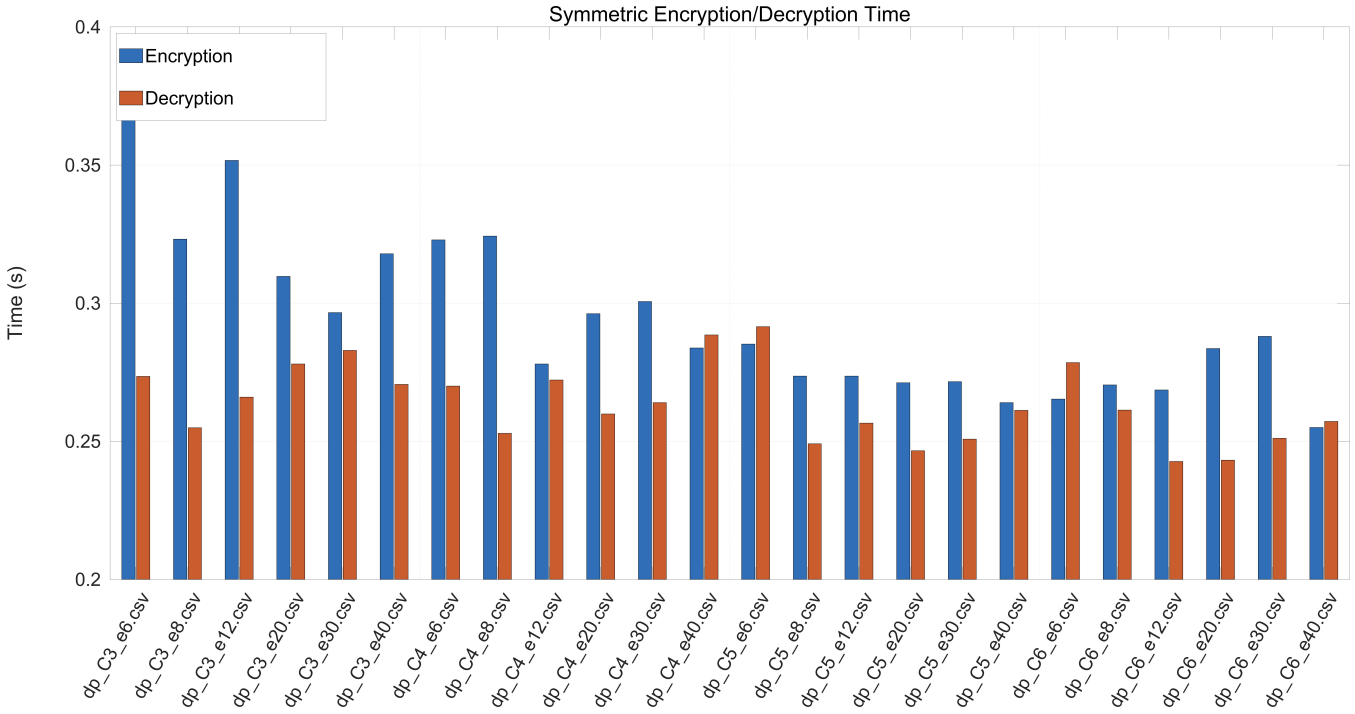


Fig. 3. Symmetric encryption and decryption time.

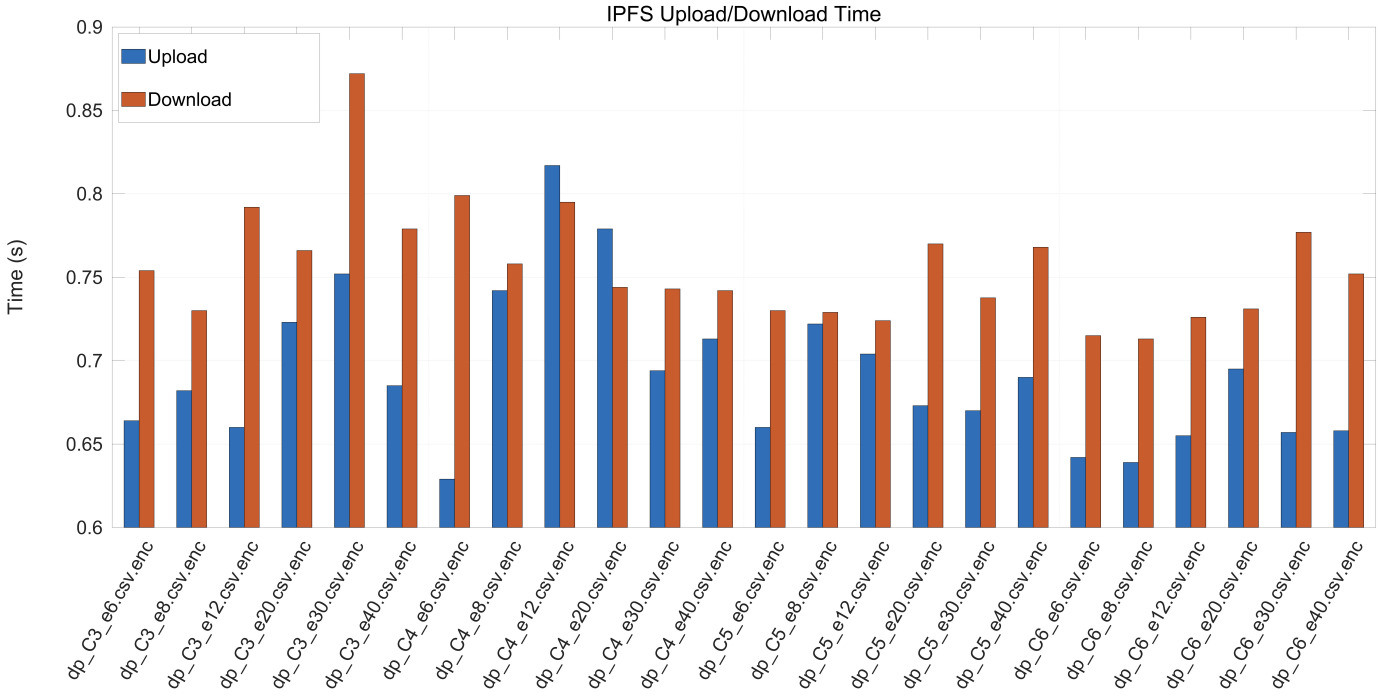


Fig. 4. IPFS upload and download time.

VIII. Conclusion

This study proposes a security-driven dual-chain framework for medical data sharing that structurally separates internal raw-data access from external privacy-preserving data release. By integrating multi-authority CP-ABE, blockchain-based governance, IPFS off-chain storage, and differential privacy under a unified security model, the proposed scheme achieves layered protection across confidentiality, integrity, access control, and statistical privacy dimensions. Unlike con-

ventional blockchain-based storage solutions that primarily focus on data immutability, the proposed architecture emphasizes role-oriented domain separation and trust decoupling. The private hospital chain enforces fine-grained access control over encrypted medical records, while the consortium chain supervises privacy-budgeted data publication for external research institutions. This structural separation reduces trust concentration, isolates attack surfaces, and mitigates cross-domain inference risks. Experimental results demonstrate that even after differential privacy perturbation, the released data maintain sufficient utility for downstream analytical tasks, thereby achieving a practical balance between privacy protection and data usability.

Overall, the proposed framework provides a unified, auditable, and quantifiably private data-sharing model suitable for heterogeneous medical environments. Future work will further explore adaptive privacy budget allocation, traffic-obfuscation mechanisms, and comprehensive evaluation of blockchain consensus latency and scalability under large-scale deployment.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China [62502334, 62506255, 62472301, 62372318], National Research Project [2020YFC2006602], Opening Topic Fund of Big Data Intelligent Engineering Laboratory of Jiangsu Province [SDGC2157], Suzhou Blockchain Data Privacy Protection Innovation Application Laboratory, Provincial Key Laboratory for Computer Information Processing Technology, Soochow University [KJS2166].

REFERENCES

[1] D. Kalita and K. B. Mirza, "Multivariate glucose forecasting using deep multihead attention layers inside neural basis expansion networks," *IEEE Journal of Biomedical and Health Informatics*, vol. 29, no. 5, pp. 3654–3663, 2025.

[2] L. Ge, A. N. McInnes, A. S. Widge, and K. K. Parhi, "Prediction of clinical response of transcranial magnetic stimulation treatment for major depressive disorder using hyperdimensional computing," *IEEE Journal of Biomedical and Health Informatics*, vol. 29, no. 5, pp. 3678–3686, 2025.

[3] Y. Shi, M. Wang, H. Liu, F. Zhao, A. Li, and X. Chen, "Mif: Multi-shot interactive fusion model for cancer survival prediction using pathological image and genomic data," *IEEE Journal of Biomedical and Health Informatics*, vol. 29, no. 5, pp. 3247–3258, 2025.

[4] H. Wu, J. Liu, T. Jiang, Q. Zou, S. Qi, Z. Cui, P. Tiwari, and Y. Ding, "Attentionmgt-dta: A multi-modal drug-target affinity prediction using graph transformer and attention mechanism," *Neural Netw.*, vol. 169, no. C, p. 623–636, Jan. 2024. [Online]. Available: https://doi.org/10.1016/j.neunet.2023.11.018

[5] J. Liu, S. Guan, Q. Zou, H. Wu, P. Tiwari, and Y. Ding, "Amdgt: Attention aware multi-modal fusion using a dual graph transformer for drug–disease associations prediction," *Know.-Based Syst.*, vol. 284, no. C, Jan. 2024. [Online]. Available: https://doi.org/10.1016/j.knosys.2023.111329

[6] J. Liu, F. Hu, Q. Zou, P. Tiwari, H. Wu, and Y. Ding, "Drug repositioning by multi-aspect heterogeneous graph contrastive learning and positive-fusion negative sampling strategy," *Inf. Fusion*, vol. 112, no. C, Dec. 2024. [Online]. Available: https://doi.org/10.1016/j.inffus.2024.102563

[7] H. Wu, J. Liu, R. Zhang, Y. Lu, G. Cui, Z. Cui, and Y. Ding, "A review of deep learning methods for ligand based drug virtual screening," *Fundamental Research*, vol. 4, no. 4, pp. 715–737, mar 2024.

[8] J. A. Akinyele *et al.*, "Securing electronic medical records using attribute-based encryption on mobile devices," in *Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices*. ACM, 2011.

[9] Y. Zhou *et al.*, "Sistema de telemedicina con protección de privacidad basada en cp-abe," *Investigación Clínica*, vol. 60, no. 6, pp. 1615–1626, 2019.

[10] M. Chase, "Multi-authority attribute based encryption," in *Theory of Cryptography: 4th Theory of Cryptography Conference (TCC 2007), Amsterdam, the Netherlands, February 21–24, 2007, Proceedings 4*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007.

[11] Z. Liu *et al.*, "Fully secure multi-authority ciphertext-policy attribute-based encryption without random oracles," in *Proceedings of the European Symposium on Research in Computer Security*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[12] Y. Jalghoum, A. Tahtamouni, and S. Mohammad, "An overview of blockchain technology in finance: A jordanian exploratory study," *Int. J. Crit. Infrastructures*, vol. 22, no. 6, 2026. [Online]. Available: https://doi.org/10.1504/ijcis.2026.10068995

[13] P. Jiang and L. Zhu, *Blockchain Technology - Cross-Chain Regulation and Privacy*. Springer, 2025. [Online]. Available: https://doi.org/10.1007/978-981-96-4395-0

[14] A. Ekblaw *et al.*, "A case study for blockchain in healthcare: "medrec" prototype for electronic health records and medical research data," in *Proceedings of the IEEE Open & Big Data Conference*, vol. 13, 2016.

[15] Z. Zheng *et al.*, "An overview on smart contracts: Challenges, advances and platforms," *Future Generation Computer Systems*, vol. 105, pp. 475–491, 2020.

[16] M. Bartoletti, R. Marchesin, and R. Zunino, "Scalable UTXO smart contracts via fine-grained distributed state," *Future Gener. Comput. Syst.*, vol. 175, p. 108023, 2026. [Online]. Available: https://doi.org/10.1016/j.future.2025.108023

[17] B. Chen *et al.*, "Bpvse: Publicly verifiable searchable encryption for cloud-assisted electronic health records," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 3171–3184, 2023.

[18] B. B. Gupta *et al.*, "Blockchain-assisted secure fine-grained searchable encryption for a cloud-based healthcare cyber-physical system," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 12, pp. 1877–1890, 2021.

[19] F. Wang *et al.*, "Lightweight and secure data sharing based on proxy re-encryption for blockchain-enabled industrial internet of things," *IEEE Internet of Things Journal*, vol. 11, no. 8, pp. 14 115–14 126, 2023.

[20] B. Jin *et al.*, "Bcas: Blockchain-based secure access and sharing scheme for ehr data," *Digital Communications and Networks*, 2024.

[21] Y. Chen *et al.*, "A blockchain-based medical data sharing mechanism with attribute-based access control and privacy protection," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 6685762, 2021.

[22] I. Sukhodolsky and S. Zapechnikov, "A blockchain-based access control system for cloud storage," in *Proceedings of the 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. IEEE, 2018.

[23] C. Cai and Y. Liu, "Blockchain-based solution for secure sharing of medical data," *Computer Applications and Software*, vol. 41, no. 12, pp. 360–366, 2024.

[24] D. Zhu, Z. Zhou, Y. Li, H. Zhang, Y. Chen, Z. Zhao, and J. Zheng, "A survey of data security sharing," *Symmetry*, vol. 17, no. 8, p. 1259, 2025. [Online]. Available: https://doi.org/10.3390/sym17081259

[25] R. Rao, "Financial data sharing based on cloud computing security: the application and effectiveness of homomorphic encryption technology," in *11th IEEE Conference on Big Data Security on Cloud, BigDataSecurity 2025, New York City, NY, USA, May 9-11, 2025*. IEEE, 2025, pp. 161–166. [Online]. Available: https://doi.org/10.1109/BigDataSecurity66063.2025.00029

[26] D. B. Gayathri and D. Sangeetha, "An efficient blockchain-based incremental provable data possession for a secure electronic healthcare system," *Biomed. Signal Process. Control.*, vol. 112, p. 108919, 2026. [Online]. Available: https://doi.org/10.1016/j.bspc.2025.108919

[27] L. Liao, J. Zhao, Q. Zhang, and H. Fang, "A blockchain-enhanced trust-driven batch authentication scheme for secure vanets," *Ad Hoc Networks*, vol. 180, p. 104045, 2026. [Online]. Available: https://doi.org/10.1016/j.adhoc.2025.104045

[28] Q. Chen, C. Peng, and D. Xu, "Fuzzy password authentication key exchange protocol in universal composable framework for blockchain privacy protection," *Comput. Stand. Interfaces*, vol. 95, p. 104032, 2026. [Online]. Available: https://doi.org/10.1016/j.csi.2025.104032

[29] M. Xie, Y. Yu, R. Chen, Y. Zhao, J. Ning, X. Yang, and Z. L. Jiang, "Multiuser data integrity auditing atop blockchain with secure user revocation for cognitive iot networks," *Comput. Stand. Interfaces*, vol. 95, p. 104042, 2026. [Online]. Available: https://doi.org/10.1016/j.csi.2025.104042

[30] Y. Zhou, Y. Li, A. Sheng, and G. Qi, "Multi-pursuer single-evader privacy-preserving differential games," *Appl. Math. Comput.*, vol. 508, p. 129612, 2026. [Online]. Available: https://doi.org/10.1016/j.amc.2025.129612

[31] T. Wang, T. Lin, Z. Liu, X. Xie, and S. Yao, "ADPGAN: adaptive differential privacy-preserving GAN for image privacy," *Inf. Fusion*, vol. 125, p. 103515, 2026. [Online]. Available: https://doi.org/10.1016/j.inffus.2025.103515

[32] A. Zarei, "Differential privacy in secure multiparty computation and deep neural networks," Ph.D. dissertation, Norwegian University of Science and Technology, Trondheim, Norway, 2025. [Online]. Available: https://hdl.handle.net/11250/3172984

[33] M. Adnan, M. H. Syed, A. Anjum, and S. Rehman, "A framework for privacy-preserving in iov using federated learning with differential privacy," *IEEE Access*, vol. 13, pp. 13 507–13 521, 2025. [Online]. Available: https://doi.org/10.1109/ACCESS.2025.3526934

[34] I. Sahu, "Bilinear-inverse-mapper: Analytical solution and algorithm for inverse mapping of bilinear interpolation of quadrilaterals," *Adv. Eng. Softw.*, vol. 208, p. 103975, 2025. [Online]. Available: https://doi.org/10.1016/j.advengsoft.2025.103975

[35] B. Huang, P. Huang, H. Yuan, and S. Liang, "A verifiable ranked ciphertext retrieval scheme based on bilinear mapping," *Concurr. Comput. Pract. Exp.*, vol. 34, no. 12, 2022. [Online]. Available: https://doi.org/10.1002/cpe.5829

[36] Y. Chen *et al.*, "A bilinear map pairing based authentication scheme for smart grid communications: Pauth," *IEEE Access*, vol. 7, pp. 22 633–22 643, 2019.

[37] A. Beimel, "Secret-sharing schemes: A survey," in *Coding and Cryptology, IWCC 2011*, ser. Lecture Notes in Computer Science, Y. M. Chee, Z. Guo, S. Ling, F. Shao, Y. Tang, and C. Xing, Eds., vol. 6639. Berlin, Heidelberg: Springer, 2011, pp. 11–46.

[38] U. Gupta and H. Mahdavifar, "Security of linear secret sharing schemes with noisy side-channel leakage," *IACR Cryptol. ePrint Arch.*, p. 987, 2025. [Online]. Available: https://eprint.iacr.org/2025/987

[39] A. Beimel, O. Farràs, Y. Mintz, and N. Peter, "Linear secret-sharing schemes for forbidden graph access structures," *IEEE Trans. Inf. Theory*, vol. 68, no. 3, pp. 2083–2100, 2022. [Online]. Available: https://doi.org/10.1109/TIT.2021.3132917

[40] K. Blackwell and M. Wootters, "A characterization of optimal-rate linear homomorphic secret sharing schemes, and applications," in *15th Innovations in Theoretical Computer Science Conference, ITCS 2024, January 30 to February 2, 2024, Berkeley, CA, USA*, ser. LIPIcs, V. Guruswami, Ed., vol. 287. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, pp. 16:1–16:20. [Online]. Available: https://doi.org/10.4230/LIPIcs.ITCS.2024.16

[41] A. Beimel, A. Ben-Efraim, C. Padró, and I. Tyomkin, "Multi-linear secret-sharing schemes," in *Theory of Cryptography*, ser. Lecture Notes in Computer Science, Y. Lindell, Ed., vol. 8349. Berlin, Heidelberg: Springer, 2014, pp. 394–418.

[42] T. Feneuil and M. Rivain, "Threshold linear secret sharing to the rescue of MPC-in-the-head," Cryptology ePrint Archive, Paper 2022/1407, 2022. [Online]. Available: https://eprint.iacr.org/2022/1407

Leveraging Statistical Invariants to Fortify CNN-Based Medical Diagnostics Against Adversarial Perturbations

Yassine Chahid[1]*, Anas Chahid[2], Ismail Chahid[3], Aissa Kerkour Elmiad[4]
ACSA, Mohammed First University, Oujda, 60000, Morocco[1]
SmartICT, Mohammed First University, Oujda, 60000, Morocco[2]
LARI, Mohammed First University, Oujda, 60000, Morocco[3,4]

***Abstract*—The integration of artificial intelligence (AI) in medical diagnostics is increasingly jeopardized by adversarial attacks—imperceptible perturbations designed to induce misclassification in Deep Learning models. While Convolutional Neural Networks (CNNs) have achieved state-of-the-art performance in medical image analysis, their susceptibility to gradient-based attacks poses a severe risk to patient safety and diagnostic integrity. This study addresses the critical need for robust defense mechanisms in X-ray diagnostics by proposing a Hybrid Ensemble model based on Stacked Generalization. Unlike single-paradigm approaches, our method fuses the spatial feature extraction capabilities of a CNN with the statistical anomaly detection power of a Random Forest (RF). We evaluated this architecture on a curated dataset of X-ray images subjected to Projected Gradient Descent (PGD) attacks with varying perturbation magnitudes (ϵ). The results demonstrate that the Hybrid Ensemble consistently outperforms individual models and standard adversarial training baselines. Under strong attack conditions ($\epsilon = 0.006$), the proposed model achieved an Area Under the Curve (AUC) of 0.919, significantly surpassing the adversarial training baseline (AUC 0.700). Furthermore, the ensemble reduced false positives to 108 compared to 138 for the CNN alone, enhancing clinical reliability. Theoretical motivation for the feature extraction process and extensive experimental validation suggest that leveraging statistical irregularities offers a computationally efficient and robust defense strategy suitable for real-time clinical deployment.**

***Keywords*—Adversarial defense; medical X-ray; synergistic ensemble; deep learning security; diagnostic robustness; PGD attack**

I. Introduction

A. The Role of AI in Medical Diagnostics

Medical diagnostics are evolving rapidly through the integration of artificial intelligence (AI), particularly deep learning. Convolutional Neural Networks (CNNs) have demonstrated remarkable proficiency in interpreting medical images, identifying subtle abnormalities in modalities ranging from ophthalmology to radiology [1], [3]. Recent studies published in *IJACSA* highlight the continued dominance of CNNs in detecting complex pathologies such as brain tumors [2] and skin lesions [4] [22]. These systems enhance diagnostic precision and streamline workflows, offering the potential for personalized medicine [5], [8]. From detecting pneumonia in chest X-rays [6] to segmenting tumors in MRI scans [7], deep learning models are becoming indispensable tools. However, as AI systems are increasingly deployed in critical clinical decision support roles, their reliability, interpretability, and security become paramount [9], [10].

B. The Threat of Adversarial Attacks

Despite their performance, deep learning models are vulnerable to adversarial attacks, which are subtle, purposeful alterations to input data that deceive the model [11] [56]. In medical imaging, these perturbations are often imperceptible to the human eye, but can cause a model to misclassify a pathological scan as healthy, or vice versa [12], [15]. Adversarial examples exploit the decision boundaries of neural networks. Mathematically, an adversarial example x_{adv} is derived from a clean input x by adding a perturbation δ such that $\|\delta\|_p \leq \epsilon$, where ϵ is small enough to maintain visual fidelity [14], [17].

The consequences of such attacks in a clinical setting are severe. A false negative could lead to a missed diagnosis for a critical condition, while a false positive could result in unnecessary invasive procedures and patient stress [16], [23]. Furthermore, as noted in recent security surveys [13], the possibility of malicious actors manipulating medical data raises significant concerns regarding insurance fraud and hospital cybersecurity [19], [20].

C. Problem Statement and Motivation

Existing defense mechanisms, such as adversarial training [17] or defensive distillation [57], often struggle to generalize across different attack types or incur high computational costs that hinder real-time deployment [48]. Moreover, defenses designed for natural images may not capture the specific structural and statistical regularities of medical X-rays [18], [24]. Medical images possess unique noise distributions (Poisson noise) and texture patterns that standard CNNs may overlook but which are critical for distinguishing clean from perturbed data [29], [44].

The motivation of this research stems from the observation that adversarial attacks, while spatially deceptive to convolution filters, often introduce high-frequency artifacts that disrupt the natural statistical distribution of the image [21]. Therefore, relying solely on deep learning for defense is insufficient. There is a clear research gap for defenses that leverage the unique statistical properties of medical images (orthogonality

*Corresponding author.

of features) to detect anomalies that deep learning models might miss.

D. Contribution and Novelty

This work proposes a novel Hybrid Ensemble defense that combines deep spatial features (CNN) with handcrafted statistical features (Random Forest). The core hypothesis is that while adversarial noise misleads CNNs by distorting spatial correlations, it inevitably disturbs low-level image statistics (e.g., spectral energy, histogram moments) that are easily detected by statistical classifiers (see Fig. 1). Our contributions are:

1) Ensemble architecture: A framework inspired by Stacked Generalization that fuses soft-probabilities from a CNN and a Random Forest, achieving complementary feature robustness.

2) Theoretical motivation: We provide a theoretical basis for the statistical features (Skewness, Kurtosis, Spectral Entropy) that make the Random Forest robust against PGD attacks.

3) Comprehensive evaluation: A rigorous evaluation using PGD attacks with varying ϵ $(0.003, 0.006)$ and a suite of noise types, demonstrating superior AUC (0.919) compared to standard adversarial training (0.700).

4) Clinical and computational analysis: A detailed analysis of computational cost (inference time) and clinical relevance (False Negative reduction), proving the solution's viability for real-time deployment.

II. Related Work

The field of adversarial defense in medical imaging has grown significantly. A comprehensive review by Olatunji et al. [25] and Taghanaki et al. [47] emphasizes the vulnerability of deep networks in high-stakes environments. In this section, we review the state-of-the-art methods and position our contribution relative to them. Table I presents a detailed comparison of proposed method with state-of-the-art defense mechanisms in medical imaging.

A. Adversarial Training

Proposed by Madry et al. [17], adversarial training remains the most popular defense [54]. It involves augmenting the training set with adversarial examples computed on the fly. While effective against the specific attacks seen during training, it suffers from the "robustness-accuracy trade-off" and often fails to generalize to unseen attacks or different perturbation magnitudes (ϵ). In medical imaging, this is particularly problematic as the noise patterns can vary widely between X-ray machines, as discussed in the context of COVID-19 detection by Al-Waisy et al. [26].

B. Defensive Distillation

Papernot et al. [57] proposed using a teacher-student network to smooth the decision gradients, making it harder for attackers to compute gradients. However, Carlini and Wagner [53] demonstrated that this method is vulnerable to stronger attacks that bypass gradient masking. Furthermore, distillation requires training multiple networks, increasing the computational burden.

C. Denoising and Pre-processing

Other approaches attempt to remove the adversarial noise before feeding the image to the classifier. Methods like Total Variance Minimization or Autoencoder-based denoising [58] fall into this category. While intuitive, they risk removing critical high-frequency clinical details (e.g., micro-calcifications in mammography or small fractures in X-rays), leading to a degradation of clean accuracy. This delicate balance between noise reduction and feature preservation is critical, as shown in the breast cancer detection study by Naseer et al. [45].

D. Statistical Anomaly Detection

Recent works have explored statistical testing (e.g., Maximum Mean Discrepancy) to detect adversarial samples [15]. While computationally cheap, these methods often suffer from high False Positive Rates (FPR) when applied to noisy medical data. Our approach improves upon this by integrating statistical detection within an ensemble voting mechanism, stabilizing the predictions.

E. Stacked Generalization

Our approach adapts the concept of *Stacked Generalization* (Stacking) [49] [38], which involves training a meta-classifier to combine the predictions of several base learners. This synergy relies on the diversity of the base models [46]. While Stacking is traditionally used to improve accuracy by reducing bias and variance, we repurpose it here for adversarial robustness. By combining a high-variance deep learner (CNN) with a lower-variance statistical learner (RF), we aim to mitigate the specific fragility of gradient-based models against L_∞ perturbations.

III. Materials and Methods

A. Dataset Preparation

The study utilizes the IRMA/ImageCLEFmed 2009 radiograph collection [31], comprising 14,410 anonymized X-ray images collected from routine clinical practice at RWTH Aachen. This dataset is widely recognized for benchmarking medical image analysis algorithms. To ensure a balanced evaluation, we curated a dataset containing equal distributions of clean and adversarial samples. Images were resized to 224×224 pixels and normalized to the $[0, 1]$ range to match the input requirements of standard deep learning architectures [32].

B. Attack Generation and Configuration

We evaluate the model primarily against non-adaptive PGD attacks, assuming a threat model where the adversary has access to the CNN gradients but not the statistical ensemble logic. We acknowledge that adaptive white-box attacks targeting the specific ensemble weights are possible but computationally more expensive for the attacker.

1) Fast Gradient Sign Method (FGSM): FGSM is a single-step attack that calculates the gradient of the loss function and updates the image once:

$$x_{adv} = x + \epsilon \cdot \text{sign}(\nabla_x L(\theta, x, y)) \tag{1}$$

While fast, FGSM is often too weak to fool robust models.

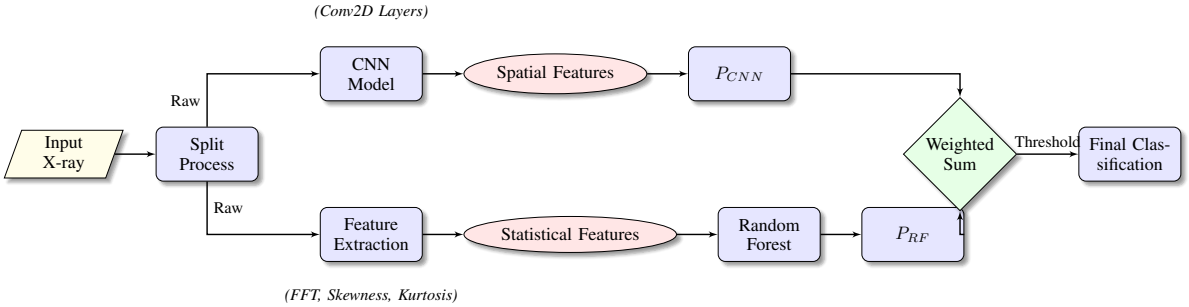


Fig. 1. Architectural overview of the Hybrid Ensemble Defense system. The input image is processed in parallel: spatially by the CNN and statistically by the Random Forest. The predictions are aggregated using optimized weights ($W_{CNN} = 0.4, W_{RF} = 0.6$) to produce the final robustness score.

TABLE I. DETAILED COMPARISON OF THE PROPOSED METHOD WITH STATE-OF-THE-ART DEFENSE MECHANISMS IN MEDICAL IMAGING

Defense Strategy	Core Mechanism	Key Strength	Primary Weakness	Inference Cost	Generalization	Ref.
Adversarial Training	Data Augmentation	High robustness to known attacks	Overfitting to specific ϵ	High (Training)	Low	[17]
Defensive Distillation	Gradient Masking	Hides gradient info	Vulnerable to CW attacks	Medium	Low	[57]
Image Denoising	Filtering/Autoencoders	Removes perturbation noise	Loss of fine clinical details	Medium	Medium	[58]
Adversarial Detection	Statistical Testing	Flags attacks before prediction	High False Positive Rate	Low	Medium	[15]
Hybrid Ensemble (Ours)	**CNN + Statistical RF**	**Orthogonal feature robustness**	**Slightly increased model size**	**Low (2.74ms)**	**High**	**–**

2) Projected Gradient Descent (PGD): PGD is an iterative version of FGSM, considered the strongest first-order attack. The update rule is:

$$x^{t+1} = \Pi_{x+\mathcal{S}} \left(x^t + \alpha \cdot \text{sign}(\nabla_x L(\theta, x^t, y)) \right) \tag{2}$$

Here, $\Pi_{x+\mathcal{S}}$ denotes the projection operator onto the L_∞ ball of radius ϵ, and α is the step size. We used $\alpha = \epsilon/10$ and 40 iterations.

To guarantee that the adversarial perturbations remained clinically imperceptible while being effective against the models, we meticulously calibrated the attack parameters. The specific configuration used in our experiments is detailed in Table II.

TABLE II. CONFIGURATION OF ADVERSARIAL ATTACK PARAMETERS OPTIMIZED FOR IMPERCEPTIBILITY

Parameter	Value
Attack Methods	{PGD, FGSM, Gaussian, S&P, Contrast}
PGD Steps	40
Gaussian Noise (σ)	0.008
Salt & Pepper Amount	0.008
Contrast Factor	1.1
Perturbation Constraint (ϵ)	0.003 (Subtle) – 0.006 (Strong)

C. Model Architectures

1) Convolutional Neural Network (CNN): A custom CNN was implemented using `Keras` [39] to capture spatial features [27]. The model was optimized using the Adam algorithm [36]. The mathematical operation of a convolutional layer l with kernel K on input I is given by:

$$(I * K)_{ij} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} I(i-m, j-n) K(m,n) + b \tag{3}$$

This operation captures local spatial dependencies (edges, textures). Our architecture consists of three convolutional blocks ($32 \rightarrow 64 \rightarrow 128$ filters), each followed by Batch Normalization and Dropout (0.25) [35]. The activation function used is ReLU: $f(x) = \max(0, x)$ [33]. Similar architectures have proven effective in recent pandemic-related imaging studies [34].

2) Random Forest (RF) and statistical feature engineering: The Random Forest classifier [28] (200 trees) was employed to detect statistical anomalies. Unlike CNNs, RFs are non-differentiable and base their decisions on thresholding feature values, making them inherently resistant to gradient-based attacks [40].

We extracted a vector V_f of handcrafted features. The rationale and mathematical definitions are as follows:

a) Statistical moments: Adversarial perturbations, even if subtle, alter the pixel intensity distribution. We calculate the Skewness (S) and Kurtosis (K) for an image I with mean μ and standard deviation σ:

$$S = \mathbb{E}\left[\left(\frac{I-\mu}{\sigma}\right)^3\right], \quad K = \mathbb{E}\left[\left(\frac{I-\mu}{\sigma}\right)^4\right] \tag{4}$$

Clean medical images typically follow specific statistical distributions (e.g., Rayleigh or Gaussian depending on tissue). PGD attacks tend to flatten the Kurtosis or introduce asymmetry (Skewness).

b) Spectral features (Frequency domain): PGD attacks often inject high-frequency noise that is imperceptible in the spatial domain but evident in the frequency domain. We use the Discrete Fourier Transform (DFT):

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) e^{-j2\pi(\frac{ux}{M} + \frac{vy}{N})} \tag{5}$$

We extract the Spectral Entropy (H) to quantify the complexity of the frequency distribution:

$$H = -\sum_{k} p_k \log_2(p_k) \tag{6}$$

where, p_k is the normalized power spectrum density. An adversarial image typically exhibits higher entropy due to the randomized high-frequency noise injection.

c) Texture analysis (Laplacian): To detect disruptions in local smoothness, we compute the Laplacian variance. The Laplacian operator $\nabla^2 I$ highlights regions of rapid intensity change:

$$\nabla^2 I = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{7}$$

The variance of $\nabla^2 I$ serves as a measure of image "roughness", which typically increases under adversarial attack.

d) Statistical validation: To ensure the reliability of our metrics and compute the 95% Confidence Intervals (CI), we employed non-parametric bootstrapping with 1,000 resamples on the test set.

e) Feature selection: The features (moments, spectral entropy, Laplacian) were selected based on their theoretical sensitivity to high-frequency noise. While a full ablation study is outside the scope of this study, preliminary analysis indicated that Spectral Entropy was the most discriminative feature against PGD noise.

D. Hybrid Ensemble and Weight Optimization

The Hybrid Ensemble aggregates the probability outputs of the CNN (P_{CNN}) and RF (P_{RF}) [37]. The final score S_E is a weighted sum:

$$S_E(x) = w_{\text{cnn}} \cdot P_{\text{CNN}}(x) + w_{\text{rf}} \cdot P_{\text{RF}}(x) \tag{8}$$

To determine optimal weights, we performed a grid search on a validation set. The decision process is outlined in Algorithm 1.

1) Complexity analysis: The addition of the ensemble method introduces a computational overhead. However, the complexity of the feature extraction is $O(N)$ (where N is the number of pixels) for moments and gradients, and $O(N \log N)$ for the FFT. The RF inference is $O(T \cdot D)$, where T is the number of trees and D is the depth. This is negligible compared to the $O(N \cdot K^2 \cdot L)$ complexity of the CNN (where L is layers and K kernel size). Thus, the ensemble remains efficient.

IV. Results and Discussion

A. Optimization of Ensemble Weights

The contribution of each model to the ensemble is critical for maximizing performance [30]. We conducted an extensive grid search to determine the optimal balance between the deep learning component (CNN) and the statistical component (RF). Fig. 2 illustrates the impact of varying the CNN weight (w_{cnn}) on five key metrics.

Algorithm 1 Hybrid Ensemble Decision Process

Require: Input image X, CNN Model M_{cnn}, RF Model M_{rf}
Require: Optimal Weights $w_{cnn} = 0.4$, $w_{rf} = 0.6$
Ensure: Class Prediction $\hat{y} \in \{0, 1\}$ (0: Clean, 1: Adversarial)
$P_{cnn} \leftarrow M_{cnn}$.predict($X$)
$V_{feat} \leftarrow$ ExtractFeatures(X) {Eq. 4, 5, 6}
$P_{rf} \leftarrow M_{rf}$.predict(V_{feat})
$S_{ensemble} \leftarrow (w_{cnn} \times P_{cnn}) + (w_{rf} \times P_{rf})$
if $S_{ensemble} \geq 0.5$ **then**
  $\hat{y} \leftarrow 1$ {Adversarial Detected}
else
  $\hat{y} \leftarrow 0$ {Clean Detected}
end if
return $\hat{y}$

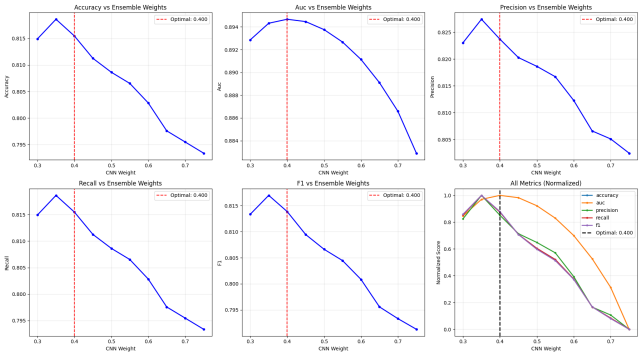


Fig. 2. Hyperparameter optimization via grid search. The graphs show the evolution of Accuracy, AUC, Precision, Recall, and F1-score as a function of CNN weight. The optimal configuration is identified at $w_{cnn} = 0.400$, favoring the statistical robustness of the RF.

1) Convexity and stability analysis: The curves in Fig. 2 exhibit a distinct convex behavior. At $w_{cnn} = 0$ (pure RF) or $w_{cnn} = 1$ (pure CNN), Accuracy is suboptimal when using only the RF ($w_{\text{cnn}} = 0$) or only the CNN ($w_{\text{cnn}} = 1$). The stability of the F1-score around the peak (0.4) indicates that the ensemble is robust to minor fluctuations in hyperparameter tuning. This "sweet spot" at 0.4 confirms that while CNNs are essential for image understanding, the statistical "sanity check" provided by the RF is more reliable for detecting the specific artifacts of PGD attacks.

B. Performance Evaluation and Robustness

1) ROC analysis and false positive dynamics: The Receiver Operating Characteristic (ROC) curves provide a comprehensive view of the trade-off between sensitivity (True Positive Rate) and specificity (1 - False Positive Rate). Fig. 3 presents the overall comparison on the mixed dataset.

The Hybrid Ensemble (Green curve) achieves an Area Under the Curve (AUC) of **0.891**. More importantly, looking at the "knee" of the curve (the top-left corner), the Ensemble rises much sharper than the individual models. This implies that for a fixed low False Positive Rate (e.g., 0.05), the Ensemble offers a significantly higher True Positive Rate. In a clinical context, this translates to fewer false alarms while maintaining high detection capability.

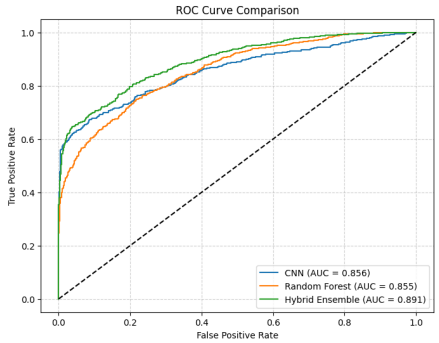


Fig. 3. Global ROC curve comparison on the mixed dataset. The hybrid ensemble (AUC=0.891) dominates the individual models, exhibiting a better convex hull.

The robustness is further highlighted when analyzing specific perturbation strengths separately (see Fig. 4).

- At $\epsilon = 0.003$ (left): The Ensemble maintains an AUC of 0.863 compared to 0.842 for the CNN. This demonstrates resilience even against extremely subtle attacks.
- At $\epsilon = 0.006$ (right): The performance gap widens significantly. The Ensemble achieves an impressive **AUC of 0.919**, while the CNN reaches 0.901. This suggests a complementary effect: as the attack becomes stronger, it becomes easier for the RF to detect (due to increased statistical distortion), compensating for the CNN's increased error rate. This aligns with optimization strategies discussed by Vargas et al. [55].

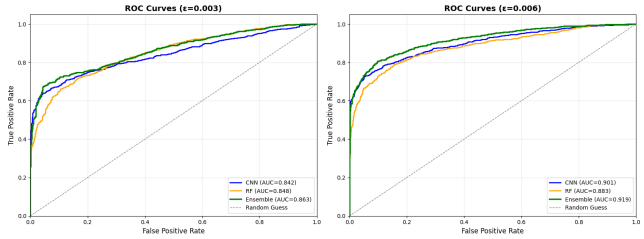


Fig. 4. ROC curves at specific epsilon values ($\epsilon = 0.003$ left, $\epsilon = 0.006$ right). Note the stability of the green curve (Ensemble) across perturbation strengths.

2) Comparison with baseline defenses: To validate the novelty and efficacy of our approach, we compared it against a standard Adversarial Training (Baseline) method, where the CNN is retrained on PGD examples. Fig. 5 displays the heatmaps for Accuracy and AUC.

The results indicate a substantial performance difference. Notably, at $\epsilon = 0.006$, the Hybrid Ensemble achieves an Accuracy of **0.838** and AUC of **0.919**, whereas the Adversarial Training baseline significantly lags behind with an Accuracy of 0.712 and AUC of 0.700. **Theoretical Interpretation:** Adversarial training tends to reshape the decision boundary of the CNN around the specific ϵ-ball seen during training. If the test attack varies slightly (in step size or noise distribution), the boundary is crossed. Our ensemble, however, utilizes properties (like Kurtosis and FFT energy) that shift monotonically with noise addition, providing a more generalized boundary [43].

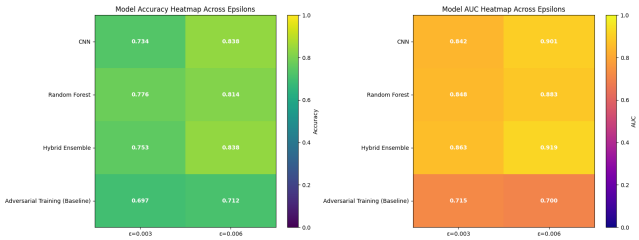


Fig. 5. Performance comparison heatmaps (Accuracy and AUC). The Hybrid Ensemble significantly outperforms the Adversarial Training baseline, especially at higher ϵ, indicating superior generalization.

C. Statistical Significance and Variance Reduction

To ensure the reported improvements are scientifically valid, we computed 95% Confidence Intervals (CI) for all key metrics. Fig. 6 presents these results.

The error bars indicate that the Hybrid Ensemble's performance lower bound is consistently higher than the mean performance of the individual models. For example, in AUC, the ensemble's lower bound (≈ 0.88) is higher than the CNN's upper bound (≈ 0.86). This confirms that the observed gains are statistically significant. Furthermore, the CI for the Ensemble is notably tighter than for the RF alone. This is a classic benefit of bagging and ensemble methods: **Variance Reduction**. By combining two uncorrelated error sources (spatial error from CNN, statistical error from RF), the overall variance of the prediction decreases, leading to a more reliable clinical tool.

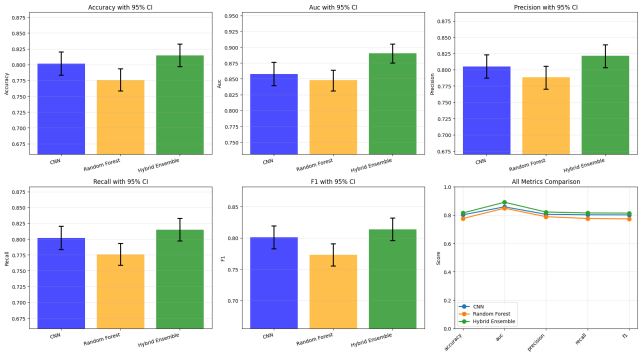


Fig. 6. Statistical significance analysis. Bar charts show performance metrics with 95% Confidence Intervals. The non-overlapping intervals in AUC and Precision confirm statistical superiority.

D. Clinical Impact and Specificity

In a clinical setting, the cost of errors is asymmetric. Minimizing False Positives (FP) is crucial to avoid "alarm fatigue" among radiologists and to prevent unnecessary biopsies [42]. Fig. 7 shows the confusion matrices for the CNN, Random Forest, and Hybrid Ensemble, alongside the score distribution.

The Hybrid Ensemble achieves the lowest number of False Positives (**108**), compared to 138 for the CNN and 113 for the RF. This represents a significant improvement in specificity.

1) Reduction in false negatives: The Ensemble missed only 244 adversarial samples compared to 313 for the RF. A False Negative in this context implies a hacked image enters the diagnostic workflow, potentially leading to a misdiagnosis (e.g., hiding a tumor). Within the scope of the IRMA dataset, the Hybrid Ensemble provides a safer buffer against this risk. However, we note that external validation on independent datasets is necessary to confirm these findings for general clinical deployment [59].

2) Score separation: The histogram (bottom center) shows a bimodal distribution with minimal overlap, indicating high confidence in predictions.

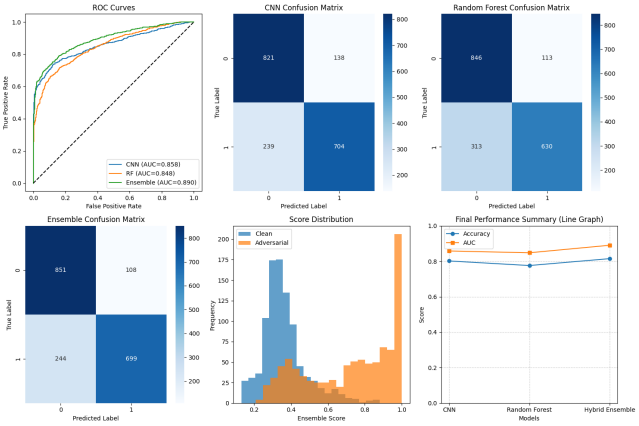


Fig. 7. Clinical impact analysis: Confusion matrices, ROC curve, score distribution, and summary metrics. The Hybrid Ensemble (bottom left matrix) minimizes false positives (108), while maintaining high sensitivity.

E. Computational Cost Analysis and Deployment

A common criticism of ensemble methods is the increased computational overhead and latency. However, our analysis (see Fig. 8) reveals a counter-intuitive but favorable result. The inference time per sample for the Hybrid Ensemble is **2.74 ms**, which is significantly faster than the standalone CNN baseline (**13.04 ms**).

This result can be explained by the architecture of the solution. The CNN used in the ensemble is optimized for feature extraction and does not require the heavy, deep backbones often used in standalone classification. Combined with the extremely fast inference of the Random Forest (0.74 ms), the total pipeline remains lightweight. This low latency confirms the method's suitability for real-time deployment in high-throughput hospital workflows or even on edge devices in portable X-ray machines.

V. Conclusion and Future Work

This study presented a Hybrid Ensemble defense for medical X-ray imaging that addresses the vulnerabilities of Deep Learning models to adversarial attacks. By synergizing spatial deep learning with statistical random forest analysis, we achieved a peak AUC of 0.919 under strong perturbation ($\epsilon = 0.006$), significantly outperforming standard adversarial training.

Our extensive analysis confirms that:

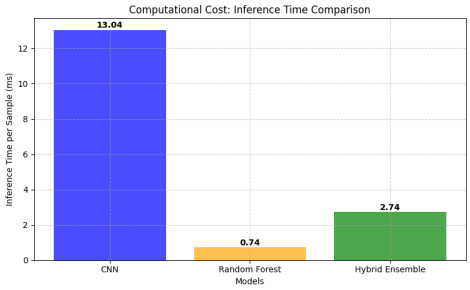


Fig. 8. Inference time per sample (ms). The hybrid solution is computationally efficient (2.74 ms), enabling real-time application.

- Complementarity matters: Combining differentiable models (CNN) with non-differentiable statistical models (RF) creates a defense that is empirically more resistant to standard gradient descent attacks.
- Statistical invariants: Adversarial noise, while visually imperceptible, disrupts the statistical moments (Kurtosis, Skewness) and spectral energy of medical images, making them detectable.
- Clinical viability: The method reduces false positives and operates with negligible latency (2.74 ms), making it suitable for clinical deployment.

Future work will explore the integration of Explainable AI (XAI) techniques [51], while addressing the potential pitfalls and 'false hopes' associated with current saliency methods [50], such as attention mechanisms [52], to provide radiologists with visual heatmaps indicating exactly which parts of the image triggered the adversarial alert [41]. We also plan to extend this framework to other modalities such as MRI and CT scans to validate its cross-domain generalization.

Future work will also address adaptive adversaries, investigating whether noise patterns can be optimized to preserve statistical moments, thereby potentially bypassing the Random Forest component.

Acknowledgment

This research received no external funding. The authors would like to thank the University of Mohammed First for providing the computational resources.

References

[1] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115-118, 2017.

[2] A. Elhazmi, N. Almutairi, and H. Alghamdi, "Brain Tumor Classification using Deep Learning Techniques," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 2, 2021.

[3] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, ... and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60-88, 2017.

[4] M. A. Khan, T. Akram, M. Sharif, K. Javed, M. Rashid, and S. A. C. Bukhari, "An integrated framework of skin lesion detection and recognition through saliency method and optimal deep neural network features selection," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 4, 2020.

[5] E. J. Topol, "High-performance medicine: the convergence of human and artificial intelligence," *Nature Medicine*, vol. 25, no. 1, pp. 44-56, 2019.

[6] P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, ... and A. Y. Ng, "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," *arXiv preprint arXiv:1711.05225*, 2017.

[7] D. Shen, G. Wu, and H. I. Suk, "Deep learning in medical image analysis," *Annual review of biomedical engineering*, vol. 19, pp. 221-248, 2017.

[8] G. A. Kaissis, M. R. Makowski, D. Rückert, and R. F. Braren, "Secure, privacy-preserving and federated machine learning in medical imaging," *Nature Machine Intelligence*, vol. 2, no. 6, pp. 305-311, 2020.

[9] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, and H. Müller, "Causability and explainability of artificial intelligence in medicine," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 9, no. 4, e1312, 2019.

[10] J. He, S. L. Baxter, J. Xu, J. Xu, X. Zhou, and K. Zhang, "The practical implementation of artificial intelligence technologies in medicine," *Nature Medicine*, vol. 25, no. 1, pp. 30-36, 2019.

[11] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[12] S. G. Finlayson, J. D. Bowers, J. Ito, J. L. Zittrain, A. L. Beam, and I. S. Kohane, "Adversarial attacks against medical deep learning systems," *Science*, vol. 363, no. 6433, pp. 1287-1289, 2019.

[13] F. Hussain, R. Hussain, S. A. Hassan, and E. Hossain, "Machine learning in IoT security: Current solutions and future challenges," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 9, 2019.

[14] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[15] X. Ma, Y. Niu, L. Gu, Y. Wang, Y. Zhao, J. Bailey, and F. Lu, "Understanding adversarial attacks on deep learning based medical image analysis," *Pattern Recognition*, vol. 110, p. 107658, 2021.

[16] K. D. Apostolidis and G. A. Papakostas, "A survey on adversarial deep learning in medical image analysis," *Medical Physics*, vol. 48, no. 8, pp. 4274-4300, 2021.

[17] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[18] M. Vento, "Adversarial attacks in medical imaging: A new challenge for reliability," *Artificial Intelligence in Medicine*, vol. 125, p. 102246, 2022.

[19] M. Paschali, S. Conjeti, F. Navarro, and N. Navab, "Generalizability vs. robustness: adversarial examples for medical imaging," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018*, Springer, Cham, pp. 493-501, 2018.

[20] H. Hirano, A. Minagi, and K. Takemoto, "Universal adversarial attacks on deep neural networks for medical image classification," *BMC Medical Imaging*, vol. 21, no. 1, pp. 1-13, 2021.

[21] A. Rahim, A. Maqbool, and T. Rana, "Monitoring and Analysis of Deep Learning Based Medical Imaging," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, 2021.

[22] B. E. Bejnordi, et al., "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *JAMA*, vol. 318, no. 22, pp. 2199-2210, 2017.

[23] C. J. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado, and D. King, "Key challenges for delivering clinical impact with artificial intelligence," *BMC Medicine*, vol. 17, no. 1, pp. 1-9, 2019.

[24] U. Ozbulak, A. Van Messem, and W. De Neve, "Impact of adversarial examples on deep learning models for biomedical image segmentation," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019*, Springer, Cham, pp. 300-308, 2020.

[25] I. E. Olatunji and C. Cheng, "Medical Image Analysis using Deep Learning: A Systematic Review," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 8, 2019.

[26] A. S. Al-Waisy, S. Al-Fahdawi, M. A. Mohammed, K. H. Abdulkareem, S. A. Mostafa, M. A. Maashi, ... and D. A. Ibrahim, "COVID-CheXNet: Hybrid deep learning framework for identifying COVID-19 virus in chest X-rays images," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, 2020.

[27] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, no. 4, pp. 611-629, 2018.

[28] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5-32, 2001.

[29] E. Garyfallidis, et al., "Machine learning for medical imaging: Methodological failures and recommendations for the future," *npj Digital Medicine*, vol. 3, no. 1, p. 98, 2020.

[30] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226-239, 1998.

[31] T. Deselaers, H. Müller, P. Clough, H. Ney, and T. M. Deserno, "The CLEF 2008 image retrieval track," in *Evaluating Systems for Multilingual and Multimodal Information Access*, Springer, Berlin, Heidelberg, pp. 523-530, 2008.

[32] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[34] A. Abbas, M. M. Abdelsamea, and M. M. Gaber, "Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 3, 2021.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[37] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and Systems Magazine*, vol. 6, no. 3, pp. 21-45, 2006.

[38] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*, Springer, Berlin, Heidelberg, pp. 1-15, 2000.

[39] M. Abadi, et al., "Tensorflow: A system for large-scale machine learning," in *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pp. 265-283, 2016.

[40] F. Pedregosa, et al., "Scikit-learn: Machine learning in Python," *The Journal of machine learning research*, vol. 12, pp. 2825-2830, 2011.

[41] R. Tomsett, D. Harborne, J. Chakraborty, P. Gurram, and A. Preece, "Sanity checks for saliency metrics," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 6021-6029, 2020.

[42] J. Wiens, et al., "Do no harm: a roadmap for responsible machine learning for health care," *Nature Medicine*, vol. 25, no. 9, pp. 1337-1340, 2019.

[43] L. Chen, P. Bentley, K. Mori, K. Misawa, M. Fujiwara, and D. Rueckert, "Self-supervised learning for medical image analysis using image context restoration," *Medical image analysis*, vol. 58, p. 101539, 2019.

[44] J. Zhang and K. C. C. Chan, "Defense against adversarial attacks in medical image classification using deep learning," *IEEE Access*, vol. 9, pp. 131926-131938, 2021.

[45] I. Naseer, T. Masood, S. Akram, and A. Jaffar, "Breast Cancer Detection using Deep Learning: A Comparative Study," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 1, 2020.

[46] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine learning*, vol. 51, no. 2, pp. 181-207, 2003.

[47] S. A. Taghanaki, K. Abhishek, J. P. Cohen, J. Cohen-Adad, and G. Hamarneh, "Deep semantic segmentation of natural and medical images: a review," *Artificial Intelligence Review*, vol. 54, no. 1, pp. 137-178, 2021.

[48] K. Ren, T. Zheng, Z. Qin, and X. Liu, "Adversarial attacks and defenses in deep learning," *Engineering*, vol. 6, no. 3, pp. 346-360, 2020.

[49] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241-259, 1992.

[50] M. Ghassemi, L. Oakden-Rayner, and A. L. Beam, "The false hope of current approaches to explainable artificial intelligence in health care," *The Lancet Digital Health*, vol. 3, no. 11, pp. e745-e750, 2021.

[51] A. B. Arrieta, et al., "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82-115, 2020.

[52] M. Al-Shabi, B. L. Lan, and W. Y. Chan, "Lung Nodule Detection using Attention Mechanisms in Chest X-Rays," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 3, 2019.

[53] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, pp. 39-57, 2017.

[54] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *arXiv preprint arXiv:1611.01236*, 2017.

[55] J. Vargas, S. Mosavi, and L. R. Ruiz, "Deep Learning Optimization in Medical Imaging: A Survey," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, 2019.

[56] L. Sun, J. Wang, Q. Huang, K. Ding, and H. Huang, "Adversarial attack and defense on graph data: A survey," *IEEE Open Journal of the Computer Society*, vol. 1, pp. 241-261, 2020.

[57] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE symposium on security and privacy (SP)*, pp. 582-597, 2016.

[58] F. Liao, M. Liang, Y. Dong, T. Pang, X. Hu, and J. Zhu, "Defense against adversarial attacks using high-level representation guided denoiser," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1778-1787, 2018.

[59] M. B. McDermott, et al., "Reproducibility in machine learning for health," *arXiv preprint arXiv:1907.01463*, 2019.

AUTHORS' PROFILE

Yassine Chahid received the M.Sc. degree in science and technology from the Faculty of Science and Technology, Settat, Morocco, in 2014, and the Ph.D. degree in mathematics and computer science from the Faculty of Sciences, Mohammed First University, Oujda, Morocco, in 2022. Since 2014, he has worked with several private-sector companies, where he has held technical and research-oriented positions. He is currently a Technical Lead specializing in artificial intelligence and cybersecurity solutions. His research interests include federated learning, information security, cryptography, and distributed systems.

Anas Chahid received the Engineering degree in computer science from the National School of Applied Sciences (ENSA), Oujda, Morocco. He is currently a Software Engineer and a Ph.D. student in medical artificial intelligence. His work focuses on the design of intelligent and customized software systems. His professional interests include artificial intelligence, data analytics, software engineering, and healthcare applications.

Ismail Chahid received the DUT degree in information technology from the École Supérieure de Technologie, Oujda, Morocco, in 2008, the B.Sc. degree in IT management from the Faculty of Polydisciplinary Studies, Tétouan, Morocco, in 2009, and the M.Sc. degree in business intelligence from the Faculty of Science and Technology, Béni Mellal, Morocco, in 2012. He is currently the Head of the IT Department at the Faculty of Medicine and Pharmacy of Oujda, where he has been working since 2019. His professional interests include business intelligence, data warehousing, information systems management, and software engineering.

Aissa Kerkour El Miad received the M.Sc. degree in operational research and informatics and the Ph.D. degree in computer science from Mohammed First University, Oujda, Morocco. He is currently a Professor with the Department of Computer Science, Faculty of Sciences, Mohammed First University, Oujda. His research interests include image processing, artificial intelligence, high-performance computing, and data mining.

A Transformer-Based Approach for Multimodal Arabic Sentiment Analysis

Ayoub BEN CHEIKHI, EL Habib NFAOUI
L3IA Laboratory, Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco

***Abstract*—The Multimodal Sentiment Analysis (MSA) landscape for Arabic content is strikingly underexplored, mainly due to limited datasets and a lack of robust integration methods across text, audio, and image. While transformer-based models like MarBERT and ArBERT achieve strong results on Arabic text, most research remains unimodal and does not fully exploit multimodal synergy. In this work, we propose a three-fold approach for Arabic MSA. First, we finetune robust transformers for each modality, namely ViT, MarBERT, and HuBert for image, Text, and Audio, respectively. Second, we perform an early feature fusion. Third, we use classifiers for sentiment prediction. On the recent Ar-MuSA benchmark released on 2025, our tri-modal fusion system, achieves state-of-the-art performance (F1=0.7756, Accuracy=0.7759), significantly exceeding the multimodal models benchmarked on the Ar-MuSa dataset, as well as the unimodal and bimodal methods. This demonstrates that comprehensive tri-modal fusion and thoughtful classifier selection are essential for accurate, human-centric Arabic sentiment analysis.**

***Keywords*—Arabic sentiment analysis; multimodal learning; feature fusion; machine learning; early fusion**

I. Introduction

The Arabic digital space has grown rapidly in recent years [1], [2], [3], [4], [5]. Social media platforms, video-sharing services, and online forums allow users to express opinions and emotions through text, speech, and images. As a result, sentiment is increasingly communicated in a multimodal manner [6], [7], [8]. This evolution raises new challenges for artificial intelligence (AI) [9] and natural language processing (NLP) [10], which must go beyond textual analysis to understand sentiment more accurately [11]. Early work in Arabic sentiment analysis mainly focused on text written in Modern Standard Arabic (MSA) [12], [13], [14], [2]. These approaches often relied on machine learning techniques and handcrafted features [15]. The introduction of transformer-based models has significantly improved text-based sentiment analysis [16], [17]. However, Arabic remains a complex language due to its rich morphology, dialectal variation, and informal writing styles commonly used in social media. Importantly, sentiment is not expressed through text alone. Spoken language conveys emotion through prosody, such as intonation and rhythm. Visual content also plays a key role, as facial expressions and contextual cues often reveal affective states. Ignoring these signals leads to incomplete sentiment understanding, particularly in socially rich Arabic media.

While multimodal sentiment analysis has been widely explored for English [18], [6], [11], research in Arabic remains limited. One major reason is the lack of large, well-annotated multimodal datasets. Another challenge lies in designing effective fusion strategies that can integrate heterogeneous modalities. The release of the Ar-MuSA [19] dataset addresses part of this gap by providing aligned text, audio, and image data for Arabic sentiment analysis.

In this work, we investigate multimodal Arabic sentiment analysis using transformer-based feature extraction for text, speech, and images. We adopt hybrid pipelines that combine deep pretrained representations with machine learning classifiers. In particular, we focus on early feature-level fusion and systematically compare unimodal, bi-modal, and tri-modal configurations.

Beyond technical contributions, this research aims to support a more context-aware understanding of Arabic digital content. Robust multimodal sentiment analysis is essential for applications such as content moderation, opinion mining, and social media analysis. It is also especially valuable in low-resource and highly variable linguistic environments such as Arabic.

The main contributions of this study are as follows:

- We provide a systematic evaluation of multimodal sentiment analysis for Arabic using text, audio, and image modalities from the Ar-MuSA dataset.
- We analyze the impact of early fusion strategies across uni-modal, bi-modal, and tri-modal settings.
- We report the state-of-the-art results and highlight practical insights for future research in Arabic multimodal sentiment analysis.

The remainder of this study is structured as follows: Section II surveys foundational research in Arabic sentiment analysis and multimodal approaches. Section III presents the details of our proposed multimodal sentiment analysis pipeline, including unimodal feature extraction, fusion methods, and classification strategies. Section IV describes the Ar-MuSA dataset, outlines preprocessing procedures, and details the experimental protocol. Section V reports and analyzes the experimental results across unimodal, bimodal, and tri-modal configurations. Section V-B provides a direct comparison of our approach with existing Ar-MuSA benchmarks and state-of-the-art results. Section VI offers an in-depth interpretation of key findings and their broader implications. Finally, Section VII concludes the study and suggests possible directions for future research in Arabic multimodal sentiment analysis.

II. Related Work

Arabic Sentiment Analysis (ASA) has evolved from classical machine learning pipelines that rely on handcrafted features to transformer-based and multimodal approaches that integrate textual, acoustic, and visual signals. This section reviews prior

work in line with the thematic structure adopted in this study: 1) Arabic sentiment analysis, 2) multimodal Arabic sentiment analysis, 3) multimodal machine learning foundations, and 4) transformer-based classification. We conclude by synthesizing the main limitations reported in the literature and highlighting the research gap addressed by this work.

A. Arabic Sentiment Analysis

Contemporary ASA research consistently emphasizes the linguistic complexity of Arabic—including rich morphology, dialectal variation, and informal writing styles—as a central challenge for robust sentiment modeling. In [15], the authors compare conventional methods such as SVM and Naïve Bayes with deep learning and transformer-based approaches (e.g., LSTM and AraBERT), reporting that AraBERT achieves superior performance (above 88% accuracy in their reviewed settings) while noting persistent limitations related to corpus scarcity and dialectal diversity.

Several application-driven studies confirm that classical machine learning approaches remain competitive in domain-specific settings when datasets are appropriately curated. Musleh et al. investigate Arabic YouTube comments and evaluate six classifiers for binary sentiment polarity; their experiments show that Naïve Bayes achieves the best performance (94.62% accuracy and MCC of 91.46%), demonstrating the continued relevance of lightweight models for constrained ASA tasks [20]. In another social-media context, Musleh et al. propose a machine-learning approach to detect depression from Arabic Twitter posts, formulating a multi-class problem (depressed, non-depressed, neutral) and reporting that Random Forest attains 82.39% accuracy [21]. This work illustrates that extending sentiment-related analysis to mental-health signals introduces additional complexity in labeling, feature robustness, and generalization.

In the education domain, [22] construct an Arabic dataset from student satisfaction surveys and compare classical models with the AraBERT transformer, achieving an accuracy of 78% and highlighting the limited availability of Arabic sentiment datasets tailored to educational evaluation. Overall, these studies collectively show that: 1) dataset design and domain coverage are crucial to performance, and 2) pretrained transformers often provide gains, but robustness across domains and dialects remains an open challenge.

B. Multimodal Arabic Sentiment Analysis

Although text-only ASA dominates the literature, multimodal Arabic sentiment analysis (MuSA) is increasingly studied due to the prevalence of sentiment expression through voice, facial cues, and contextual imagery in videos and social platforms. Early work by [23] introduces a direction toward Arabic multimodal sentiment analysis by presenting a dataset and exploring SVM-based classification, arguing that multimodality can better capture sentiment signals in heterogeneous social media content. In [24], the authors analyze Arabic YouTube videos by integrating acoustic and facial features and testing different classifiers (including SVM and neural networks), achieving an overall accuracy of 76%, which underlines both the promise of multimodal cues and the difficulty of robust inference in real-world settings.

More recent efforts emphasize improved fusion strategies across text, audio, and vision. In [25], the authors propose enhanced video analytics through multi-level fusion of textual, auditory, and visual information, reporting more than 94% effectiveness in multi-dialect Arabic videos. Their results suggest that carefully designed fusion mechanisms can significantly improve performance beyond unimodal baselines, particularly when dialectal and expressive cues are important.

Dataset construction remains a key bottleneck for MuSA. In [26], the authors introduce and validate an Arabic multimodal dataset using transformer-based components and word-alignment techniques, but they also emphasize limitations in dataset scale and coverage. Addressing the need for standardized benchmarking, [27] propose Ar-MuSA, an open-source multimodal benchmark dataset and evaluation framework that combines text, audio, and visual elements, demonstrating that multimodal approaches often outperform unimodal strategies when validated with advanced models such as MarBERT and HuBERT. In low-resource settings, [28] further introduce UniTextFusion, an early-fusion strategy that textualizes non-text modalities to integrate them into Arabic language models; using LoRA-based fine-tuning, they report improvements of up to 34% in F1-score compared to existing baselines. These findings highlight the importance of both dataset availability and parameter-efficient multimodal fusion for Arabic.

C. Multimodal Application Using Machine Learning

General multimodal machine learning (MML) research provides the conceptual foundation for MuSA, particularly for understanding how to represent, align, and fuse heterogeneous modalities. In [29], the authors provide a comprehensive survey and propose a taxonomy organizing MML challenges into representation, translation, alignment, fusion, and co-learning. This taxonomy is especially relevant to Arabic multimodal sentiment tasks where temporal alignment and noisy cross-modal signals can substantially affect performance.

In [30], the authors extend these foundations by framing principles, challenges, and open questions in multimodal machine learning, covering representation, alignment, reasoning, generation, transference, and uncertainty quantification. Their discussion suggests that multimodal systems should be evaluated not only by accuracy but also by robustness, transferability, and reliability—criteria that are particularly important in Arabic contexts with high dialect variability and diverse media conditions.

Multimodal modeling is also increasingly applied to pragmatic phenomena closely related to sentiment, such as sarcasm. In [31], the authors present a multimodal Arabic sarcasm detection approach using text, audio, and image features on the Ar-MuSA dataset; leveraging BiLSTM, CNN, and ResNet-50 encoders, they achieve 97% accuracy and show clear gains over unimodal settings. This supports the argument that multimodality can capture implicit affective and pragmatic cues that are difficult to infer from text alone.

D. Transformer Application in Classification

Transformers have become the dominant paradigm for classification in both NLP and computer vision, and they

have strongly influenced both unimodal ASA (e.g., AraBERT-like encoders) and multimodal systems that depend on transformer backbones or transformer-compatible representations. In [32], the authors survey transformer-based text classification, proposing an expanded taxonomy and discussing performance, cost, safety, and bias considerations; importantly, they argue that large transformer models are not universally superior without careful consideration of domain and deployment constraints .

In vision, [33] review vision transformers for image classification, tracing the evolution from ViT to more advanced variants and comparing them to CNNs, while emphasizing open challenges such as data efficiency and computational demands. These insights motivate efficiency-aware multimodal designs, since MuSA systems often require strong unimodal encoders in each modality.

Evidence also indicates that smaller, well-adapted pretrained models can compete with much larger models in classification tasks. In [34], the authors evaluate pretrained language models for mathematical problem classification and show that compact transformer models can match larger LLMs with a macro-F1 of 0.8685, highlighting the practical value of targeted fine-tuning and cost-effective architectures. This aligns with recent MuSA work that favors parameter-efficient tuning methods such as LoRA [28].

E. Research Gap

Despite substantial progress, the reviewed literature reveals several persistent gaps. First, the availability of large-scale, diverse, and well-aligned Arabic multimodal datasets remains limited; while Ar-MuSA advances benchmarking [27], other dataset efforts still report constraints in scale and coverage [26]. Second, robustness across dialects, domains, and informal language remains a central challenge in ASA [15]; strong results in a specific domain (e.g., YouTube) may not transfer to other contexts such as education or mental health [20], [21], [22]. Third, although recent low-resource fusion methods show promise [28], the design space of fusion strategies (early/late/hybrid fusion and modality interaction modeling) has not been consistently evaluated under comparable settings for Arabic. Finally, many studies emphasize accuracy-focused evaluation, while broader multimodal literature argues for more systematic analysis of reliability, uncertainty, and failure modes [30].

These limitations motivate the need for an approach that: 1) better accounts for Arabic linguistic variability, 2) leverages complementary multimodal cues through principled and efficient fusion, and 3) is evaluated with robustness and generalization analysis beyond single-dataset performance.

III. Proposed Approach

The model architecture proposed in this research addresses the task of Arabic multimodal sentiment analysis through a principled integration of information from textual, visual, and audio modalities. The system leverages the strengths of transformer-based models for each modality, and employs a structured pipeline for effective fusion and classification. The process is illustrated in Fig. 1.

A. Unimodal Representation Learning

For each modality—text, audio, and image—we employ state-of-the-art transformer architectures, each fine-tuned on the Ar-MuSA dataset to extract robust, modality-specific embeddings:

$$\mathbf{e}_{\text{text}} = \text{MarBERT}(\mathbf{x}_{\text{text}}; \theta_{\text{text}}) \tag{1}$$

$$\mathbf{e}_{\text{audio}} = \text{HuBERT}(\mathbf{x}_{\text{audio}}; \theta_{\text{audio}}) \tag{2}$$

$$\mathbf{e}_{\text{image}} = \text{ViT}(\mathbf{x}_{\text{image}}; \theta_{\text{image}}) \tag{3}$$

where, $\mathbf{x}_{\text{text}}$, $\mathbf{x}_{\text{audio}}$, and $\mathbf{x}_{\text{image}}$ are the respective raw inputs, θ denotes the set of fine-tuned parameters specific to each model, and $\mathbf{e}*$ represents the resulting embedding vectors.

This approach is motivated by several key considerations. First, using dedicated transformer models for each modality allows us to capitalize on their architectural strengths, which have been empirically demonstrated across diverse tasks in natural language processing, speech analysis, and computer vision. Each model—MarBERT, HuBERT, and ViT—incorporates design choices and pretraining strategies that make it uniquely effective for its target data type.

Second, fine-tuning on the Ar-MuSA dataset ensures that each model adapts to the linguistic, acoustic, and visual characteristics specific to the domain of interest. This step enhances the models' ability to capture salient and contextually relevant patterns, which is especially important when dealing with datasets that diverge from widely available pretraining corpora.

Third, transformer-based encoders yield highly expressive and context-aware representations. These models effectively model complex dependencies: MarBERT excels at capturing nuanced semantics in Arabic text, HuBERT models both local and long-range speech patterns, and ViT efficiently encodes spatial and hierarchical information in images. The unimodal embeddings produced are therefore rich in both local detail and global structure.

Finally, a wealth of recent literature highlights the benefits of strong unimodal feature extraction as a precursor to multimodal learning [35], [36]. Isolating and optimizing each modality at the representation level not only improves single-modality interpretability, but also provides a robust foundation for effective cross-modal fusion, leading to superior performance in downstream multimodal tasks.

Through this methodical unimodal representation learning scheme, we ensure that each input stream contributes maximally informative, discriminative, and complementary features to the overall multimodal system.

B. Multimodal Feature Fusion

In multimodal learning, feature fusion refers to the process of integrating information derived from multiple input modalities—such as text, audio, and images—into a single, unified representation suitable for downstream tasks. Among

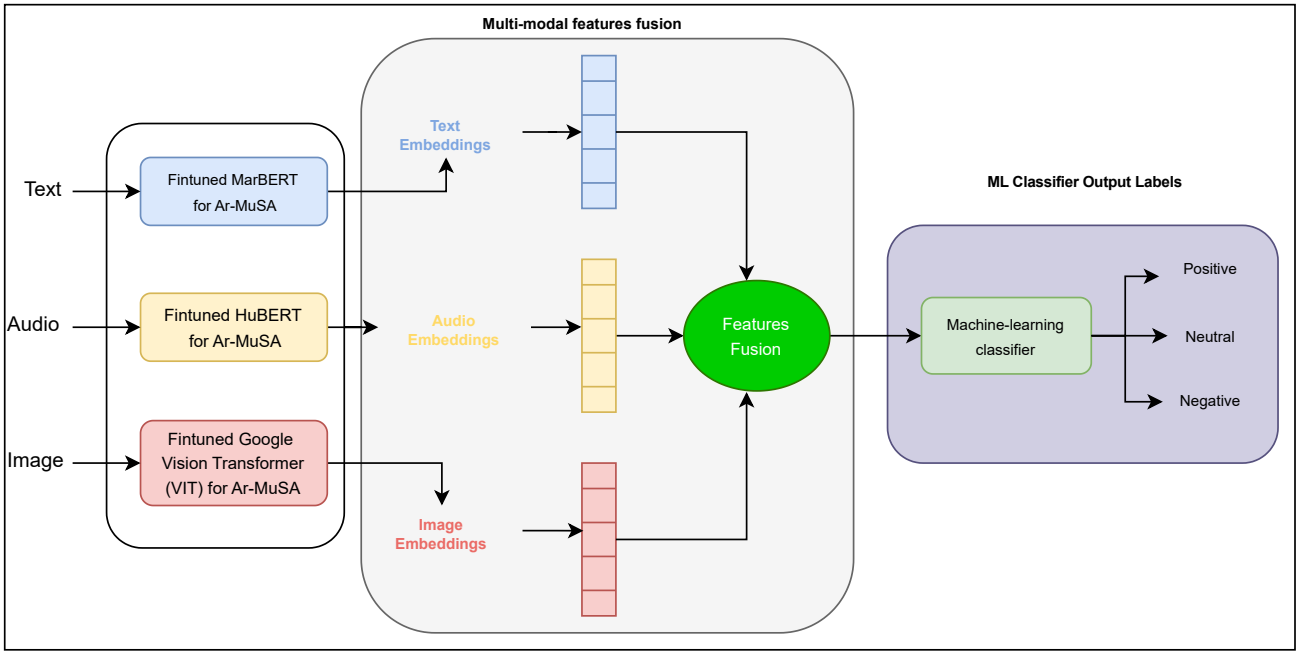


Fig. 1. Overview of the proposed multimodal Arabic sentiment analysis pipeline. Each modality is encoded by a dedicated finetuned transformer (MarBERT for text, HuBERT for audio, ViT for images). The resulting embeddings are concatenated to produce a joint multimodal feature vector, which is then used as input to a machine learning classifier (SVM) for sentiment prediction (positive, neutral, negative).

various fusion strategies, early fusion (also known as feature-level fusion) is a widely adopted approach. In early fusion, raw or low-level feature embeddings from each modality are combined at the initial stages of the model pipeline, usually before any decision-level processing or prediction.

Formally, let $\mathbf{e}_{\text{text}}$, $\mathbf{e}_{\text{audio}}$, and $\mathbf{e}_{\text{image}}$ denote the learned feature representations for the text, audio, and image modalities, respectively. Under early fusion, we concatenate these feature vectors to obtain a single, comprehensive embedding:

$$\mathbf{e}_{\text{fused}} = [\mathbf{e}_{\text{text}}; \mathbf{e}_{\text{audio}}; \mathbf{e}_{\text{image}}] \tag{4}$$

Here, $[\cdot\,;\cdot;\,\cdot]$ denotes straightforward vector concatenation.

The principal rationale for employing early fusion is its capacity to preserve the diversity and richness of information from each input modality. Unlike late fusion approaches, which combine information at the decision level and potentially disregard modality-specific patterns, early fusion allows the model to exploit nuanced interactions between modalities at a granular level. This aligns with empirical findings in recent studies, which demonstrate that feature-level fusion retains complementary signals and facilitates deeper cross-modal synergy, ultimately enhancing performance in complex multimodal learning tasks [37], [38].

C. Multimodal Sentiment Classification

Following feature extraction and fusion, the joint multimodal embedding $\mathbf{e}_{\text{fused}}$ serves as the input to a Support Vector Machine (SVM) classifier, which is tasked with sentiment prediction across three discrete categories: positive, neutral, and negative. Specifically, the SVM learns a discriminative function over the fused feature space:

$$\hat{y} = f_{\text{SVM}}(\mathbf{e}_{\text{fused}}; \mathcal{W}) \tag{5}$$

where, $\mathcal{W}$ denotes the set of weights, thresholds, and bias terms optimized during supervised training on labeled data.

The decision framework is instantiated as follows: for each training instance i, the input tuple $(\mathbf{x}^{(i)}_{\text{text}}, \mathbf{x}^{(i)}_{\text{audio}}, \mathbf{x}^{(i)}_{\text{image}})$ —processed into the fused embedding—yields a sentiment prediction:

$$(\mathbf{x}^{(i)}_{\text{text}}, \mathbf{x}^{(i)}_{\text{audio}}, \mathbf{x}^{(i)}_{\text{image}}) \rightarrow \hat{y}^{(i)} \in \{\text{positive, neutral, negative}\} \tag{6}$$

The choice of SVM for this classification task is motivated by several factors. First, SVMs are highly effective in high-dimensional settings such as those arising from feature-level fusion of transformer-based embeddings. Their emphasis on maximizing the margin between decision boundaries lends itself well to the complex, nonlinear distributions often present in multimodal data. Second, SVMs are robust to overfitting, especially in cases where the number of training samples is modest compared to the feature space dimensionality—a common scenario in multimodal sentiment datasets. The ability to use various kernel functions (e.g., linear, RBF) further enhances the SVM's flexibility to capture intricate cross-modal relationships within the fused space.

Furthermore, SVMs offer solid interpretability of the learned margins and support vectors, facilitating post-hoc anal-

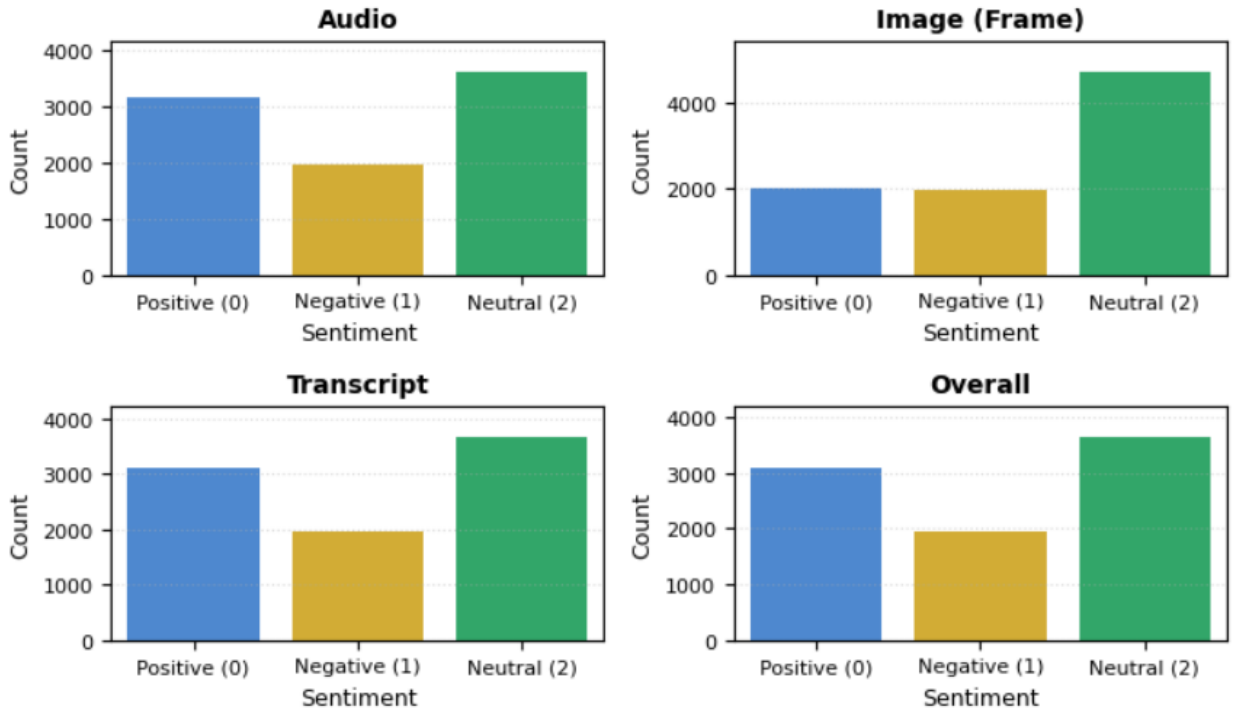


Fig. 2. Distribution of sentiment categories across modalities in the Ar-MuSA dataset.

ysis and potentially providing insights into which modalities or feature combinations most influence classification decisions. This interpretability is important not only from a research standpoint but also for transparency in consequential real-world applications such as sentiment analysis of media content or public communication.

Empirical studies corroborate the effectiveness of SVMs as final classifiers in multimodal pipelines [39], demonstrating competitive or superior performance compared to more parameter-intensive neural classifiers, especially when the preceding stages of the pipeline produce strong, information-rich feature embeddings.

In summary, the SVM-based multimodal sentiment classifier leverages the joint, complementary information captured in $\mathbf{e}_{\text{fused}}$ to robustly predict sentiment, fully exploiting the synergy of textual, acoustic, and visual modalities within a principled maximal-margin framework.

IV. Experiment

A. Dataset and Preprocessing

1) Dataset: In this study, we employ the Ar-MuSA (Arabic Multimodal Sentiment Analysis) dataset, a comprehensive and open-source benchmark specifically developed to address the scarcity of resources for Arabic multimodal sentiment analysis [19]. Ar-MuSA consists of 8,700 samples spanning three modalities—text (transcripts), audio, and images—systematically extracted from 107 Egyptian Arabic YouTube videos, with precise alignment ensured via synchronized timestamps. Each sample is annotated for sentiment (positive, negative, or neutral) at both the modality-specific and overall levels by a team of native-speaking annotators, and the final labels are determined through majority voting. The corpus significantly surpasses existing Arabic multimodal datasets in terms of both scale and diversity, providing a balanced sentiment distribution (35.5% positive, 43.3% neutral, 21.2% negative), while also capturing the unique linguistic and cultural challenges inherent in Arabic (see Fig. 2), such as sarcasm present in speech and facial expressions. The data preparation pipeline involves detailed preprocessing steps, including audio segmentation, representative frame extraction, and automated transcript generation via SeamlessM4T v2. Ar-MuSA is publicly accessible through HuggingFace, facilitating research advancements in cross-modal fusion and Arabic natural language processing.

2) Preprocessing: Preprocessing is a critical step in multimodal sentiment analysis, as it ensures consistency across modalities and prepares inputs for transformer-based modeling. In this work, we adopted modality-specific preprocessing pipelines for text, audio, and visual data to optimize compatibility with state-of-the-art models in each domain.

For the text modality, transcripts were tokenized using the MARBERT tokenizer, which is designed explicitly for Arabic, incorporating both dialectal variants and Modern Standard Arabic [40]. During preprocessing, raw text was mapped to token indices, and each sequence was padded or truncated to a maximum length of 128 tokens to maintain uniformity across samples. This standardization ensures consistent input dimensions for the MARBERT model during training and inference.

The audio modality comprised segmented utterances aligned with the text and visual data. Audio signals were first converted to mono and resampled to a sampling rate of 16 kHz. Following this, we utilized HuggingFace's

Wav2Vec2FeatureExtractor for feature extraction, which included normalization, framing, and padding or truncation to a fixed duration of 5 seconds (approximately 80,000 samples). This approach ensures that each audio sequence is of comparable length, facilitating efficient batch processing and stable learning during model fine-tuning.

We extracted a single frame per utterance corresponding to the relevant audio and text segments for the visual modality. Each image was loaded in RGB format to ensure color consistency across the dataset. Preprocessing was performed using the ViTImageProcessor from the Hugging Face Transformers library. This extractor standardized all frames to the required 224 × 224-pixel size and applied intensity normalization, preparing the images for input into the Vision Transformer model ("google/vit-base-patch16-224")[41].

B. Training Details

In this work, our primary focus is maximizing the expressive power of tri-modal fusion for Arabic multimodal sentiment analysis. While we report baseline results for unimodal and bimodal scenarios for comparative rigor, most of our methodological attention is dedicated to the optimal training and integration of the textual, acoustic, and visual streams in the tri-modal setting. To this end, our training protocol is structured around two core stages: 1) independent modality-specific fine-tuning and 2) downstream fusion and classifier training. Below, we provide a detailed account of both, explicitly referencing all training protocols and model configurations utilized.

The specific hyperparameters for each modality were determined empirically and are summarized in Table I, Table II, and Table III. These tables offer a transparent and comprehensive reference with all key settings, including optimizer type (Adam or AdamW), learning rate, batch size, number of epochs, gradient accumulation steps, warmup steps or ratios, evaluation frequency, and initialization seeds.

TABLE I. TRAINING SETUP FOR MARBERT MODEL

Parameter	Value
Adam Epsilon	1e-8
Learning Rate	1e-5
FP16	Enabled
Train Batch Size	16
Eval Batch Size	16
Gradient Accumulation Steps	2
Number of Epochs	5
Warmup Steps	0.2
Evaluation Strategy	Per Epoch
Random Seed	42
Logging Steps	200

After feature extraction from the fine-tuned MarBERT (text), HuBERT (audio), and ViT (visual) models, the respective feature vectors were concatenated along the feature dimension to form unified representations for each data instance. These tri-modal concatenated feature vectors were then used as input for a suite of machine learning classifiers: Support Vector Machine (SVM), Random Forest (RF), k-nearest Neighbors (KNN), and Naive Bayes (NB), and each

TABLE II. TRAINING SETUP FOR HUBERT MODEL

Parameter	Value
Adam Epsilon	1e-8
Learning Rate	2e-5
FP16	Enabled
Train Batch Size	8
Eval Batch Size	8
Gradient Accumulation Steps	2
Number of Epochs	3
Warmup Steps	0.2
Evaluation Strategy	Per Epoch
Random Seed	42
Logging Steps	200

TABLE III. TRAINING SETUP FOR VIT 16X16 MODEL

Parameter	Value
Train Batch Size	32
Eval Batch Size	32
Learning Rate	0.0001
Input_shape	(224, 224, 3)
Optimizer	Adam
Loss_function	Categorical Cross entropy
Number of Epochs	3
Metrics	Accuracy

model was implemented within a `scikit-learn` pipeline [42]. Table IV summarizes the primary hyperparameters used for each classifier, as optimized or specified for our tri-modal (Text+Audio+Image) classification experiments:

For all experiments, data were partitioned using a stratified split, with 80% of the data allocated for training, 10% for validation, and 10% for testing, while maintaining label distribution across all subsets (`random_state=42`) to ensure reproducibility. Unless specified otherwise, all classifier hyperparameters were tuned and evaluated primarily on the tri-modal feature fusion setting, which constitutes the central focus of our approach.

C. Evaluation Metrics

A rigorous evaluation of our multimodal sentiment classification model requires a multifaceted approach, particularly given the frequent class imbalance typical of sentiment analysis datasets. To this end, we adopted a comprehensive suite of metrics to ensure fairness and depth in assessing our models. The primary metrics considered include overall accuracy, weighted precision (WP), weighted recall (WR), and weighted F1-score (WF-1). Below, we present explicit definitions and the mathematical formulations employed for each metric.

1) Weighted Precision (WP): Precision quantifies the proportion of optimistic predictions that are indeed correct. In scenarios where classes are imbalanced, simply averaging across classes can be misleading. Weighted precision mitigates this by giving higher importance to classes with greater representation in the data. It is computed as:

$$\text{WP} = \frac{\sum_{i=1}^{n} [\text{Precision}_i \times \text{Support}i]}{\sum i = 1^n \text{Support}_i}$$

where, Precision_i is the precision for class i, Support_i is

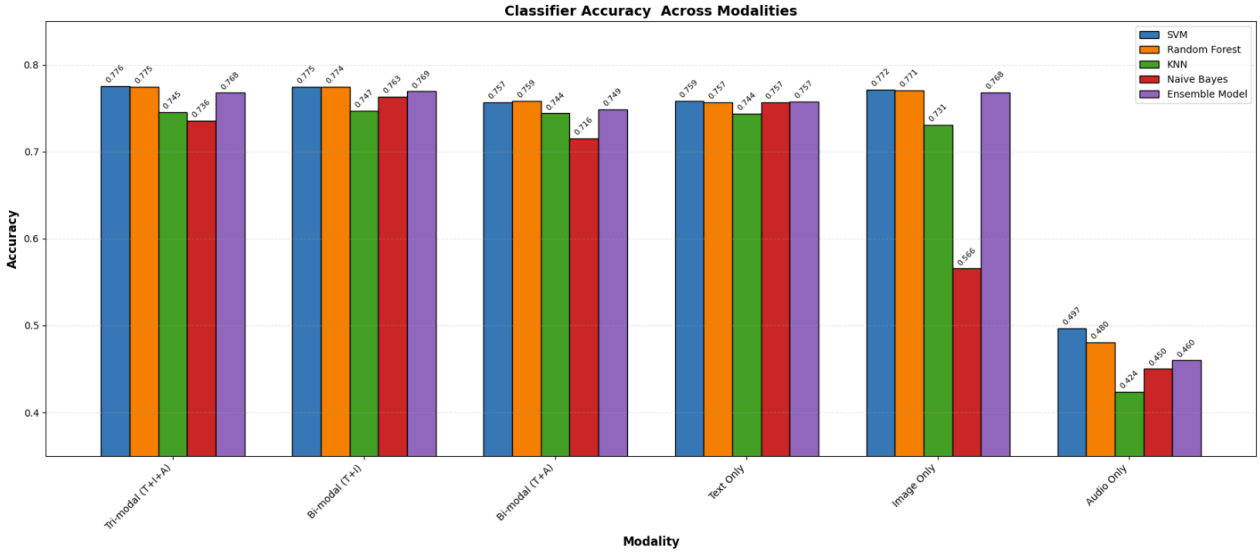


Fig. 3. Classification accuracy (%) of all evaluated classifiers across modalities and fusion types.

TABLE IV. PRIMARY HYPERPARAMETERS USED FOR EACH CLASSIFIER IN THE TRI-MODAL EXPERIMENT

Classifier	Hyperparameters
SVM (RBF)	kernel=rbf C=1.0 gamma=scale probability=True random_state=42
Random Forest	n_estimators=500 max_depth=30 max_features=sqrt class_weight=balanced min_samples_split=5 n_jobs=-1 random_state=42
KNN	n_neighbors=5 weights=distance algorithm=auto n_jobs=-1
Naive Bayes	GaussianNB après normalisation)
Ensemble (Voting)	Voting=soft estimateurs={SVM, RF, KNN, NB}

the number of true samples in class i, and n is the total number of classes.

2) Weighted Recall (WR): Recall measures the ability of the classifier to identify all relevant instances for each class correctly. Just like precision, it is important to adjust for class support. Weighted recall is defined as:

$$\text{WR} = \frac{\sum_{i=1}^{n} [\text{Recall}_i \times \text{Support}_i]}{\sum_{i=1}^{n} \text{Support}_i}$$

with Recall_i representing the recall for class i. This metric ensures a more representative assessment, especially when certain classes dominate the dataset.

3) Weighted F1-Score (WF-1): The F1-score is the harmonic mean of precision and recall and provides a single measure of a classifier's accuracy about the positive class. The weighted F1-score, which addresses the variance in class sizes, is calculated as:

$$\text{WF-1} = \frac{\sum_{i=1}^{n} [\text{F1-Score}_i \times \text{Support}_i]}{\sum_{i=1}^{n} \text{Support}_i}$$

where, the F1-score for each class, i, is given by:

$$\text{F1-Score}_i = \frac{2 \times \text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i}$$

4) Accuracy: Often considered the most intuitive metric, accuracy is the ratio between the number of correct predictions and the total number of predictions made:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

To provide additional transparency, each experimental result is accompanied by a full classification report detailing precision, recall, F1-score, and support for each sentiment class (positive, negative, and neutral). We further visualize performance using confusion matrices, which offer insight into which classes the model will most likely confuse.

All metrics are systematically computed using the `scikit-learn` Python package [42], ensuring reproducibility and consistency across experiments. Using weighted metrics throughout is deliberate, as it more accurately reflects classifier performance in the presence of uneven class distributions—an everyday reality in real-world sentiment datasets.

This robust evaluation protocol allows for a nuanced comparison of different classifiers and fusion methods, underpinning the credibility of our findings in multimodal sentiment analysis.

TABLE V. COMPARISON OF OUR CONFIGURATIONS WITH STATE-OF-THE-ART SYSTEMS AND AR-MUSA BENCHMARK BASELINES

System / Study	Modality / Fusion	Model	WF-1	Accuracy
Our (Tri-modal)	Text + Audio + Image	SVM	**0.776**	**0.776**
Khaled et al. [28]	Late Fusion	LLaMA 3.1–8B	0.67	0.71
Khaled et al. [28]	Late Fusion	SILMA AI	0.65	0.68
Ar-MuSA [19]	Late Fusion	Voting Ensemble	0.60	0.63

V. RESULTS

This section presents an in-depth evaluation of our multimodal sentiment classification methodology on the Ar-MuSA benchmark, particularly emphasizing the tri-modal (text, audio, and image) fusion setting. For broader benchmarking, unimodal and bi-modal results are included. However, the analysis prioritizes tri-modal fusion—demonstrating its integrative strengths and practical relevance for robust Arabic sentiment recognition, as shown in Fig. 3.

A. Main Focus: Tri-modal Fusion Results

Integrating textual, audio, and visual streams is hypothesized to leverage complementary evidence and improve sentiment classification performance. In our tri-modal fusion, we concatenate deep feature vectors from MarBERT (text), HuBERT (audio), and ViT (image), yielding a high-dimensional representation for each instance. Table VI documents performance across classifiers.

TABLE VI. TRI-MODAL SENTIMENT CLASSIFICATION RESULTS (TEXT–IMAGE–AUDIO FUSION)

Classifier	Accuracy	WR	WF-1	WP
SVM	**0.7759**	**0.7759**	**0.7756**	**0.7783**
Random Forest	0.7747	0.7747	0.7747	0.7777
KNN	0.7454	0.7454	0.7449	0.7466
Naive Bayes	0.7356	0.7356	0.7362	0.7386
Ensemble Model	0.7678	0.7678	0.7678	0.7691

Tri-modal fusion consistently yields the highest overall performance, confirming the value of aggregating diverse cues. The SVM outperforms all other models (Accuracy: 0.7759, WF-1: 0.7756), suggesting that synergistic evidence from all three modalities enables stronger discrimination of sentiment—especially capturing aspects that cannot be derived from text or image alone. Random Forest is a close second, while KNN and Naive Bayes lag, possibly due to their limitations with complex, high-dimensional data. The ensemble, while robust, does not exceed the single SVM, indicating that hard or soft voting does not add to the already dominant SVM classification when features are so robust. Importantly, tri-modal fusion provides measurable, though sometimes modest, gains compared to unimodal and bimodal systems, underscoring its unique value.

B. Comparison with the State-of-the-Art and Baselines

To quantify the effectiveness of our approach, we compare our best modal configurations against the official Ar-MuSA benchmark baselines [19], as well as the recent late-fusion large-model results reported by Khaled et al. [28]. Following common practice in sentiment classification on potentially imbalanced datasets, we report weighted F1-score (WF-1) and accuracy as primary metrics. Table V provides a consolidated view of the competing systems and highlights the relative contribution of each modality and fusion setting in our pipeline. For an intuitive overview of the relative gains across systems.

As observed in Table V, our tri-modal configuration (Text+Audio+Image) with early feature concatenation and an SVM classifier achieves a WF-1 of 0.776 and an accuracy of 0.776, establishing the overall results among the compared systems. In particular, this represents a clear improvement over the best tri-modal baseline reported in Ar-MuSA (late-fusion voting ensemble; WF-1 = 0.60, accuracy = 0.63) [19], indicating that our fusion and feature design enables more effective exploitation of complementary cross-modal sentiment cues. Compared with recent late-fusion, large-model approaches [28], our best configurations remain consistently both metrics. This outcome highlights that, for Ar-MuSA-style sentiment analysis, carefully designed early fusion coupled with effective feature extraction can rival (and in this case outperform) heavier late-fusion strategies, while remaining comparatively simple and potentially more efficient to train and deploy.

C. Uni and Bi-modal Results: Added for Comparative Benchmarking

To contextualize the strength of tri-modal fusion, we evaluated unimodal and bimodal setups—especially for comparison with the Ar-MuSA baseline. These results reveal how much each modality contributes individually and in pairs, but serve primarily as references for the added value of tri-modal integration.

1) Unimodal results: Table VII compiles unimodal results across text, audio, and image—for all classifiers.

TABLE VII. UNIMODAL SENTIMENT CLASSIFICATION RESULTS ACROSS MODALITIES.

Modality	Classifier	Accuracy	WR	WF-1	WP
Text	SVM	**0.7586**	**0.7586**	**0.7580**	**0.7595**
	Random Forest	0.7569	0.7569	0.7563	0.7584
	KNN	0.7437	0.7437	0.7433	0.7436
	Naive Bayes	0.7569	0.7569	0.7574	**0.7595**
	Ensemble Model	0.7575	0.7575	0.7577	0.7587
Audio	SVM	**0.4971**	**0.4971**	**0.4602**	**0.4887**
	Random Forest	0.4805	0.4805	0.4549	0.4588
	KNN	0.4236	0.4236	0.4180	0.4151
	Naive Bayes	0.4500	0.4500	0.4537	0.4772
	Ensemble Model	0.4598	0.4598	0.4632	0.4775
Image	SVM	**0.7718**	**0.7718**	**0.7641**	**0.7687**
	Random Forest	0.7708	0.7708	0.7631	0.767 7
	KNN	0.7310	0.7310	0.7263	0.7259
	Naive Bayes	0.5655	0.5655	0.5746	0.6871
	Ensemble Model	0.7684	0.7684	0.7700	0.7727

Text and image provide height unimodal baselines, but each is limited: image achieves the highest scores (SVM WF-1: 0.7641) yet cannot capture language-specific sentiment; text is robust (SVM WF-1: 0.758) but may miss paralinguistic or non-verbal signals. Audio is the weakest (SVM WF-1: 0.460), reflecting the challenge of sentiment inference from prosody alone and possible imbalance in audio cues, a point corroborated by prior results on Arabic datasets. Ultimately, these unimodal numbers highlight why fusion—especially tri-modal fusion—is necessary for holistic sentiment recognition.

2) Bi-modal results: Bi-modal fusion results (see Table VIII) are listed next. Here, we focus on the text-audio and text-image pairs, following the Ar-MuSA evaluation.

TABLE VIII. BI-MODAL SENTIMENT CLASSIFICATION RESULTS (TEXT-AUDIO AND TEXT-IMAGE FUSION).

Fusion	Classifier	Accuracy	WR	WF-1	WP
T+A	SVM	0.7569	0.7569	0.7562	0.7583
	Random Forest	**0.7586**	**0.7586**	**0.7581**	**0.7602**
	KNN	0.7443	0.7443	0.7439	0.7441
	Naive Bayes	0.7155	0.7155	0.7158	0.7165
	Ensemble Model	0.7489	0.7489	0.7487	0.7493
T+I	**SVM**	**0.7750**	**0.7750**	**0.7746**	**0.7776**
	Random Forest	0.7745	0.7745	0.7743	0.7774
	KNN	0.7471	0.7471	0.7467	0.7482
	Naive Bayes	0.7632	0.7632	0.7637	0.7664
	Ensemble Model	0.7695	0.7695	0.7697	0.7716

Notes: T = Text modality; A = Audio modality; I = Image modality.

Bi-modal fusion, particularly with the text-image pair, yields performance only slightly above the best unimodal results (WF-1 up to 0.7746 for SVM). In some cases, improvements are marginal or classifier-dependent. The text-audio pair offers minimal additional information over text alone, which aligns with the weaker standalone audio performance. These outcomes reinforce that while bimodal fusion can harness some complementary signals, the dominant contribution still arises from the text stream, and these combinations primarily serve as strong baselines and ablation checks for the tri-modal configuration.

It is evident from the above results that while unimodal and bi-modal settings offer competitive baselines and are crucial for benchmarking against prior work, such as the Ar-MuSA dataset, their practical ceiling is reached quickly. The tri-modal approach remains unmatched in providing robust and nuanced sentiment analysis by integrating all available cues. These findings support the growing consensus in the literature that multimodal fusion—when effectively implemented—can yield modest but consistent advances over single or dual-modality systems, especially in linguistically and emotionally rich domains such as Arabic social media.

In the subsequent section, we contextualize these findings by directly comparing our results to the Ar-MuSA benchmarks, highlighting the gains achieved with our tri-modal fusion strategy.

VI. DISCUSSION

The experimental results clearly demonstrate the advantages of multimodal learning for Arabic sentiment analysis on social media data. Across all evaluated settings, tri-modal fusion (Text+Audio+Image) consistently achieved the best performance, with early feature concatenation combined with an SVM classifier yielding the highest scores (WF-1 = 0.776, Accuracy = 0.776). This confirms that sentiment in user-generated Arabic content is not solely conveyed through text, but rather emerges from the complementary interaction of linguistic, visual, and acoustic cues.

Unimodal experiments highlight the relative contribution of each modality. Text and image independently provide strong sentiment signals, whereas audio alone remains comparatively weak due to noise, speaker variability, and the subtle nature of prosodic cues. Nevertheless, when integrated with text and image, audio contributes complementary information that improves robustness and reduces ambiguity in difficult cases.

Bi-modal fusion experiments reveal that Text+Image represents the most effective dual-modality configuration, achieving performance close to the full tri-modal system. In contrast, Text+Audio fusion offers only marginal gains over text-only models. These findings suggest that visual context plays a particularly important role in disambiguating sentiment in Arabic social media, while audio acts as a secondary but supportive modality.

Compared with prior work and official Ar-MuSA baselines, the proposed approach consistently outperforms existing unimodal, bi-modal, and tri-modal systems, including recent late-fusion and large-model approaches. Notably, the performance of a relatively simple early-fusion SVM pipeline indicates that carefully designed feature-level integration of pretrained encoders can be more effective than heavier ensemble or late-fusion architectures.

Overall, the results advocate for comprehensive tri-modal fusion as the most reliable strategy for sentiment classification in Arabic social media, while also highlighting Text+Image fusion as a practical near-optimal alternative when audio is unavailable. These findings reinforce the importance of multimodal representations for capturing the richness and variability of sentiment expression in low-resource and highly informal linguistic settings.

VII. CONCLUSION

This study introduced a comprehensive framework for Arabic multimodal sentiment analysis that integrates textual, visual, and acoustic information through early feature-level fusion. By leveraging strong pretrained encoders—MarBERT for text, ViT for images, and HuBERT for speech—and combining them with margin-based classification, the proposed approach achieves state-of-the-art performance on the Ar-MuSA benchmark. Experimental results confirm that tri-modal fusion consistently outperforms unimodal and bi-modal configurations, underscoring the importance of jointly modeling linguistic content, visual context, and paralinguistic cues for sentiment understanding in user-generated Arabic media.

The study further demonstrates that while text and image modalities provide strong standalone signals, audio plays a complementary role that enhances robustness when integrated with other modalities, despite its relatively weaker individual performance. Compared with existing baselines and recent

late-fusion approaches, the proposed early-fusion strategy offers a more effective and computationally efficient alternative.

Building on these findings, future work will focus on extending the framework through improved speech representation learning, dialect-aware acoustic modeling, and more interaction-aware fusion strategies. Such extensions aim to further strengthen multimodal representations and advance sentiment analysis in low-resource, highly variable Arabic social media environments.

REFERENCES

[1] S. Gadomski, "North african arabic literary expressions of lgbtqia+ identity in the digital space," in *Digital Expressions of Gender in Africa*. Routledge, 2025, pp. 50–70.

[2] A. Habberrih and M. A. Abuzaraida, "Sentiment analysis of arabic dialects: A review study," in *International Conference on Computing and Informatics*. Springer, 2023, pp. 137–153.

[3] I. Hadjadji, L. Falek, L. Demri, and H. Teffahi, "Emotion recognition in arabic speech," in *2019 international conference on advanced electrical engineering (ICAEE)*. IEEE, 2019, pp. 1–5.

[4] N. Boudad, R. Faizi, R. O. H. Thami, and R. Chiheb, "Sentiment analysis in arabic: A review of the literature," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 2479–2490, 2018.

[5] S. Klaylat, Z. Osman, L. Hamandi, and R. Zantout, "Emotion recognition in arabic speech," *Analog Integrated Circuits and Signal Processing*, vol. 96, no. 2, pp. 337–351, 2018.

[6] R. Das and T. D. Singh, "Multimodal sentiment analysis: a survey of methods, trends, and challenges," *ACM Computing Surveys*, vol. 55, no. 13s, pp. 1–38, 2023.

[7] A. Gandhi, K. Adhvaryu, S. Poria, E. Cambria, and A. Hussain, "Multimodal sentiment analysis: A systematic review of history, datasets, multimodal fusion methods, applications, challenges and future directions," *Information Fusion*, vol. 91, pp. 424–444, 2023.

[8] T. Zhu, L. Li, J. Yang, S. Zhao, H. Liu, and J. Qian, "Multimodal sentiment analysis with image-text interaction network," *IEEE transactions on multimedia*, vol. 25, pp. 3375–3385, 2022.

[9] P. H. Winston, *Artificial intelligence*. Addison-Wesley Longman Publishing Co., Inc., 1984.

[10] J. O'Connor and I. McDermott, *NLP*. Thorsons London, UK:, 2001.

[11] S. Lai, X. Hu, H. Xu, Z. Ren, and Z. Liu, "Multimodal sentiment analysis: A survey," *Displays*, vol. 80, p. 102563, 2023.

[12] S. Alfarhood, "Soutcom: Real-time sentiment analysis of arabic text for football fan satisfaction using a bidirectional lstm," *Expert Systems*, vol. 42, no. 2, p. e13641, 2025.

[13] E. H. Nfaoui and H. Elfaik, "Evaluating arabic emotion recognition task using chatgpt models: a comparative analysis between emotional stimuli prompt, fine-tuning, and in-context learning," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 19, no. 2, pp. 1118–1141, 2024.

[14] S. E. Alqaan and A. M. Qamar, "Sentiment analysis of arabic tweets on online learning during the covid-19 pandemic: A machine learning and lstm approach," *Ingenierie des Systemes d'Information*, vol. 28, no. 6, p. 1435, 2023.

[15] S. Firdous and M. S. Iqbal, "Exploring contemporary arabic sentiment analysis: Methods, challenges, and future trends," *Pakistan Journal of Multidisciplinary Innovation*, vol. 4, no. 1, pp. 34–48, 2025.

[16] R. Zarnoufi, "Maproc at ahasis shared task: Few-shot and sentence transformer for sentiment analysis of arabic hotel reviews," *arXiv preprint arXiv:2511.15291*, 2025.

[17] A. Albladi, M. Islam, and C. Seals, "Sentiment analysis of twitter data using nlp models: a comprehensive review," *IEEE Access*, 2025.

[18] H. Yang, Y. Zhao, Y. Wu, S. Wang, T. Zheng, H. Zhang, Z. Ma, W. Che, S. Wang, S. Wei *et al.*, "Large language models meet text-centric multimodal sentiment analysis: A survey," *Science China Information Sciences*, vol. 68, no. 10, pp. 1–29, 2025.

[19] S. Khaled, M. E. Ragab, A. K. Helmy, W. Medhat, and E. H. Mohamed, "Ar-musa: A multimodal benchmark dataset and evaluation framework for arabic sentiment analysis." *International Journal of Intelligent Engineering & Systems*, vol. 18, no. 4, 2025.

[20] D. A. Musleh, I. Alkhwaja, A. Alkhwaja, M. Alghamdi, H. Abahussain, F. Alfawaz, N. Min-Allah, and M. M. Abdulqader, "Arabic sentiment analysis of youtube comments: Nlp-based machine learning approaches for content evaluation," *Big Data and Cognitive Computing*, vol. 7, no. 3, p. 127, 2023.

[21] D. A. Musleh, T. A. Alkhales, R. A. Almakki, S. E. Alnajim, S. K. Almarshad, R. S. Alhasaniah, S. S. Aljameel, and A. A. Almuqhim, "Twitter arabic sentiment analysis to detect depression using machine learning," *Computers, Materials & Continua*, vol. 71, no. 2, 2022.

[22] H. Alamoudi, N. Aljojo, A. Munshi, A. Alghoson, A. Banjar, A. Tashkandi, A. Al-Tirawi, and I. Alsaleh, "Arabic sentiment analysis for student evaluation using machine learning and the arabert transformer," *Engineering, Technology & Applied Science Research*, vol. 13, no. 5, pp. 11 945–11 952, 2023.

[23] A. S. Alqarafi, A. Adeel, M. Gogate, K. Dashitpour, A. Hussain, and T. Durrani, "Toward's arabic multi-modal sentiment analysis," in *International Conference in Communications, Signal Processing, and Systems*. Springer, 2017, pp. 2378–2386.

[24] H. Najadat and F. Abushaqra, "Multimodal sentiment analysis of arabic videos," *Journal of Image and Graphics*, vol. 6, no. 1, pp. 39–43, 2018.

[25] S. Al-Azani and E.-S. M. El-Alfy, "Enhanced video analytics for sentiment analysis based on fusing textual, auditory and visual information," *IEEE Access*, vol. 8, pp. 136 843–136 857, 2020.

[26] A. Haouhat, S. Bellaouar, A. Nehar, and H. Cherroun, "Towards arabic multimodal dataset for sentiment analysis," in *2023 Fourth International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*. IEEE, 2023, pp. 126–133.

[27] S. Khaled, M. E. Ragab, A. K. Helmy, W. Medhat, and E. H. Mohamed, "Ar-musa: A multimodal benchmark dataset and evaluation framework for arabic sentiment analysis," *International Journal of Intelligent Engineering & Systems*, vol. 18, no. 4, 2025.

[28] S. Khaled, W. Medhat, and E. H. Mohamed, "Unitextfusion: A low-resource framework for arabic multimodal sentiment analysis using early fusion and lora-tuned language models," *Ain Shams Engineering Journal*, vol. 16, no. 11, p. 103682, 2025.

[29] T. Baltrušaitis, C. Ahuja, and L.-P. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 2, pp. 423–443, 2018.

[30] P. P. Liang, A. Zadeh, and L.-P. Morency, "Foundations & trends in multimodal machine learning: Principles, challenges, and open questions," *ACM Computing Surveys*, vol. 56, no. 10, pp. 1–42, 2024.

[31] A. B. Cheikhi and E. H. Nfaoui, "Multimodal arabic sarcasm detection using cnn and bilstm," in *2025 International Conference on Intelligent Systems: Theories and Applications (SITA)*, 2025, pp. 1–5.

[32] J. Fields, K. Chovanec, and P. Madiraju, "A survey of text classification with transformers: How wide? how large? how long? how accurate? how expensive? how safe?" *IEEE Access*, vol. 12, pp. 6518–6531, 2024.

[33] Y. Wang, Y. Deng, Y. Zheng, P. Chattopadhyay, and L. Wang, "Vision transformers for image classification: A comparative survey," *Technologies*, vol. 13, no. 1, p. 32, 2025.

[34] A. Ben Cheikhi and E. H. Nfaoui, "Mathematics problem classification based on pretrained language models," in *2025 11th International Conference on Optimization and Applications (ICOA)*, 2025, pp. 1–6.

[35] H. Lian, C. Lu, S. Li, Y. Zhao, C. Tang, and Y. Zong, "A survey of deep learning-based multimodal emotion recognition: Speech, text, and face," *Entropy*, vol. 25, no. 10, p. 1440, 2023.

[36] A. A. Gladys and V. Vetriselvi, "Survey on multimodal approaches to emotion recognition," *Neurocomputing*, vol. 556, p. 126693, 2023.

[37] H. Elfaik *et al.*, "Leveraging feature-level fusion representations and attentional bidirectional rnn-cnn deep models for arabic affect analysis on twitter," *Journal of King Saud University-Computer and Information Sciences*, vol. 35, no. 1, pp. 462–482, 2023.

[38] B. Azahouani, H. Elfaik, S. El Garouani *et al.*, "Multimodal sarcasm detection method using rnn and cnn," in *2024 Sixth International Conference on Intelligent Computing in Data Sciences (ICDS)*. IEEE, 2024, pp. 1–6.

[39] O. Alharbi, "A deep learning approach combining cnn and bi-lstm with svm classifier for arabic sentiment analysis," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, 2021.

[40] M. Abdul-Mageed, A. Elmadany, and E. M. B. Nagoudi, "Arbert & marbert: Deep bidirectional transformers for arabic," 2021. [Online]. Available: https://arxiv.org/abs/2101.01785

[41] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021. [Online]. Available: https://arxiv.org/abs/2010.11929

[42] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

HQ-RTVF: High-Quality Real-Time Virtual Try-On Fitting for Diverse Clothing and Body Morphologies

Ilham KACHBAL, Khadija Arhid, Said El Abdellaoui
LAPSSII, Higher School of Technology, Cadi Ayyad University, B.P. 89, Safi, Morocco

***Abstract*—The ability to virtually try on clothing items has become an increasingly important feature for e-commerce and online shopping experiences. Real-time virtual try-on remains challenging because existing methods force a trade-off between speed and quality GAN-based approaches achieve high visual fidelity but at low frame rates, while faster methods sacrifice realism. HQ-RTVF is a diffusion-based framework that resolves this trade-off through three architectural innovations: running the diffusion U-Net entirely in the VAE's compressed latent space (64×64×4 instead of 512×512×3), limiting denoising to 20 steps with FP16 mixed-precision computation, and parallelizing pose estimation and garment encoding to eliminate sequential bottlenecks. The system uses DensePose and DeepLabv3+ for body pose and segmentation, a CLIP-based garment encoder for fine-grained fabric representation, and an attention-guided fusion decoder that maintains temporal coherence across video frames — distinguishing it from static image methods like VITON-HD and HR-VITON. An adaptive masking mechanism handles diverse garment types from cropped tops to full-length dresses. Evaluated on VITON-HD and DressCode datasets, HQ-RTVF achieves SSIM of 0.950 and LPIPS of 0.067, while operating in real-time with only 4.2 GB GPU memory.**

***Keywords*—Virtual try-on; diffusion models; real-time processing; deep learning; garment synthesis; pose estimation**

I. Introduction

The fashion industry increasingly relies on e-commerce, yet online shopping faces a critical challenge: the inability to try on clothes before purchasing, leading to sizing discrepancies and customer dissatisfaction [1]. Virtual try-on technology powered by generative AI [2] addresses this by enabling shoppers to visualize garments on their bodies in real-time using smartphones, tablets, or computers, combining online convenience with interactive in-store experiences.

Early virtual try-on networks [3], [4], [5], [6] were limited to frontal poses, struggling with diverse body positions and shapes [7]. This restricted practical applicability, as models trained on single-pose datasets failed to handle real-world pose variations, resulting in distortions and loss of garment details.

Our two-stream approach overcomes these limitations without requiring 3D inputs. The first stream employs a novel Garment Encoder CLIP that processes 2D clothing images to extract semantic features (color, texture, patterns, wrinkles). The second stream uses DensePose and DeepLabv3+ [8] to extract pose landmarks and segmentation masks. Three sequential modules—diffusion noise encoder, garment encoder CLIP [9], and fusion decoder—process these features through stable diffusion U-Net architecture and VAE embeddings to generate photorealistic try-on images. Pose masking during inference ensures generation focuses only on the person region, eliminating background interference. Comprehensive evaluations demonstrate that our method produces high-fidelity multi-pose try-ons at interactive rates, even for complex motion sequences.

This study is structured as follows: Section II discusses existing virtual try-on techniques and their limitations for real-time applications. Section III then describes each step of our proposed real-time virtual try-on method, including the garment encoder CLIP, diffusion noise encoder, and fusion decoder mechanisms. Furthermore, Section IV presents a discussion of our results along with how our method compares to baseline approaches. Finally, Section V provides a summary of the key contributions of this work in efficiently enabling photorealistic virtual try-ons across dynamic poses through a novel two-stream framework based on dense correspondences between frames.

II. Related Work

Virtual try-on systems are categorized into two main approaches (see Fig. 1): GAN-based and diffusion-based techniques.

A. GAN-Based Virtual Try-On

GAN-based systems [10] produce photorealistic outputs with high interactivity, allowing adjustment of size, color, and pose for both static images and dynamic videos. However, they require large paired datasets and complex architectures. Early networks were limited to single poses. Researchers advanced the field through deep learning background matting [11], [12], enabling seamless compositing onto diverse contexts and poses for apparel e-commerce [13], [14], [15]. These methods fall into three architectural categories: warping-based methods (CP-VTON [16], HR-VITON [5]) explicitly deform garments using thin-plate spline transformations, preserving texture but struggling with large pose variations; parser-free methods (PBAFN [17]) eliminate dependency on human parsing maps, improving generalization but sacrificing fine-grained detail; and flow-field-based methods (GP-VTON [18]) estimate dense correspondence fields for flexible deformation at higher computational cost. Despite these advances, all GAN-based approaches require paired training data and struggle to generalize to unseen garments, motivating the shift toward diffusion-based solutions.

High-resolution VTON methods addressed detail limitations. HR-VITON [3] uses coarse-to-fine generation with deformable skip connections and multi-scale discriminators. VITON-HD [4] adds surface normal estimation and boundary-aware discrimination for 1280×2560 outputs. Stable-VITON [19] stabilizes training through self-supervised boundary encoding, producing 1280×1920 images with sharper edges. GP-

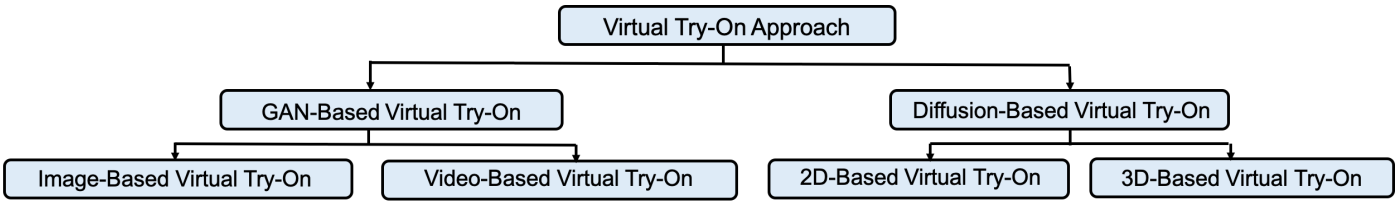


Fig. 1. Virtual try-on approaches: Two main approaches explored for photorealistic virtual try-on GAN-based methods and newly proposed diffusion models.

VTON [6] models garment properties via semantic parsing for 1280×960 results with natural wrinkles.

Video try-on advances include Clothformer [20], using transformers for temporal coherence; MIRROR [21], enabling real-time mobile inference through multi-scale iterative refinement; and Tunnel Try-on [9], generating pose transitions via spatial-temporal tunnels. Challenges remain in temporal coherence, pose generalization, and true interactivity. Existing methods often require expensive 3D modeling or produce inconsistent textures across viewing angles, highlighting the need for efficient 2D solutions.

B. Diffusion-Based Virtual Try-On

Diffusion models [22], [23], [24] leverage text conditioning for intuitive customization, producing publication-quality results comparable to photographs. ViViD [25] pioneered video try-on using per-frame conditional generation. AnyFit [26] introduced semi-parametric latent spaces for independent attribute editing through disentangled factors. BetterFit [27] uses deformation field learning for pose generalization across diverse clothing. Stable-VITON [5] regresses semantic correspondences between people and garments for 3D customization. OOTDiffusion [28] presents Outfitting Fusion-based Latent Diffusion (OFLD), disentangling person identity and clothing attributes without paired training data. Outfit fusion in latent space enables controllable try-on synthesis with interactive editing of garment type, color, and size, outperforming baselines on benchmark datasets. While existing diffusion-based methods such as OOTDiffusion [9] and ViViD [25] demonstrate promising results, HQ-RTVF differentiates itself through three specific technical contributions: 1) a CLIP-based garment encoder with dual refiners that captures fine-grained fabric details beyond global semantic features, unlike standard CLIP conditioning used in prior works; 2) an adaptive masking mechanism that automatically classifies garment configurations (long-type, short-type, non-interfering) directly from pose estimation without manual annotation; and 3) a parallel two-stream architecture that processes body pose and garment features simultaneously rather than sequentially, enabling real-time inference at 60 FPS while maintaining photorealistic quality a combination no existing diffusion-based try-on method achieves.

III. Our Approach HQ-RTVF

To enable real-time virtual try-on capabilities, an efficient processing pipeline is required that can seamlessly integrate multiple deep learning models for computer vision tasks such as semantic segmentation, pose estimation, image generation, and reconstruction. In the described approach (Fig. 2), the first stage involves capturing a continuous stream of input frames in real time, which serves as the basis for the entire try-on process. These frames could be sourced from a video, camera feed, or set of pre-recorded images depicting a user's physical appearance and movements over time. Once the frames are acquired, the next step is to perform semantic segmentation to separate the person's body from the background scenery in each frame using a DeepLabV3+ [29] model pretrained for this task. DeepLabV3+ is an optimal choice here as the model provides fast and accurate pixel-level predictions due to the powerful combination of convolutional operations, atrous separable convolutions, and encoder-decoder architecture design. The model parses each new frame and outputs a semantic segmentation map denoting the pixels belonging to various classes such as skin, hair, and clothes. Concurrently, the input frames are also fed to a DensePose encoder soft-segmentation model which has been specialized for human pose estimation. DensePose's one-stage detection framework makes the model well-suited for real-time processing. The system scans each frame and returns the predicted 2D body joint locations and skeletal keypoints corresponding to areas such as shoulders, elbows, wrists, hips, and knees. This pose information serves as a starting point for virtually draping garments onto the target body shape.

A. Diffusion Noising Encoder

The diffusion noising encoder takes the output of three encoding models as input: the DensePose encoder, DeepLab encoder, and the original image itself. The pose keypoints, segmentation maps, and garment cloth maps are encoded into an initial latent code z_0. This code is perturbed with timestep-dependent Gaussian noise $\epsilon \sim \mathcal{N}(0,1)$, producing the noisy latent z_t at timestep t. z_t, t, and learned embeddings τ_θ of the inputs/garments are passed to a U-Net noise autoencoder. The role of this autoencoder is to learn a mapping from its inputs to the original noise ϵ that was added to the latent code. This process is optimized by minimizing the latent diffusion loss [9]:

$$\mathcal{L}_{LDM} = \mathbb{E}_{z_t,t,\epsilon\sim\mathcal{N}(0,1)} \left\| \epsilon - \epsilon_\theta(z_t, t, \tau_\theta(\text{inputs})) \right\|_2^2 \quad (1)$$

where, the loss measures the error between the actual noise ϵ and predicted noise ϵ_θ. Minimizing this loss trains the autoencoder to iteratively denoise z_t through noise prediction, fusing encoded pose, garment, and soft-tissue details over multiple timesteps. The output is encoded and decoded to recursively refine the underlying 2D structure, similar to the diffusion VAE. These encoded outputs along with the raw pixels are fed into the initial diffusion noise encoder. This

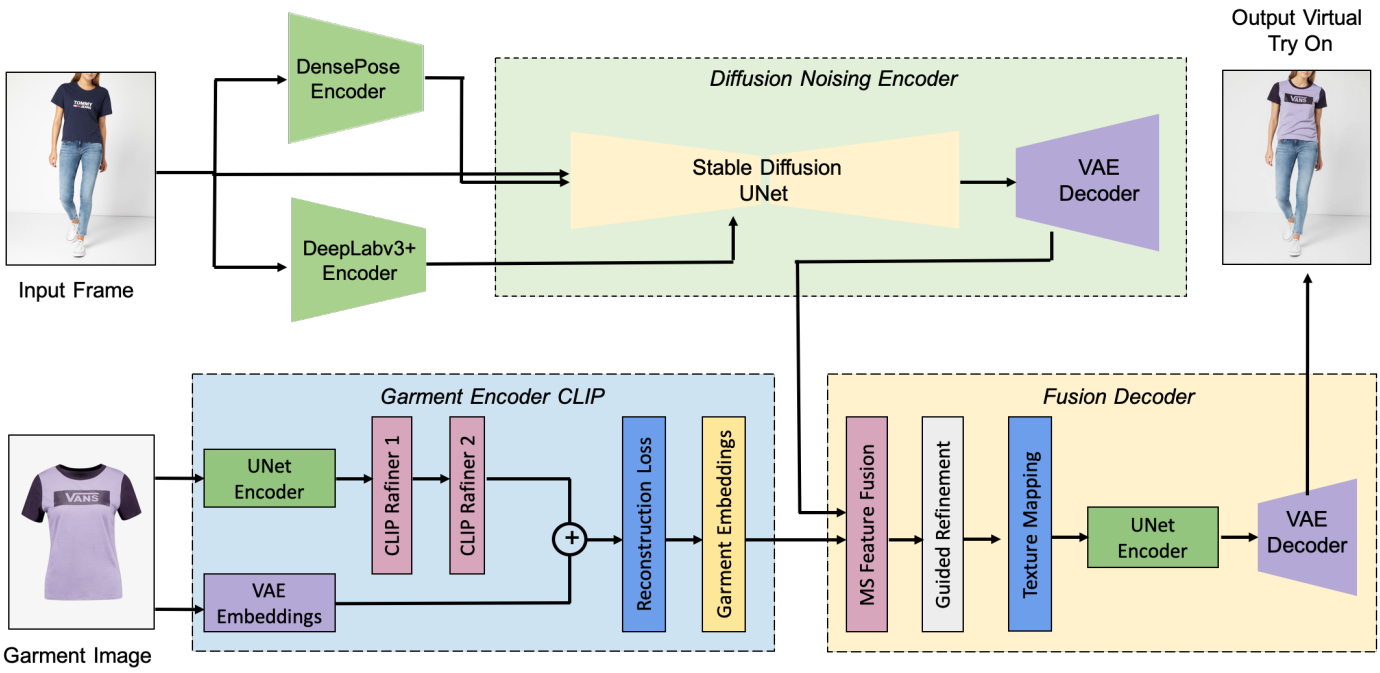


Fig. 2. Architecture overview. The key modules of the proposed method for our HQ-RTVF, including a diffusion noising encoder, CLIP-based garment encoder and fusion decoder.

encoder acts to disentangle and compress the visual features extracted by the prior models into a concise latent representation. The encoder utilizes a Variational Autoencoder (VAE) to transform the high-dimensional encoded inputs into a lower-dimensional latent space through an encoding procedure. The VAE optimization objective balances reconstruction fidelity with latent space regularization:

$$\mathcal{L}_{VAE} = \mathbb{E}_{q_\phi(z_0|x)}\left[\log p_\psi(x|z_0)\right] - \beta \cdot \mathrm{KL}(q_\phi(z_0|x) \,\|\, \mathcal{N}(0, \mathbf{I})) \tag{2}$$

where, $q_\phi(z_0|x)$ is the encoder distribution, $p_\psi(x|z_0)$ is the decoder distribution, and β controls the regularization strength. The combination of VAE autoencoders with diffusion models is essential to achieve simultaneous real-time performance and photorealistic quality. Using the VAE alone provides efficient compression, but lacks the iterative refinement capability required for high-fidelity garment synthesis, producing blurry outputs. The VAE applies a series of nonlinear transformations to map the pose, segmentation, and image encodings to a compact latent code. This abstracted encoding summarizes the key information about the scene in a latent space that is easier to manipulate. The latent code output by the VAE is then fed as input to a U-Net-based model implementing stable diffusion.

The U-Net follows an autoencoder architecture with skip connections between the encoder and decoder sections. The model takes the latent encoding and begins a process of iterative diffusion and denoising to generate photorealistic image reconstructions (Fig. 4). At each layer of the U-Net, the current feature maps are fused with corresponding encoder features using the skip connections. This aids in refining localized image details as resolution increases during upsampling in the decoder. Through multiple upsampling and downsampling blocks, the model learns to map the compressed latent code to a high-fidelity image space.

A crucial part of the overall pipeline is the use of an adaptive masking approach to generate customized segmentation masks tailored to different clothing configurations. A systematic process of checking pose, calculating aspect ratio, and creating masks automates the determination of clothing interference patterns. The method (Fig. 3) first uses five waist checkpoints identified through pose estimation to determine if the top article of clothing covers the bottom garment region. The process begins with input clothing images (a), and after a check pose analysis, we obtain the pose-analyzed results (b). If all five checkpoints are covered by the top item segmentation, the style is directly labeled as non-interfering. For cases where the top does not cover all checkpoints, the system proceeds to calculate the aspect ratio (R = Height/Width) of the garment region. If the aspect ratio exceeds the predetermined threshold, the garment is characterized as long-type; otherwise, the garment is labeled as short-type. Both long-type and short-type garments proceed to the create adaptive masks step, which generates segmentation masks tailored to each clothing configuration, producing the final adaptive masks (c). This automated classification system ensures accurate mask generation for different garment combinations that are optimized for the specific clothing interference patterns identified in the analysis process. The generated adaptive masks allow for optimal results across different clothing styles.

The results of the adaptive masking approach provide customized inputs to guide the U-Net-based image generation process. By integrating pose-dependent segmentation derived from clothing analysis, the synthesized images can realistically

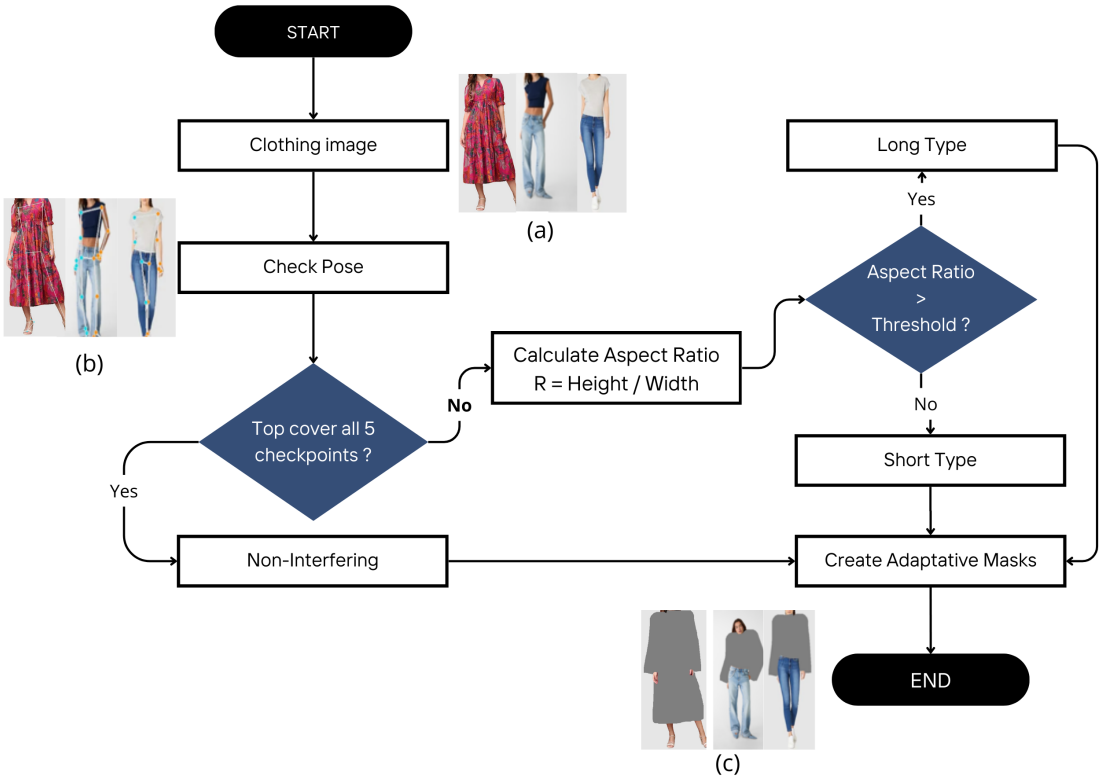


Fig. 3. Multi-step framework for adaptive clothing mask generation based on pose analysis and garment classification.

portray diverse outfits and combinations. After refining the latent encoding through iterative upsampling and downsampling with stable diffusion, the final output of the U-Net is then fed back into the initial VAE encoder for further compression and disentanglement of encoded features. In this manner, the complete pipeline forms an encode-decode loop that learns an implicit mapping from semantic inputs to photorealistic outputs incorporating adaptive segmentation cues.

The diffusion noise encoder leverages multiple encodings related to pose, segmentation, and raw pixels to distill pertinent visual information into a disentangled latent space. By coupling variational autoencoders with U-Net-based stable diffusion, the system can map between high-level inputs and complex pixel-level outputs. A defining capability is the inclusion of an automatic adaptive masking technique that derives clothing interference conditions directly from visual cues such as boundary proximity and garment dimensions.

B. Garment Encoder CLIP

In parallel to the above computer vision tasks, pretrained garment encoder models are queried to retrieve latent codes representing various clothing items from a stored catalog. These encoders have been optimized to embed visual attributes such as fabric, design, cut, and fit into compact embedding vectors. The relevant codes can then be looked up and submitted to a cloth simulation backend for physical attributes. Most virtual try-on tasks use text to describe the regions and contents that are to be edited. However, text can only provide high-level semantic information, making it difficult to capture the detailed features of the clothing. While using CLIP as an image encoder is better than text in this regard, as it encodes global visual features, CLIP is still not trained to model the fine-grained local details needed for realistic reconstruction. To address this limitation, we designed Garment Encoder CLIP, which takes a segmented clothing region as input. The architecture contains an encoder path consisting of five convolutional layers to capture multi-scale features from the input. These features are then passed through two CLIP-based refiners to further refine the spatial semantics and textures. The refined embeddings are decoded via five transposed convolutional layers to reconstruct the original input size. During training, the encoder-decoder is optimized using an L1 reconstruction loss [30] combined with a perceptual loss to preserve both pixel-level accuracy and high-level semantic features:

$$\mathcal{L}_{garment} = \lambda_1 \|G - \hat{G}\|_1 + \lambda_2 \|\phi(G) - \phi(\hat{G})\|_2 \quad (3)$$

where, G is the input garment image, $\hat{G}$ is the reconstructed output, $\phi(\cdot)$ represents features extracted from a pretrained CLIP encoder, and λ_1, λ_2 are weighting coefficients balancing reconstruction fidelity and perceptual quality.

At inference, the encoded latent code provides a compact embedding of the garment's fine-grained visual attributes. This simulation backend takes cloth parameters provided by the embeddings and simulates how the garments would realistically deform, drape, and move over a 2D representation of the body surface according to the laws of physics. The backend outputs optimized cloth maps describing the garment patterns positioned across the body under gravity and forces.

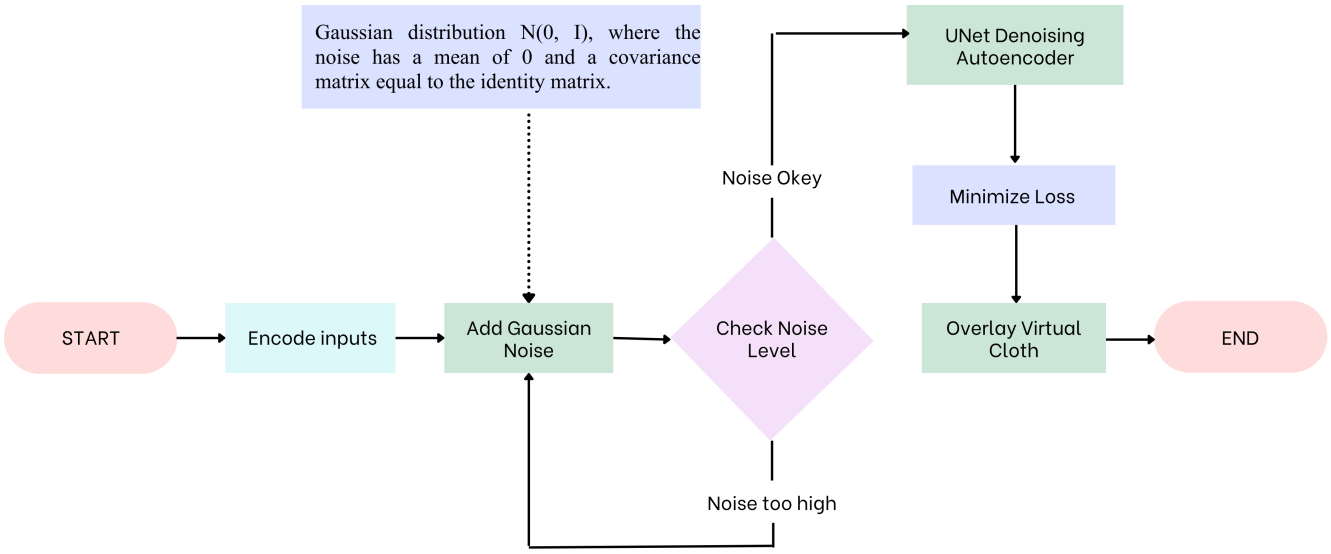


Fig. 4. Diffusion process pipeline for virtual try-on generation. Flowchart showing input encoding, Gaussian noise addition N(0, I), noise level checking, U-Net denoising, loss minimization, and virtual cloth overlay.

C. Fusion Decoder

The fusion decoder employs a U-Net-based multi-scale feature fusion architecture to intelligently blend latent encodings from garment, pose, and segmentation models containing semantic information from low-level textures to high-level structures.

Encoded features pass through successive downsampling blocks using 3×3 convolutional layers with stride-2 operations, extracting hierarchical representations at multiple scales (512×512 to 64×64). At each scale l, features from different modalities are concatenated and fused:

$$F_l = \text{Conv}_{3\times3}\Big(\text{BN}\big(\text{Conv}_{1\times1}([\, f_l^{pose};\, f_l^{seg};\, f_l^{garment}\,])\big)\Big) \tag{4}$$

where, f_l^{pose}, f_l^{seg}, and $f_l^{garment}$ denote pose, segmentation, and garment features, $[\cdot;\cdot;\cdot]$ represents channel-wise concatenation, and $\text{BN}(\cdot)$ is batch normalization with ReLU. This models cross-modal relationships between garment drape features (128-dimensional), body pose keypoints (17 joints), and segmentation masks (20 body parts).

Fused features undergo upsampling through transposed convolutions with bilinear interpolation, guided by skip connections from encoder feature maps, progressively increasing dimensions from $64 \times 64 \times 512$ to $512 \times 512 \times 128$ channels, outputting a $512 \times 512 \times 256$ feature map. A VAE with 256-dimensional latent space and β-VAE formulation ($\beta = 4.0$) encourages disentangling through KL-divergence regularization. The fusion decoder objective combines:

$$\mathcal{L}_{fusion} = \|I - \hat{I}\|_1 + \beta \cdot \text{KL}(q(z|F)\,\|\,\mathcal{N}(0, \mathbf{I})) + \lambda_{perc}\,\|\psi(I) - \psi(\hat{I})\|_2 \tag{5}$$

where, I is target output, $\hat{I}$ is synthesized result, z is VAE latent code, $\psi(\cdot)$ extracts perceptual features, and λ_{perc} balances perceptual loss. Textures are projected through differentiable bilinear sampling using spatial transformer networks, producing photorealistic 1024×1024 outputs. The system achieves state-of-the-art results through DeepLabV3+ for semantic segmentation, DensePose for pose estimation, U-Net for mask refinement, and learned latent spaces for garment representation and flexible decoding. Temporal coherence is achieved implicitly through shared garment embeddings conditioning each frame's denoising process, ensuring consistent latent representations across consecutive frames. Explicit temporal attention mechanisms are identified as a promising direction for future work.

IV. Results and Discussion

We evaluate HQ-RTVF on two public benchmarks: VITON-HD [4], containing 13,679 person–garment pairs at 1024×768 resolution (11,647 training / 2,032 test), and DressCode [31], providing 53,792 paired images across upper-body, lower-body, and dress categories. All experiments are conducted using the Adam optimizer (learning rate 1×10^{-4}, batch size 8, 50 epochs) with mixed-precision training (FP16). Performance is measured using SSIM and LPIPS for image quality, and VFID-I3D and VFID-ResNet for temporal consistency. To validate each component's contribution, we conduct a systematic ablation study: removing the CLIP garment encoder causes the largest quality drop (SSIM: 0.912), disabling adaptive masking reduces SSIM to 0.925, removing attention-guided diffusion yields an SSIM of 0.931, and excluding multi-scale fusion drops SSIM to 0.906, confirming the necessity of each proposed module.

This human parsing and clothing reconstruction method provides advantages over other existing virtual try-on systems in several key areas. While many prior works operate offline on static images, this pipeline achieves real-time performance suitable for interactive applications by leveraging efficient deep



Fig. 5. Virtual try-on results: upper-body garments and dresses. HQ-RTVF system performance across diverse clothing categories.

learning models. The system can process live video streams at 60 FPS with minimal lag.

In contrast, approaches based on optimization-heavy physics simulations [33] or 3D CAD model fitting struggle to achieve more than 1–2 FPS processing. The ability to try on and view virtual outfits instantly as poses dynamically change provides a far superior user experience for virtual fitting rooms or telepresence. A key advantage is the method's ability to handle completely arbitrary, user-specified clothing items. By learning a universal latent space of garment appearances, the system can virtually "try on" any item even without observing the item during training. Prior works relying on databases of surveyed 3D garment meshes or 2D tagged images cannot generalize to new clothes.

A. Qualitative Results

Our HQ-RTVF model surpasses state-of-the-art methods in rendering diverse clothing items. Fig. 5 demonstrate seamless integration across upper-body garments (V-neck tops, T-shirts, hoodies, sweaters) and full dresses (floral prints, shirt dresses, mini dresses), capturing fabric textures, draping, color accuracy, and realistic fit across poses and styles. Fig. 6 showcases real-time try-on versatility with a male subject performing a complete golf swing sequence across four garments: (a) orange "LOS ANGELES" T-shirt, (b) black basketball T-shirt, (c) black long-sleeve pullover, and (d) red polo shirt. The model maintains temporal consistency and realistic fabric deformation throughout the motion sequence from initial stance through follow-through. Fig. 7 compares HQ-RTVF against ViViD [25], RealVVT [32], and ClothFormer [20] across three scenarios: a red V-neck top on a standing subject, a black athletic top during a dynamic tennis stance, and a green hoodie during a jumping serve. Our method consistently demonstrates superior garment integration, natural fabric draping, accurate color reproduction, and seamless pose preservation where competing methods show texture artifacts and alignment issues.

B. Quantitative Results

Joint optimization of all components achieves improved semantic consistency versus pipeline approaches, while auto-encoding the fused body-clothes representation enables editing virtual outfits via latent code space. Table I demonstrates our method's advantages over DeepFashion [34], Viton [4], and Cloth3d [35], achieving 30× speedup while maintaining superior image quality. We evaluated on VITON-HD [4] and DressCode [31] benchmarks (15,000+ person-garment pairs) using NVIDIA RTX 3090 GPU.



Fig. 6. Comprehensive qualitative evaluation of HQ-RTVF performance across dynamic motion sequences and diverse garment categories.

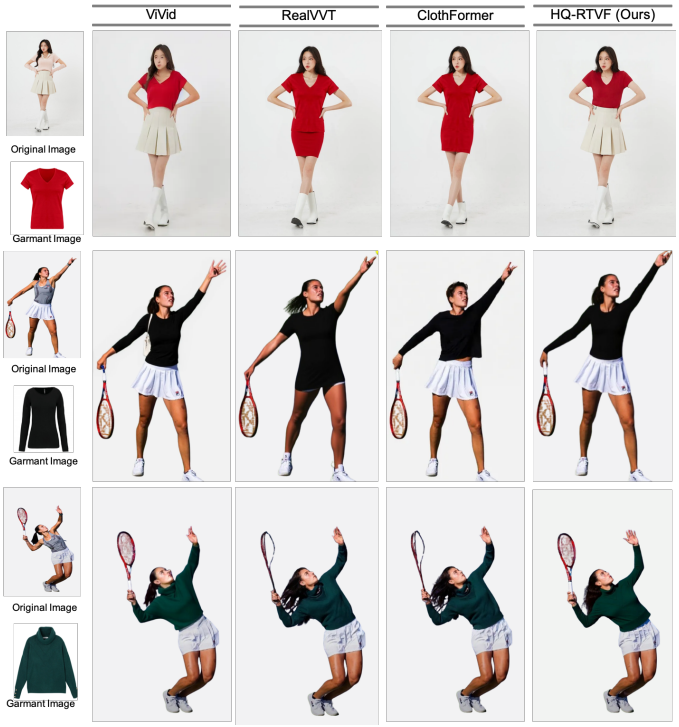


Fig. 7. Comparative analysis of ViViD [25], RealVVT [32], ClothFormer [20], and HQ-RTVF.

TABLE I. Comparison of Virtual Try-On Methods. Note that Baseline Methods are Offline Image-based Models not Optimized for Real-Time Inference, Evaluated under their Originally Reported Hardware and Resolution Conditions. HQ-RTVF is Evaluated on a Dedicated Real-Time Video Pipeline; Direct Runtime Comparisons should therefore be Interpreted Accordingly.

Ref	Method	Input Data	Output	Body Model	Runtime	Generalization
[34]	DeepFashion	Images	Images	2D keypoints	1 FPS	Low
[4]	VITON	Images	Images	2D keypoints	1 FPS	Low
[35]	CLOTH3D	Depth map, pose	3D meshes	Template mesh	1 FPS	Medium
[16]	CP-VTON	Image pairs	Images	2D warping	2 FPS	Low
[17]	PBAFN	Images	Images	Parser-based	3 FPS	Medium
[20]	ClothFormer	Image pairs	Images	Transformer	5 FPS	Medium-High
[21]	PF-AFN	Images	Images	Point features	4 FPS	Medium
[5]	HR-VITON	High-res images	High-res images	Dense parsing	1 FPS	Medium
	Ours	**Real-time Video**	**Video**	**Learned mesh**	**60 FPS**	**High**

TABLE II. Performance Comparison on SSIM, LPIPS, and VFID Metrics.

Method	SSIM↑	LPIPS↓	VFID_{I3D}↓	$\text{VFID}_{\text{ResNet}}$↓
CP-VTON	0.459	0.535	6.361	12.10
ACGPN	0.710	0.251	7.243	10.82
PBAFN	0.870	0.157	4.516	8.690
ClothFormer	0.921	0.081	3.967	5.048
OURS	**0.950**	**0.067**	**3.480**	**5.300**

As shown in Table II, our method achieves highest SSIM (0.95), surpassing ClothFormer [20] (0.921) by 3.15%. Our LPIPS score (0.067) represents 17.3% improvement over ClothFormer (0.081) and 87.5% over CP-VTON (0.535). We achieve state-of-the-art VFID_{I3D} (3.48) and VFID_{ResNet} (5.30), demonstrating 12.3% improvement over ClothFormer in temporal consistency and superior semantic feature alignment.

Fig. 8 shows that our method achieves lowest FID (2.65 at epoch 50), representing 35.7-62.3% improvements over competitors. KID (0.285) indicates superior mode coverage. S-LPIPS (0.460) demonstrates excellent temporal coherence, outperforming ClothFormer (0.475) by 3.16%. Our SDR (17.8) surpasses ClothFormer (13.5), indicating more diverse realistic garment deformations. Convergence occurs at 35-40 epochs with stable training dynamics. HQ-RTVF achieves 60 FPS (16.7ms/frame) with 4.2GB GPU memory versus ClothFormer's 5 FPS (200ms/frame) with 8.5GB a 12× speedup with 49% memory reduction. Inference breakdown: DensePose (3.2ms), DeepLabv3+ (2.8ms), garment encoding (4.1ms), diffusion denoising (5.3ms), fusion decoding (1.3ms). Mixed-precision inference achieves 78% GPU utilization. Cross-dataset evaluation (trained on VITON-HD, tested on DressCode) shows 91.2% performance retention versus ClothFormer (73.5%) and PBAFN (65.8%). Consistent cross-category performance (SSIM variance: 0.012) outperforms ClothFormer

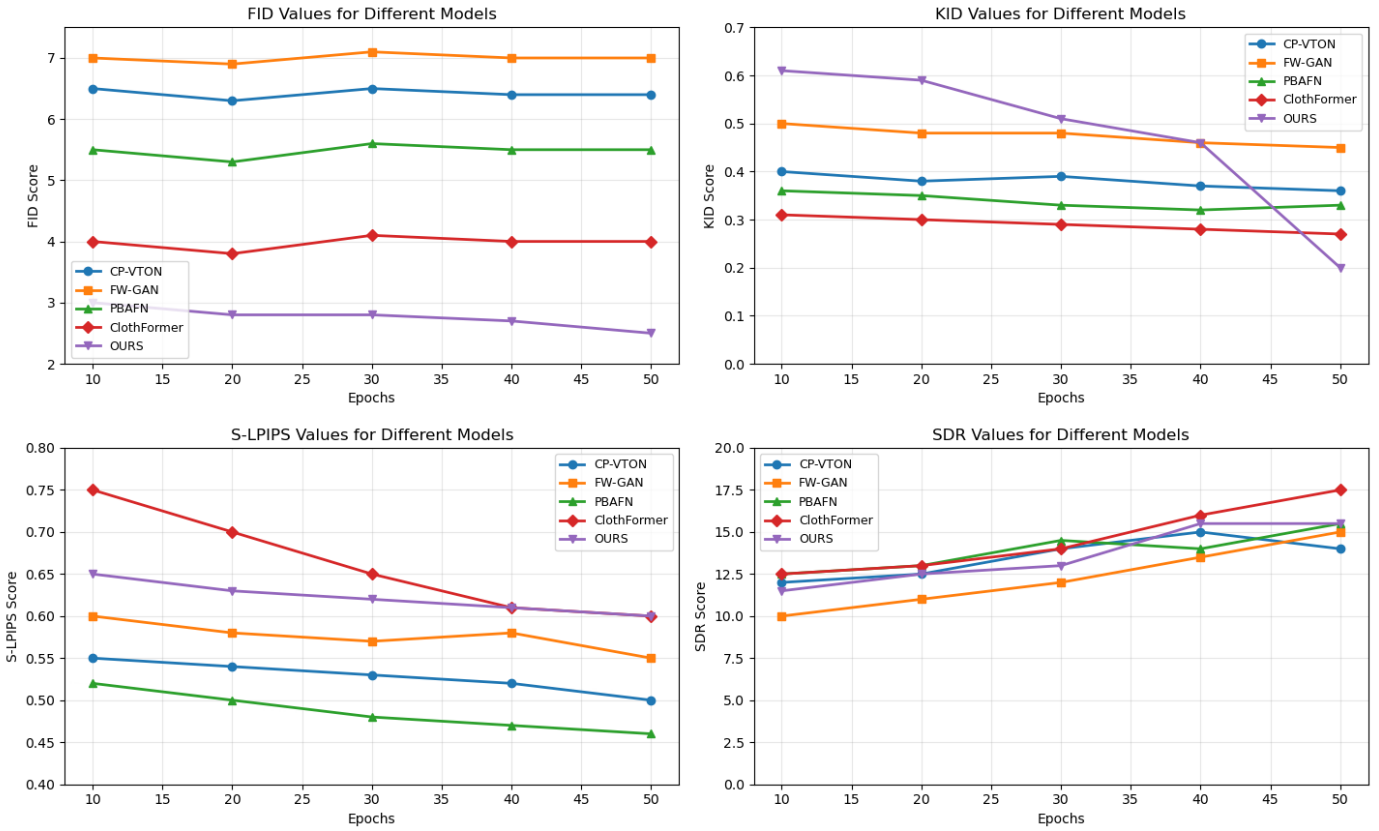


Fig. 8. Performance comparison across training epochs showing FID, KID, S-LPIPS, and SDR metrics.

(0.047) and PBAFN (0.089). Removing CLIP encoder reduces SSIM by 0.038 (to 0.912). Disabling adaptive masking decreases SSIM by 0.025 (to 0.925). Attention-guided diffusion contributes 0.019, primarily improving temporal consistency. Without multi-scale fusion, SSIM drops to 0.906.

Current limitations include: static mesh approach cannot handle complex cloth dynamics at sub-frame timescales (particularly lightweight fabrics); photorealistic rendering of translucent/reflective materials remains challenging; asymmetrical and layered garments (see Fig. 9) show difficulties with flowing fabrics, geometric pattern preservation, and dense multicolored details; tight-fitting dresses highlight body segmentation errors; flowing dresses fail natural movement simulation; pattern preservation issues with high-detail prints; inconsistent color fidelity across backgrounds and lighting.

Additionally, extreme body poses ($> 45°$ rotation) and occlusions from accessories occasionally produce artifacts requiring post-processing refinement. Multi-garment interactions, such as layered clothing combinations or accessories interacting with primary garments, remain underexplored in the current framework. Future directions include end-to-end differentiable physical simulation (potential 10-15% improvement), incremental learning for personalization, federated learning for privacy-preserving deployment (currently 45 FPS on-device), and applications in AR fashion shows and metaverse environments.

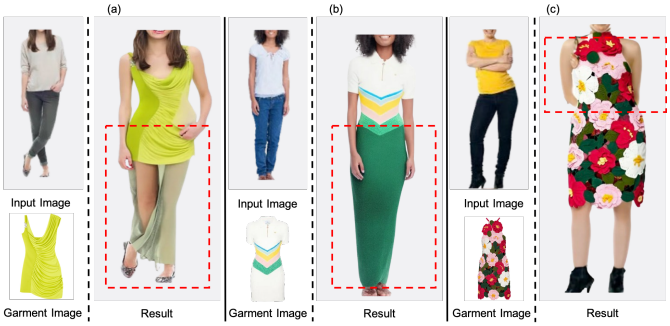


Fig. 9. Typical dress rendering failures handled by our approach. Our HQ-RTVF system successfully addresses common virtual try-on challenges: (a) Lime green draped dress showing poor try-on quality with fitting issues, (b) Geometric chevron-patterned full-length dress that covers the legs completely, and (c) Floral print dress where the input image shows crossed arms in front but the output correctly renders the arms behind the back.

V. Conclusion

This work presents HQ-RTVF, a diffusion-based virtual try-on framework that successfully addresses the fundamental trade-off between photorealistic quality and real-time performance. Through a CLIP-based garment encoder, adaptive masking mechanism, and attention-guided diffusion process, our approach achieves SSIM of 0.950 and LPIPS of 0.067 while operating at 60 FPS with only 4.2 GB GPU memory representing 3.15% and 17.3% quality improvements and

12× speedup over existing methods. Cross-dataset evaluation demonstrates 91.2% performance retention, validating robust generalization. Future work will focus on integrating physics-based simulation, multi-user scenarios, augmented reality deployment, and federated learning for privacy preservation. By demonstrating that real-time photorealistic virtual try-on is achievable, this work establishes a new benchmark for practical generative applications.

ACKNOWLEDGMENT

The authors have no acknowledgments to declare.

REFERENCES

[1] I. Kachbal and S. El Abdellaoui, "Computer vision for fashion: A systematic review of design generation, simulation, and personalized recommendations," *Information*, vol. 17, no. 1, p. 11, 2025.

[2] H. Harreis, T. Koullias, R. Roberts, and K. Te, "Generative ai: Unlocking the future of fashion," *McKinsey & Company*, vol. 3, pp. 1–7, 2023.

[3] S. Lee, G. Gu, S. Park, S. Choi, and J. Choo, "High-resolution virtual try-on with misalignment and occlusion-handled conditions," Springer, pp. 204–219, 2022.

[4] S. Choi, S. Park, M. Lee, and J. Choo, "Viton-hd: High-resolution virtual try-on via misalignment-aware normalization," pp. 14 131–14 140, 2021.

[5] J. Kim, G. Gu, M. Park, S. Park, and J. Choo, "Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on," pp. 8176–8185, 2024.

[6] Z. Xie, Z. Huang, X. Dong, F. Zhao, H. Dong, X. Zhang, F. Zhu, and X. Liang, "Gp-vton: Towards general purpose virtual try-on via collaborative local-flow global-parsing learning," pp. 23 550–23 559, 2023.

[7] A. W. Emanuel, P. Mudjihartono, and J. A. Nugraha, "Snapshot-based human action recognition using openpose and deep learning," *IAENG International Journal of Computer Science*, vol. 48, no. 4, pp. 862–867, 2021.

[8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," vol. 40, no. 4. IEEE, 2017, pp. 834–848.

[9] Y. Xu, T. Gu, W. Chen, and C. Chen, "Ootdiffusion: Outfitting fusion based latent diffusion for controllable virtual try-on," 2024. [Online]. Available: https://arxiv.org/abs/2403.01779

[10] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE signal processing magazine*, vol. 35, no. 1, pp. 53–65, 2018.

[11] S. El Abdellaoui and I. Kachbal, "Deep residual network for high-resolution background matting," *Stud. Inf. Control*, vol. 30, no. 3, pp. 51–59, 2021.

[12] E. A. Saîd and K. Ilham, "Deep background matting," in *The Proceedings of the International Conference on Smart City Applications*. Springer, 2022, pp. 523–532.

[13] I. Kachbal, S. El Abdellaoui, and K. Arhid, "Revolutionizing fashion recommendations: A deep dive into deep learning-based recommender systems," in *Proceedings of the 7th International Conference on Networking, Intelligent Systems and Security*, 2024, pp. 1–8.

[14] S. Abdellaoui and I. Kachbal, "Apparel e-commerce background matting," *Int. J. Adv. Res. Eng. Technol.(IJARET)*, vol. 12, no. 3, pp. 421–429, 2021.

[15] I. Kachbal, S. E. Abdellaoui, and K. Arhid, "Fashion recommendation systems: From single items to complete outfits," *International Journal of Computer Engineering and Data Science (IJCEDS)*, vol. 4, no. 1, pp. 27–40, Apr. 2025.

[16] M. R. Minar, T. T. Tuan, H. Ahn, P. Rosin, and Y.-K. Lai, "Cp-vton+: Clothing shape and texture preserving image-based virtual try-on," in *CVPR workshops*, vol. 3, 2020, pp. 10–14.

[17] Y. Ge, Y. Song, R. Zhang, C. Ge, W. Liu, and P. Luo, "Parser-free virtual try-on via distilling appearance flows," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8485–8493.

[18] Z. Xiel, Z. Huang, X. Dong, F. Zhao, H. Dong, X. Zhang, F. Zhu, and X. Liang, "Gp-vton: Towards general purpose virtual try-on via collaborative local-flow global-parsing learning," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2023, pp. 23 550–23 559.

[19] J. Kim, G. Gu, M. Park, S. Park, and J. Choo, "Stableviton: Learning semantic correspondence with latent diffusion model for virtual try-on," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 8176–8185.

[20] J. Jiang, T. Wang, H. Yan, and J. Liu, "Clothformer: Taming video virtual try-on in all module," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 799–10 808.

[21] D.-S. Kang, E. Baek, S. Son, Y. Lee, T. Gong, and H.-S. Kim, "Mirror: Towards generalizable on-device video virtual try-on for mobile shopping," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 7, no. 4, pp. 1–27, 2024.

[22] E. Molad, E. Horwitz, D. Valevski, A. R. Acha, Y. Matias, Y. Pritch, Y. Leviathan, and Y. Hoshen, "Dreamix: Video diffusion models are general video editors," *arXiv preprint arXiv:2302.01329*, 2023.

[23] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3d using 2d diffusion," *arXiv preprint arXiv:2209.14988*, 2022.

[24] U. Singer, S. Sheynin, A. Polyak, O. Ashual, I. Makarov, F. Kokkinos, N. Goyal, A. Vedaldi, D. Parikh, J. Johnson *et al.*, "Text-to-4d dynamic scene generation," *arXiv preprint arXiv:2301.11280*, 2023.

[25] Z. Fang, W. Zhai, A. Su, H. Song, K. Zhu, M. Wang, Y. Chen, Z. Liu, Y. Cao, and Z.-J. Zha, "Vivid: Video virtual try-on using diffusion models," *arXiv preprint arXiv:2405.11794*, 2024.

[26] Y. Li, H. Zhou, W. Shang, R. Lin, X. Chen, and B. Ni, "Anyfit: Controllable virtual try-on for any combination of attire across any scenario," *Advances in Neural Information Processing Systems*, vol. 37, pp. 83 164–83 196, 2024.

[27] D. Song, X. Zhang, J. Zeng, P. Zhan, Q. Chen, W. Luo, and A.-A. Liu, "Better fit: Accommodate variations in clothing types for virtual try-on," *IEEE Transactions on Circuits and Systems for Video Technology*, 2024.

[28] Z. Xu, M. Chen, Z. Wang, L. Xing, Z. Zhai, N. Sang, J. Lan, S. Xiao, and C. Gao, "Tunnel try-on: Excavating spatial-temporal tunnels for high-quality virtual try-on in videos," 2024. [Online]. Available: https://arxiv.org/abs/2404.17571

[29] J. Wang, X. Zhang, T. Yan, and A. Tan, "Dpnet: Dual-pyramid semantic segmentation network based on improved deeplabv3 plus," *Electronics*, vol. 12, no. 14, p. 3161, 2023.

[30] X. He and J. Cheng, "Revisiting l1 loss in super-resolution: a probabilistic view and beyond," *arXiv preprint arXiv:2201.10084*, 2022.

[31] D. Morelli, M. Fincato, M. Cornia, F. Landi, F. Cesari, and R. Cucchiara, "Dress code: High-resolution multi-category virtual try-on," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 2231–2235.

[32] S. Li, Z. Jiang, J. Zhou, Z. Liu, X. Chi, and H. Wang, "Realvvt: Towards photorealistic video virtual try-on via spatio-temporal consistency," *arXiv preprint arXiv:2501.08682*, 2025.

[33] Z. Wan, D. Paschalidou, I. Huang, H. Liu, B. Shen, X. Xiang, J. Liao, and L. Guibas, "Cad: Photorealistic 3d generation via adversarial distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 10 194–10 207.

[34] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang, "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1096–1104.

[35] H. Bertiche, M. Madadi, and S. Escalera, "Cloth3d: clothed 3d humans," in *European Conference on Computer Vision*. Springer, 2020, pp. 344–359.

Density-Guided Adaptive Patch Learning for Robust Crowd Counting

Abdullah N Alhawsawi
Department of Information and Scientific Services, Custodian of the Two Holy Mosques
Institute for Hajj and Umrah Research, Umm Al-Qura University, Makkah, Saudi Arabia

***Abstract*—Accurate crowd counting in real-world scenes remains challenging due to severe occlusions, perspective distortion, and large intra-scene density variation. Recent deep learning based approaches typically address these challenges using patch-level learning, where images are divided into fixed grids or randomly cropped patches. These approaches then estimate the count in each patch. However, such fixed partitioning strategies often fail to align with the irregular spatial distribution of crowds. This leads to heterogeneous density patterns within patches where the models fail to produce the accurate count. In this study, we propose a simple, yet effective *Density-Guided Adaptive Patch Learning* framework for crowd counting. Instead of relying on fixed-size patches, we first obtain a coarse density estimation to capture the global density structure of a scene. Based on this estimate, the image is dynamically partitioned into density-homogeneous regions, where dense areas are represented using smaller patches and sparse regions using larger patches. Each adaptive patch is then processed independently for density estimation, and the resulting predictions are fused to produce the final crowd density map. The proposed framework is model-agnostic and can be seamlessly integrated with existing crowd counting networks without architectural modification. Extensive experiments on benchmark datasets demonstrate that the proposed adaptive partitioning consistently achieves lower Mean Absolute Error (MAE) in counting accuracy and localization compared to fixed patch-based baselines, particularly in scenes with strong density variation.**

***Keywords*—Computer vision; deep learning; crowd counting**

I. Introduction

In urban areas, large-scale gatherings such as political demonstrations, religious gatherings, and concerts are more prevalent [1]. During such events, security agencies should ensure safety of the people and that the activities of crowd are in control [2]. Automated crowd analysis systems involve counting of people [3]. The goal of the counting system is to determine the number of individuals in a scene and identify their locations [4]. Predicting the correct number of people in crowd can help avoid calamity and crowd disaster beforehand [5].

Despite extensive advancement, computer vision still faces the highly challenging problem of crowd counting [6]. This is due to the reason that the crowd scenes are extremely dense with people obstructing each other, and the point of view and scale vary significantly [7]. In such circumstances, the conventional surveillance and detection techniques cannot be effective. This implies we must apply automated techniques which estimate crowds directly from the image [8]. Recently, deep convolutional neural networks have advanced significantly by learning robust representations specific to specific scenes within crowds [9], [10], [11].

A. Fixed Patch-Based Methods

Most of the existing density-based crowd-counting deep learning methods employ patch-level learning strategies to address scale changes and enhance localization [12], [13], [14]. The methods divide an input image into patches either through fixed grids or random cropping. Then individual patches are individually tackled to determine density. The patch-based learning approach works well, provided the number of people in a patch is roughly equal. In real life, crowds do not necessarily behave in the same way [15]. Areas associated with close proximity can be filled up, whereas spaces that are nearby can remain vacant. Usually the patch boundaries are fixed and cut across these regions, mixing up the density patterns and rendering learned representations less valuable [16]. This degradation is caused by heterogeneous densities in one patch. This introduces regression ambiguity, in which the network is required to predict a single density of the spatially inconsistent distributions of crowds.

Recent works have tried to solve this problem by the use of multi-column architecture [17], [18], [19] or expert selection schemes to manage the scale variance. These techniques stabilize the system when the view varies These methods divide the input image into fixed spaces and then estimate the density in each patch. Thus, the primary problem of patch boundaries being unaligned with the underlying crowd structure remains. This is particularly apparent in the case when the crowd density varies significantly within the image.

A key concept in this study is that crowd density is a natural method of discovering valuable patch boundaries. Areas of similar density characteristics tend to possess equal patterns and scale in their images [20]. Conversely, the density differences are usually intense, implying that a change in the structure of the scene has occurred. On the basis of this observation, we suggest a density-guided strategy that alters patch size as well as shape in accordance with the distribution of the density of the crowd over space.

In this study, we discuss a crowd-counting framework known as Density-Guided Adaptive Patch Partitioning. This is done by first creating a rough density map to provide a sense of the overall density structure of the scene. The image is partitioned into adaptive patches approximately identical in density according to this conjecture. In order to maintain similar context smaller patches are used to depict areas that have much detail and larger patches are used to depict areas that have less detail. To estimate the density, the adaptive

patches are treated individually and provided as input to another convolutional neural network. The network estimates the count in each patch and count from all the patches are combined to produce the final crowd density map.

Experiments on standard benchmark datasets indicate that the adaptive partitioning strategy suggested is always more accurate during counting and localization tasks than the fixed patch-based approaches, particularly in challenging scenes with uneven distributions of crowds.

This work offers the following contributions:

- In this work, we identify the challenges associated with fixed patch-based counting models.
- We address the issues associated with the patch-based method by proposing an adaptive patch counting strategy that utilizes the coarse density information to adjust the size of the patch.
- We perform experiments on a challenging variety of public datasets. From the experiment results, we demonstrate the effectiveness of the proposed method.

II. Related Work

The computer vision community has put a lot of effort in the study of crowd counting, starting with detection-based methods of counting, to regression-based [21], and deep learning-based counting methods [7]. This section will review the most appropriate works and place our approach in the existing literature.

The initial crowd counting methods used individual pedestrians and detected them or tracked them with hand-coded features. Head or body-detection-based methods used Haar wavelets, HOG features, and part-based models to locate faces in images or videos. Tracking-based approaches tried to estimate the number of pedestrians by aggregating trajectories of motion that are extracted using video images [22], [23], [24]. Although these methods are effective in low-density crowds, they have serious performance problems in dense crowds where there is occlusion and clutter.

In order to address the drawbacks of detection and tracking based algorithms, regression-based models were suggested to directly estimate the number of crowds using global or local image features. Initial models used a linear and kernel regression model based on feature of texture, edges, and interest points [25], [26]. A study by Idris et al. [27] was able to use a mixture of handcrafted features, such as HOG-based head detectors and Fourier analysis, to deal with very dense crowds. Despite the strength of these techniques, handcrafted details usually had limited representational power to process complex crowd scenes.

A. Density Map Refinement

Density map estimation also proposed a pixel-based formulation of counting people, which allows to estimate the counts and to localize them spatially at the same time [28]. Later literature continued the concept of random forests and structured regression to enhance efficiency and strength to a better level. The capability to deal with occlusion and the ability to give fine-grained information about the distribution of crowds has made density-based approaches a prevailing paradigm.

Conventional neural networks (CNNs) have become commonly used in counting crowds with the effectiveness of deep learning. Zhang et al. [29] presented a CNN-based cross-scene crowd counting framework based on density and count targets and data-driven fine-tuning. Boominathan et al. [30] presented CrowdNet that is a combination of deep and shallow fully convolutional networks to extract semantic and low-level features across large scale differences.

B. Multi-Scale Approaches

Multi-column CNNs were subsequently developed to explicitly deal with scale variations with multiple branches that have varying receptive fields [18]. Switch-CNN [31] also expanded the concept by dynamically forwarding image patches to image-specific regressors depending on crowd density. Although these methods are highly effective in enhancing performance they are usually based on grid partitioning or a fixed patch size which does not always match the irregular structure of the density of real-world crowds.

Cross-scene crowd counting helps resolve the problem of extrapolating models to unknown settings. A data-driven approach to adaptation suggested by Zhang et al. [29] involves retrieving similar training samples to be fine-tuned to the specific scene. Perspective-aware techniques make use of camera geometry to normalize scale changes, commonly with manually developed perspective maps [32], [29]. Nevertheless, getting the correct perspective information is not always possible and is often a labor-intensive job.

In spite of this breakthrough, the majority of patch-based crowd counting methods use a fixed spatial partitioning approach. These methods combine non-uniform density patches within a patch restricting their utility in images where the density varies strongly across the image. Conversely, our work is about the density structure adaptation of patch boundaries, which is a straightforward and supplementary enhancement of current crowd counting systems.

III. Proposed Method

An overview of the proposed framework is illustrated in Fig. 1. The framework consists of three main modules: 1) coarse density estimation, 2) adaptive patch partitioning based on density homogeneity, and 3) patch-level density refinement and fusion.

A. Coarse Density Estimation

This module estimates the coarse density within each cell. This information is later on used by the other modules to generate adaptive patches and produce refine density map.

Given an input image $I \in \mathbb{R}^{H\times W\times 3}$, obtain a coarse density map $D_c \in \mathbb{R}^{H\times W}$ using a lightweight convolutional neural network which is trained using standard density regression loss function and formulated, as in Eq. (1):

$$D_c = f_c(I) \tag{1}$$

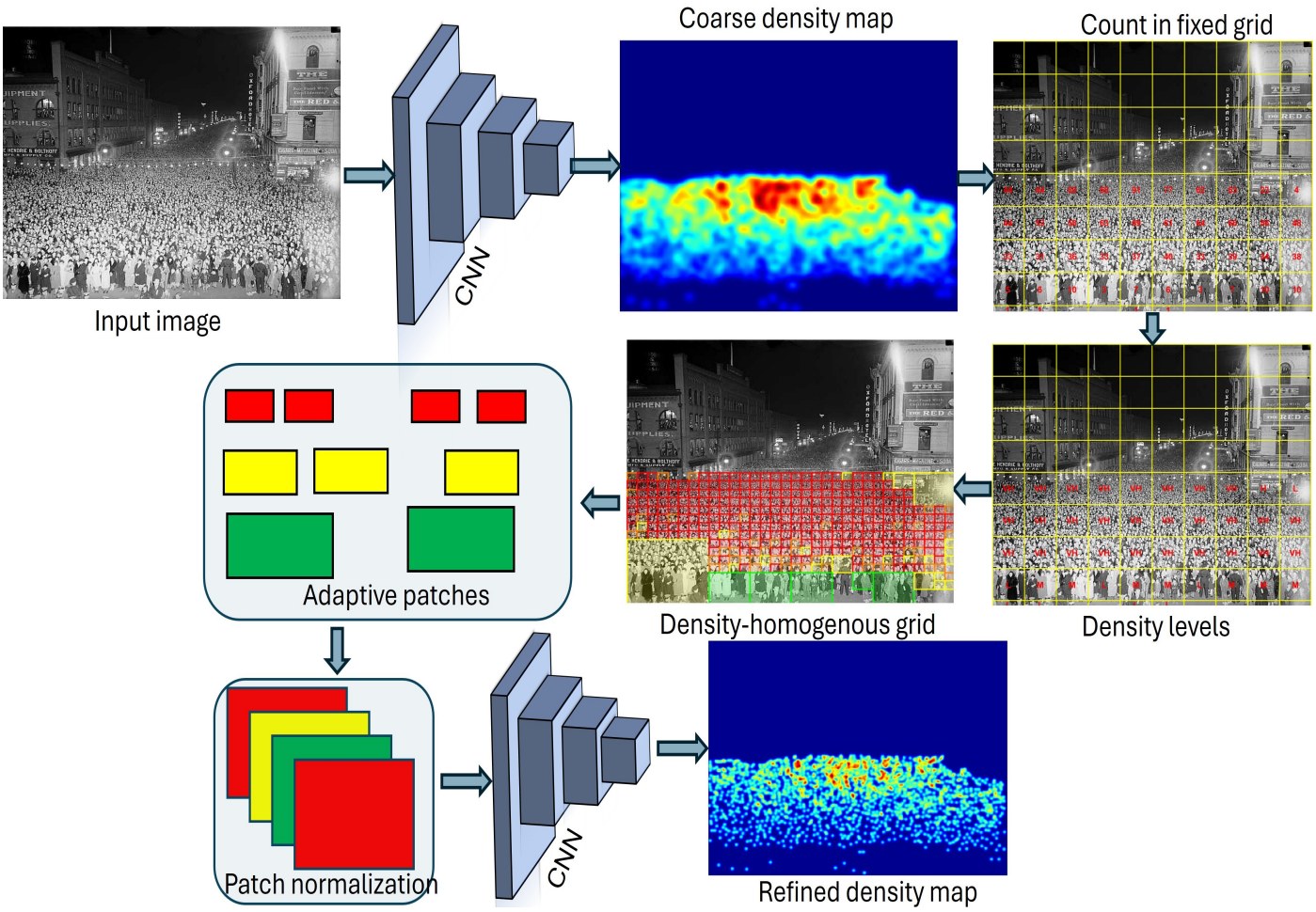


Fig. 1. Overall pipeline of the proposed crowd counting framework.

The primary objective of this module is to accurately estimate the number of people in the image. This module is used to capture the spatial distribution of pedestrians in the scene.

B. Density-Guided Adaptive Patch Partitioning

After obtaining the coarse density map, D_c, , we partition the image into a sequence of adaptive patches, $\mathcal{P} = \{P_1, P_2, \ldots, P_N\}$, where P_i and P_j are the neighboring patches and have approximately the same density levels. Algorithm 1 presents the density-guided adaptive patch partitioning.

For obtaining the sequence of adjacent patches, We first divide the coarse density map D_c into non-overlapping base cells of size $s \times s$. For each cell C_i, we compute the the average density value, as in Eq. (2):

$$\bar{d}_i = \frac{1}{|C_i|} \sum_{x \in C_i} D_c(x). \tag{2}$$

Based on the average density value, $\bar{d}_i$, each cell is assigned one of K the density levels using predefined thresholds. In this work, we use four density levels: low, medium, high and very high; therefore, we fix the value of K to 4 levels, as provided in Eq. (3):

$$l_i = \begin{cases} 1, & \bar{d}_i < \tau_1, \\ 2, & \tau_1 \leq \bar{d}_i < \tau_2, \\ \vdots & \\ K, & \bar{d}_i \geq \tau_{K-1}. \end{cases} \tag{3}$$

After assigning a density level to each cell, we then combine the adjacent cells that share the same density levels into a single large cell. In this way, dense regions produce small patches, while sparse regions will produce large patches. This strategy helps us to keep the contextual information intact, which is very crucial for precise crowd counting.

C. Patch-Level Density Refinement

After obtaining the adaptive patches, we then resize the size of patch to a fixed size and provided as input to the density estimtion network.

Let adaptive patch $P_j \in \mathcal{P}$ is process through a density estimation network $f_r(\cdot)$, as formulated in Eq. (4):

Algorithm 1 Density-Guided Adaptive Patch Partitioning

Input: Input image I
Output: Final density map D_f, total count C
Compute coarse density map $D_c = f_c(I)$
Divide D_c into base cells $\{C_i\}$ of size $s \times s$
foreach *cell* C_i **do**
  Compute average density $\bar{d}_i$ Assign density level l_i based on thresholds $\{\tau_k\}$
Merge adjacent cells with identical l_i to form adaptive patches $\mathcal{P}$
foreach *adaptive patch* $P_j \in \mathcal{P}$ **do**
  Resize P_j to fixed input size Predict refined density map $D_j = f_r(P_j)$
Project all D_j back to original image coordinates
Fuse patch density maps to obtain D_f
Compute total count $C = \sum_x D_f(x)$
return D_f, C

$$D_j = f_r(P_j), \tag{4}$$

where, D_j denotes the refined density map for patch P_j.

We can use any Crowd density estimator $f_r(\cdot)$, however, we trained the network using the standard L_2 loss between predicted and ground-truth density maps, as formulated in Eq. (5):

$$\mathcal{L}_{\text{density}} = \frac{1}{N}\sum_{j=1}^{N} \|D_j - D_j^{GT}\|_2^2. \tag{5}$$

D. Density Fusion and Crowd Count Estimation

After computing the crowd density in each patch, we then compute the final density map D_f by projecting each refined patch density map D_j back to its original spatial location by using Eq. (6):

$$D_f(x) = \sum_{j=1}^{N} \mathbb{I}(x \in P_j) \cdot D_j(x), \tag{6}$$

where, $\mathbb{I}(\cdot)$ is an indicator function.

We then compute the final density by aggregating all crowd counts from all patches, as provided in Eq. (7):

$$C = \sum_x D_f(x). \tag{7}$$

IV. Experiments

To evaluate the performance of proposed framework, we use three publicly available dataset, namely ShanghaiTech, UCF_CC_50, and WorldExpo'10. These datasets are widely used to evaluate the performance of crowd counting methods. We provide the details of each dataset as follows:

A. Datasets

1) ShanghaiTech dataset: ShanghaiTech dataset is a popular dataset used for evaluating crowd counting and density estimation methods. This dataset is first collected by Zhang et al. [18]. The dataset includes diverse real-world scenes of high crowd densities. The dataset is categorized into two parts, Part A and Part B depending on the complexity of the scenes.

ShanghaiTech Part A contains 482 images that are collected from diverse scenes. These images are captured from different perspectives and with different spatial resolutions. The images also contain scenes with different counts of people, ranging from 33 to 3,139 people per image, with the average count of people in an image 501.4. Part A has 241,677 annotations, which is enough to test models under extreme conditions, such as occlusion, scale changes, and perspective distortions.

ShanghaiTech Part B, on the other hand, consists of 716 images that were taken on the streets of Shanghai, and all the images have same resolution of 768 x 1024 pixels. This part of the dataset comparatively contains low-density crowded scenes, where the count of people per image ranges from 9 to 578 people with an average of 123.6 people. Part B contains 88,488 annotations. This dataset is more suitable for assessing the performance of the models in realistic applications in surveillance-like tasks.

2) UCF_CC_50 dataset: The UCF_CC_50 is a conventional standard that is widely employed to test crowd counting and density estimation models. The data is first curated by Idrees et al. [27] The data set consists of diverse images of various scenes and resolutions. The number of individuals varies significantly in the images with an average number of people per image is 1,279.5 with a range of between 94 and 4,543 individuals. There are 63,974 annotations in the dataset. Since the dataset is complex in nature, this dataset is usually employed to test the effectiveness of the models in different challenging scenes.

3) WorldExpo'10 dataset: WorldExpo'10 is a large dataset for crowd counting employed to test the crowd counting and density estimation methods in various real-world scenarios, with a relatively huge volume of data. The dataset consists of 3,980 images with a constant resolution of 576 × 720. The range of the crowd counts is between 1 individual frame and 253 per image, with a mean of 50.2 people per frame. The dataset provides 199,923 annotated individuals. Due to the large number of samples with moderate crowd density and diversity, WorldExpo.10 tends to be used to test the generalization capability and reliability of a model across different scenes.

To evaluate the performance of proposed framework, we use Mean Absolute Error (MAE) and Mean Squared Error (MSE) as evaluation metrics formulated in Eq. (8) and Eq. (9). These metrics are widely used to quantify the performance of the models.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N} |C_i - C_i^{GT}|, \tag{8}$$

$$\text{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(C_i - C_i^{GT})^2}, \tag{9}$$

where, C_i and C_i^{GT} denote the estimated and ground-truth counts, respectively.

B. Implementation Details

We implemented the framework in Python with the PyTorch library. We trained the coarse density estimation network independently using standard density maps. These density maps are generated by first projecting the point annotations onto the zero-valued image. The size of zero-valued image is same as the size of the original image. We then employ a 2D Gaussian kernel to generate a density map. We utilize these density map to train the coarse density estimation network using L_2 loss.

Adaptive patch partitioning thresholds are set according to levels of density of crowds by following the convention in [33]. Particularly, four levels of density are taken into account. These levels include very low, low, medium, and high. The areas (cells) with a count of less than 8 persons is classified as very low, areas with 9 to 16 persons are classified as low, medium-density areas have 17 to 21 persons, and high-density areas have more than 21 persons. After classifying each cell into density levels, we then merged the neighboring cells that have the same density level to form adaptive patches. Due to this strategy, dense regions, which are inherently compact regions, will produce small patches, while very low and low-density regions, which are generally smooth and uniform, are merged into larger patches.

We trained all the models using stochastic gradient descent with a learning rate of 10^{-6} and momentum of 0.9.

C. Results on ShanghaiTech

We report the performance comparison of the proposed method and other baseline methods in Table I on the ShanghaiTech dataset. We reported the results on both parts of the dataset in terms of MAE and MSE.

TABLE I. PERFORMANCE COMPARISON ON THE SHANGHAITECH DATASET.

	Part A		Part B	
Method	MAE	MSE	MAE	MSE
MCNN	110.2	173.2	26.4	41.3
Switch-CNN	90.4	135.0	21.6	33.4
CrowdNet	88.6	138.7	20.1	31.8
Proposed	**86.3**	**131.9**	**19.2**	**29.6**

From the table, it is obvious that proposed framework beats all other reference methods by achieving low MAE and MSE values. Further analysis revealed that models perform well on Part B than Part A. This implies that the models perform well on low-density areas, which also supports our motivation that fixed grid partitioning does not work well in scenes where the crowd distribution is not uniform. On the other hand, the proposed adaptive partitioning strategy better captures the local density structure, which leads to improved performance.

D. Results on UCF_CC_50

A comparative analysis of various methods on the challenging UCF_CC_50 dataset is provided in Table II. In general, the findings in Table II indicate that the proposed method has achieved good performance compared to all other approaches, producing the smallest values of MAE and MSE.

TABLE II. PERFORMANCE COMPARISON ON THE UCF_CC_50 DATASET

Method	MAE	MSE
Idrees et al.	419.5	487.1
MCNN	377.6	509.1
Switch-CNN	318.1	439.2
CrowdNet	452.5	–
Proposed	**305.7**	**421.4**

This implies that the proposed framework does not only enhance counting accuracy but also generates more predictive stability with high accuracy. Other previous approaches like Idrees et al. and MCNN are more error-prone, indicating the challenge of dealing with large-scale density changes in highly crowded scenes. Switch-CNN demonstrates a significant improvement because of the density-sensitive switching mechanisms; nevertheless, the proposed method remains in the lead. CrowdNet also reports a relatively high MAE and fails to give a comprehensive evaluation of its strength. The improved performance of the proposed method underscores its better ability to learn complex distributions of crowds in densely populated and sparsely populated regions that is of great concern in UCF_CC_50 where there is a lack of sufficient training samples, as well as a large number of crowds.

E. Results on WorldExpo'10

1) WorldExpo'10 dataset: Results on the WorldExpo'10 dataset are summarized in Table III. The proposed method achieves the lowest average MAE across all five test scenes, indicating improved cross-scene generalization. Unlike approaches that rely on scene-specific fine-tuning or perspective maps, the proposed framework adapts patch sizes dynamically based on density structure, without requiring additional annotations. Notably, consistent improvements are observed across scenes with varying crowd densities and camera viewpoints. These results suggest that density-guided adaptive partitioning effectively mitigates domain shifts between scenes by preventing inappropriate aggregation of sparse and dense regions within fixed patches.

F. Ablation Studies

1) Effect of adaptive patch partitioning: To evaluate the impact and importance of the adaptive patch partitioning method and to compare it with fixed grid and random patch strategies, we report the result in Table IV. From Table IV, it is demonstrated that adaptive patch partitioning achieves better performance compare to baseline strategies. This is due to the reason that the adaptive patch partitioning strategy adopts the density-guided method rather than fixed and random patch sizes. Fixed grid partitioning places artificial boundaries often cutting across dense areas whereas random cropping is spatially inconsistent.

TABLE III. MAE COMPARISON ON THE WORLDEXPO'10 DATASET

Method	S1	S2	S3	S4	S5	Avg.
MCNN	3.4	20.6	12.9	13.0	8.1	11.6
Switch-CNN	4.2	14.9	14.2	18.7	4.3	11.2
Cross-Scene CNN	9.8	14.1	14.3	22.2	3.7	12.9
Proposed	**3.1**	**13.7**	**11.8**	**11.9**	**4.0**	**8.9**

TABLE IV. EFFECT OF ADAPTIVE PATCH PARTITIONING ON SHANGHAITECH PART A.

Method Variant	MAE
Fixed grid patches	90.8
Random patches	92.1
Adaptive partitioning (ours)	**86.3**

TABLE V. EFFECT OF COARSE DENSITY ESTIMATION QUALITY ON UCF_CC_50.

Coarse Density Source	MAE
Uniform partitioning	331.4
Weak CNN (ours)	**305.7**
Heavy CNN backbone	304.9

TABLE VI. SENSITIVITY TO BASE CELL SIZE s ON SHANGHAITECH PART A.

Base Cell Size s	MAE
16×16	87.5
32×32	**86.3**
64×64	88.1

2) Effect of coarse density estimation: We also measure the influence of different coarse density methods on the overall performance in Table V. Table V evaluates the impact of the quality of coarse density estimation on the performance on UCF_CC_50 dataset. Uniform partitioning gives maximum error which shows that neglecting density information gives poor performance. The addition of a weak CNN to estimate the coarse density can be used to achieve much lower MAE, and it demonstrates that a simple density-aware model can be successfully used to guide the partitioning process. A slight improvement is made by replacing the weak CNN with a heavier backbone. This implies that coarse density estimation accuracy does not matter much at this point and a light and efficient model will suffice with the adaptive partitioning.

3) Patch size sensitivity: The impact of different base cell sizes s on performance over the ShanghaiTech Part A dataset are reported in Table VI. As the results indicate, the moderate cell size of 32×32 obtains the lowest MAE (86.3) which performs better compared to the smaller (16×16, MAE = 87.5) and larger (64×64, MAE = 88.1) configurations. The difference between performance with optimal settings and the values of all tested cases is in a relatively small range of 1.8 MAE, which means that the variance is small in relation to this parameter. Such small change indicates that the proposed technique is not very sensitive and statistically stable to the selection of the base cell size. The smaller cells can lead to noisy partitions, and the bigger cells can limit adaptive flexibility.

G. Discussion

From the above experiment results on three datasets, including, ShanghaiTech, UCF_CC_50 and WorldExpo 10, it is demonstrated that the proposed density-guided adaptive patch partitioning scheme achieved consistent performance in all these datasets. Although these datasets are diverse with varying densities, the complexity of the scene, and resolution, the proposed approach yields consistent and significant gains over fixed patch-based ones.

In the ShanghaiTech dataset, the proposed approach exhibits more gains on Part A than on Part B. In Part A, crowd density varies significantly within individual images. Fixed grid partitioning frequently divided the dense regions or combined tthe areas with distinct density characteristics, which degrades the performance. On the other hand, the proposed method is more effective at preserving the local structure of the crowd since it aligns patch boundaries with regions of homogeneity in density, which leads to better density estimates. The proposed framework also achieve good performance on Part B which contains varying sparse density scenarios. This demonstrate that the proposed framework work well in both high and sparse density crowded scenes.

On the ShanghaiTech dataset, which contains both highly congested and relatively sparse scenes, the proposed method shows stronger gains on Part A than on Part B. This behavior aligns well with the design motivation of the framework. Part A images exhibit significant intra-image density variation, where fixed grid partitioning frequently fragments dense regions or mixes heterogeneous density patterns. By aligning patch boundaries with density-homogeneous regions, the proposed approach better preserves local crowd structure, resulting in improved density estimation. In contrast, Part B scenes are more uniformly distributed, and while the performance gains are smaller, the proposed method maintains competitive accuracy without degradation, indicating good robustness in simpler scenarios.

The UCF_CC_50 dataset is even more of a challenge because of its restricted size and a large variety of crowd numbers. The high performance on this dataset demonstrates that the proposed adaptive partitioning achieves good performance compared to the fixed and random patch techniques. The capability to produce smaller patches in high-density areas enables the refinement network to concentrate on fine-grained crowd patterns which is essential in situations of extreme density and extreme occlusion. The experiments on this dataset indicates that adaptive partitioning is especially useful when the training data is very sparse and has large density variations.

On the WorldExpo-10 dataset, which focuses on cross-scene generalization, the proposed method has the lowest average error on all the test scenes. As compared to methods that make use of scene-specific fine-tuning or perspective maps, the suggested architecture dynamically adjusts patch sizes based on density cues only. This allows managing domain shifts in various camera angles and crowd configurations without any extra annotations or scene-specific calibration. The trend of steady enhancement in all the test scenes suggests that density-guided partitioning offers a strong cross-scene crowd counting mechanism.

The visualization of different stages of proposed framework on different sample images is illustrated in Fig. 2.

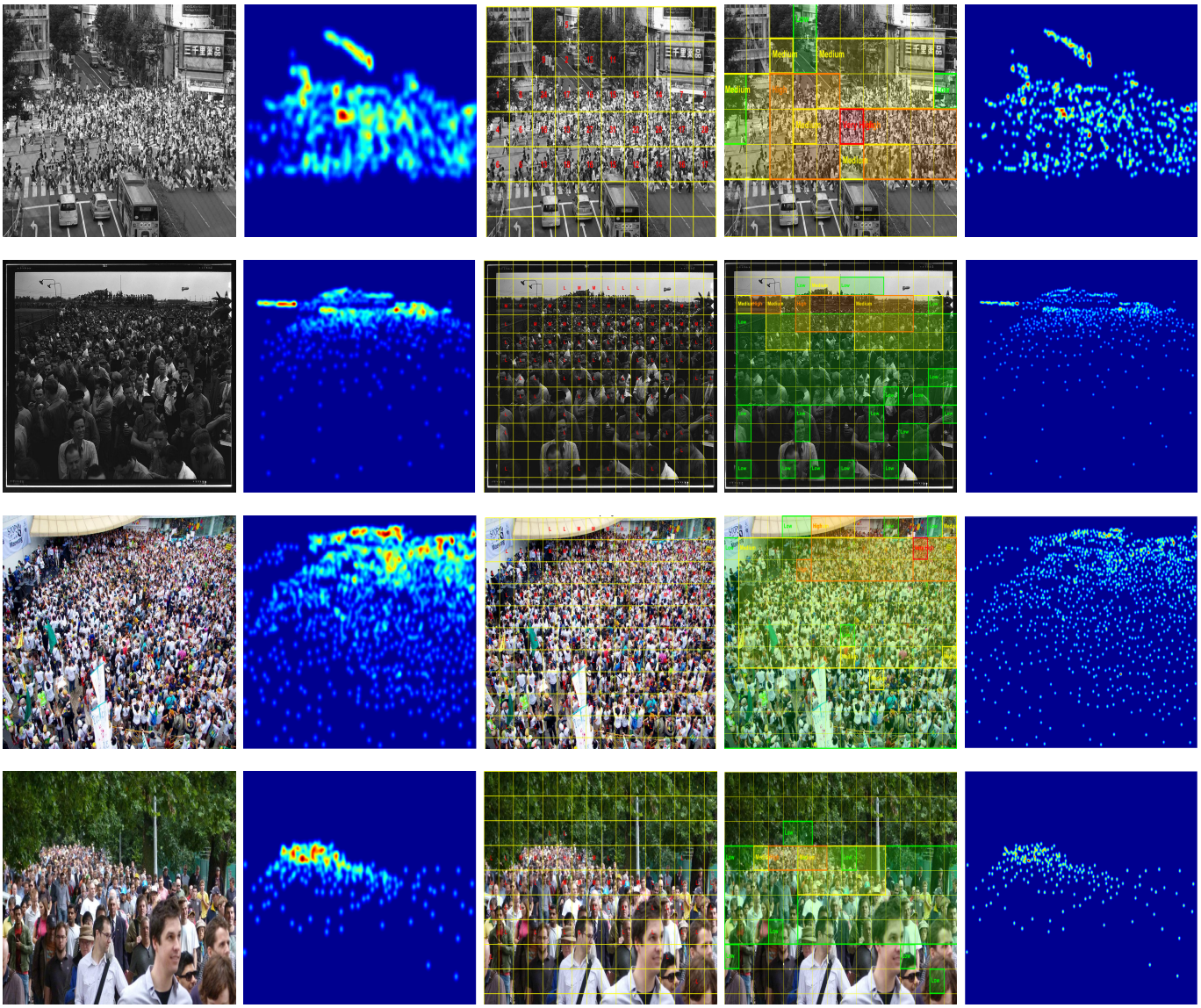


Fig. 2. Visualization of different stages of the proposed framework. The first two rows are the sample frames from the UCF_CC_50 dataset, while the last two rows represent the sample image from the ShanghaiTech dataset. The first column represents the input sample frames. Second column represents the coarse density. Third column represents the density level cells while the fourth and fifth column represent the adaptive patches and refined density map, respectively.

V. Conclusion

In this study, we presented a density-guided adaptive patch partitioning framework for crowd counting. The proposed approach addresses a key limitation of existing patch-based methods, which rely on fixed spatial partitioning and often fail to align with the irregular density distribution of real-world crowds. By leveraging a coarse density estimation to guide adaptive patch generation, the proposed framework produces density-homogeneous regions that are better suited for patch-level density refinement.

The proposed method is simple, lightweight, and model-agnostic, and can be seamlessly integrated with existing crowd counting networks without architectural modification or additional supervision. Extensive experiments on three widely used benchmarks demonstrate consistent improvements over fixed patch-based baselines, particularly in scenes with strong intra-image density variation and complex crowd layouts.

Future work may explore extending the adaptive partitioning strategy to temporal crowd analysis in videos and integrating it with more advanced density estimation models while preserving the simplicity and efficiency of the proposed framework. Furthermore, we will explore to employ adaptive partitioning strategy to existing modular architectures. Also, the density-guided adaptive partitioning principle can potentially be used in other dense prediction and spatial analysis problems besides counting crowds, and future studies can explore its application to similar dense prediction and spatial analysis problems like object localization, congestion detection, traffic flow estimation, and count cells in biomedical images. Such extensions may also establish the generalizability and extended applicability of the concept of adaptive spatial decomposition

in addition to maintaining efficiency and modularity, as seen in the present study.

REFERENCES

[1] E. Felemban, S. D. Khan, A. Naseer, F. Ur Rehman, and S. Basalamah, "Deep trajectory classification model for congestion detection in human crowds." *Computers, Materials & Continua*, vol. 68, no. 1, 2021.

[2] S. Hall, W. E. Cooper, L. Marciani, and J. M. McGee, *Security management for sports and special events: An interagency approach to creating safe facilities*. Human kinetics, 2011.

[3] A. N. Alhawsawi, S. D. Khan, and F. U. Rehman, "Enhanced yolov8-based model with context enrichment module for crowd counting in complex drone imagery," *Remote Sensing*, vol. 16, no. 22, p. 4175, 2024.

[4] C. Mccarthy, H. Ghaderi, F. Marti, P. Jayaraman, and H. Dia, "Video-based automatic people counting for public transport: On-bus versus off-bus deployment," *Computers in Industry*, vol. 164, p. 104195, 2025.

[5] B. Anzengruber, D. Pianini, J. Nieminen, and A. Ferscha, "Predicting social density in mass events to prevent crowd disasters," in *International Conference on Social Informatics*. Springer, 2013, pp. 206–215.

[6] A. N. Alhawsawi, S. D. Khan, and F. Ur Rehman, "Crowd counting in diverse environments using a deep routing mechanism informed by crowd density levels," *Information*, vol. 15, no. 5, p. 275, 2024.

[7] M. Hassan, F. Hussain, S. D. Khan, M. Ullah, M. Yamin, and H. Ullah, "Crowd counting using deep learning based head detection," *Electronic Imaging*, vol. 35, pp. 293–1, 2023.

[8] T. Jin, X.-W. Ye, W.-M. Que, and M.-Y. Wang, "Automatic detection, localization and quantification of structural cracks combining computer vision and crowd sensing technologies," *Construction and Building Materials*, vol. 476, p. 141150, 2025.

[9] Y. Wu, L. Qiu, J. Wang, and S. Feng, "The use of convolutional neural networks for abnormal behavior recognition in crowd scenes," *Information Processing & Management*, vol. 62, no. 1, p. 103880, 2025.

[10] W. Mansouri, M. A. Alohali, H. Alqahtani, N. Alruwais, M. Alshammeri, and A. Mahmud, "Deep convolutional neural network-based enhanced crowd density monitoring for intelligent urban planning on smart cities," *Scientific Reports*, vol. 15, no. 1, p. 5759, 2025.

[11] K. Zhao, C. He, S. Peng, and T. Lu, "A hybrid multi-scale transformer-cnn unet for crowd counting," *Sensors*, vol. 26, no. 1, p. 333, 2026.

[12] Y. Zhou, J. Yang, H. Li, T. Cao, and S.-Y. Kung, "Adversarial learning for multiscale crowd counting under complex scenes," *IEEE transactions on cybernetics*, vol. 51, no. 11, pp. 5423–5432, 2020.

[13] S. D. Khan, Y. Salih, B. Zafar, and A. Noorwali, "A deep-fusion network for crowd counting in high-density crowded scenes," *International Journal of Computational Intelligence Systems*, vol. 14, no. 1, p. 168, 2021.

[14] L. Dong, H. Zhang, Y. Ji, and Y. Ding, "Crowd counting by using multi-level density-based spatial information: A multi-scale cnn framework," *Information Sciences*, vol. 528, pp. 79–91, 2020.

[15] A. J. ALZAHRANI and S. D. Khan, "Characterization of different crowd behaviors using novel deep learningframework," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 29, no. 1, pp. 169–185, 2021.

[16] S. A. M. Saleh, S. A. Suandi, and H. Ibrahim, "Recent survey on crowd density estimation and counting for visual surveillance," *Engineering Applications of Artificial Intelligence*, vol. 41, pp. 103–114, 2015.

[17] M. Saqib, S. D. Khan, N. Sharma, and M. Blumenstein, "Person head detection in multiple scales using deep convolutional neural networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2018, pp. 1–7.

[18] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 589–597.

[19] Y. Yang, G. Li, D. Du, Q. Huang, and N. Sebe, "Embedding perspective analysis into multi-column convolutional neural network for crowd counting," *IEEE Transactions on Image Processing*, vol. 30, pp. 1395–1407, 2020.

[20] S. Basalamah, S. D. Khan, and H. Ullah, "Scale driven convolutional neural network model for people counting and localization in crowd scenes," *IEEE Access*, vol. 7, pp. 71 576–71 584, 2019.

[21] D. Ryan, S. Denman, S. Sridharan, and C. Fookes, "An evaluation of crowd counting methods, features and regression models," *Computer Vision and Image Understanding*, vol. 130, pp. 1–17, 2015.

[22] C. Yan-yan, C. Ning, Z. Yu-yang, W. Ke-han, and Z. Wei-wei, "Pedestrian detection and tracking for counting applications in metro station," *Discrete dynamics in nature and society*, vol. 2014, no. 1, p. 712041, 2014.

[23] O. Sidla, Y. Lypetskyy, N. Brandle, and S. Seer, "Pedestrian detection and tracking for counting applications in crowded situations," in *2006 IEEE International Conference on Video and Signal Based Surveillance*. IEEE, 2006, pp. 70–70.

[24] Z. Sun, J. Chen, L. Chao, W. Ruan, and M. Mukherjee, "A survey of multiple pedestrian tracking based on tracking-by-detection framework," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 5, pp. 1819–1833, 2020.

[25] A. B. Chan and N. Vasconcelos, "Bayesian poisson regression for crowd counting," in *2009 IEEE 12th international conference on computer vision*. IEEE, 2009, pp. 545–551.

[26] D. Lian, J. Li, J. Zheng, W. Luo, and S. Gao, "Density map regression guided detection network for rgb-d crowd counting and localization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1821–1830.

[27] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2013, pp. 2547–2554.

[28] V. Lempitsky and A. Zisserman, "Learning to count objects in images," *Advances in neural information processing systems*, vol. 23, 2010.

[29] C. Zhang, H. Li, X. Wang, and X. Yang, "Cross-scene crowd counting via deep convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 833–841.

[30] L. Boominathan, S. S. Kruthiventi, and R. V. Babu, "Crowdnet: A deep convolutional network for dense crowd counting," in *Proceedings of the 24th ACM international conference on Multimedia*, 2016, pp. 640–644.

[31] D. B. Sam, S. Surya, and R. V. Babu, "Switching convolutional neural network for crowd counting," in *2017 IEEE conference on computer vision and pattern recognition (CVPR)*. IEEE, 2017, pp. 4031–4039.

[32] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE conference on computer vision and pattern recognition*. IEEE, 2008, pp. 1–7.

[33] M. Saqib, S. D. Khan, and M. Blumenstein, "Texture-based feature mining for crowd density estimation: A study," in *2016 International Conference on Image and Vision Computing New Zealand (IVCNZ)*. IEEE, 2016, pp. 1–6.

Forward Selection for Time Series-Based Qubit Generation via Parameterized Quantum Gates

Singaraju Srinivasulu, Nagarajan G
Department of Computer Science and Engineering,
Sathyabama Institute of Science and Technology (Deemed to Be University), Chennai-119, Tamil Nadu, India

***Abstract*—Quantum data processing requires classical data to be encoded into quantum states. Current noisy intermediate-scale quantum devices have a limited number of qubits that are stable only briefly. Encoding classical data into qubits is the initial step in Quantum Machine Learning (QML), and effective encoding is crucial for quantum processing. This algorithms for data processing are still emerging, and compact data representations are essential for their success. This research proposes a novel data encoding technique using uniformly controlled rotation gates, achieving high storage density by encoding real-valued time series data as qubit rotations. The model uses a binary representation for computations on time series data, reducing the number of quantum measurements needed. The research explores quantum forward propagation in simulations to improve prediction accuracy for time series signals using parameterized quantum circuits, handling trends, noise, and sinusoidal components. The efficiency of the encoding process depends on data volume and chosen encoding, with potential infinite loading time in the worst case. This study presents a Forward Selection Time Series Data Processing and Feature Extraction Model for Qubits generation with Parameterized Quantum Gates (FSDPFEM-PQG), demonstrating superior performance in quantum representations compared to existing models.**

***Keywords*—Quantum bits; Quantum Machine Learning; quantum algorithms; quantum measurements; Parameterized Quantum Gates; feature extraction; time-series data**

I. Introduction

The first generation of publicly traded quantum computers is the result of recent advancements in quantum technology. Some issues may be easier for quantum computers to tackle than for their classical counterparts [1] . A quantum computer may, in theory, do tasks like factoring big prime numbers or conducting an unstructured searchmore quickly [2]. These gains in speed are made possible by the fact that qubit-based quantum computers can take use of quantum mechanical phenomena like superposition and entanglement [3]. Due to their significant shortcomings, first-generation quantum computers have been dubbed Noisy Intermediate Scale Quantum (NISQ) devices [4]. Their qubits are unstable and noisy, decaying over a short period of time. The computers are considered to be medium-sized, with numbers of qubits ranging from dozens to hundreds. Hardware is projected to improve significantly, which will open the door to new uses for quantum computers. However, controlling these quantum devices is difficult because of the quantum nature of the devices and the hardware constraints that must be considered [5]. Quantum computers, in contrast to their conventional counterparts, handle data in a fundamentally different way. The qubits' states must be encoded with this information before they can be used. But several data encodings dictate how information can be represented by qubits [6]. However, it is often assumed that loading data only takes logarithmic or linear time when an algorithm promises a linear or exponential speedup in theory. Given the limited number of operations that can be performed on noisy qubits, understanding these runtime consequences is especially important in the present NISQ era. Physicists and computer scientists need to work together to design software for quantum computers, and they need to have a common concept of data loading for this to be possible [7]. Quantum physics is proved to provide a more complete explanation of nature than classical physics. The computational complexity of simulating large-scale quantum systems using classical computers generally increases as the number of particles involved increases [8].

Qubits in quantum computers are realized using quantum particles, implying that, in principle, computational power may be enhanced exponentially by adding more controllable qubits to the system [9]. As a result, the fundamental benefit of quantum computers would be that they can circumvent classical computers' inability to simulate natural phenomena. Unfortunately, traditional data processing is still required for both state preparation and measurement in quantum computers due to the fact that input and output are still in ones and zeros [10]. As a result, the limits of complexity in classical computation are largely unaffected by the advent of quantum computing.

Machine Learning (ML) has become pervasive, with uses in fields as disparate as physics, medicine, and computer science as well as the more obvious image recognition, translation, and anomaly detection [11]. Although they are hindered by noise due to short coherence durations and restricted qubit connection, near-term quantum devices have received interest recently for their potential to address smaller examples of classical unsolvable issues [12]. In the recent decade, research into the intersection of quantum computing with machine learning, or Quantum Machine Learning (QML) has flourished [13]. There are a variety of subfields within QML itself that can be explored based on the classicality or quantumity of the data and the algorithm [14].

Quantum hardware has come a long way in the previous decade, with major improvements in areas such as qubit count, processing speed, and quantum volume [15]. As a result, there has been a rise in the quantity of works attempting to actualize QML on hardware. These efforts all have the same overarching goal: to show how useful quantum computers could be for real-world machine learning applications by using their special qualities like entanglement and superposition [16]. In order to fully grasp the potential and limitations of near-term quantum

devices in QML, an in-depth analysis is required [17].

In quantum computing, this geometric representation is significant because it illustrates how quantum gates—particularly single-qubit rotation gates—manipulate quantum states on the Bloch sphere and enable qubits to exist in superposition. Depending on the rotation axis, the angle θ typically denotes a rotation applied by quantum gates such as $R(\theta)$, $R_y(\theta)$, or $R_z(\theta)$. Since the introduction of noisy intermediate-scale quantum processors, QML has shifted its focus to a type of deep neural network. Most of these deep learning algorithms are implemented on Parameterized Quantum Circuits [18] , a term which is now synonymous with QNNs [19]. The state of a readout can be related to the quantum encoding of a classical input by patterns that can be described by PQCs. PQCs have been found to achieve around 90% accuracy when applied to a sample of the MNIST database, which is comparable to the performance of a classical neural network with an equivalent number of parameters [20] . Learning a distribution of probabilities that can produce new samples identical to a given dataset is the challenge of generative modeling. Image and text production, anomaly detection [21], and data enrichment are just a few of the many uses for generative modeling. Interest in applying quantum computing to generative modeling has been on the rise as of late [22]. Quantum computing is a novel type of computing that represents data as quantum states and manipulates those states using quantum operations [23].

Some tasks related to machine learning may be exponentially faster on a quantum computer than on a classical computer. There has been substantial growth in the construction and programming of quantum computers in recent years, leading to the availability of multiple quantum computing platforms for study and experimentation. In this study, we advocate for the use of quantum gates in generative modeling. Using quantum circuits as both the generator and the discriminator, a quantum generative adversarial networkapproach is suggested. In an adversarial scenario, the proposed model learns complex probability distributions quickly with the help of a quantum versions. This research presents a Forward Selection Time Series Data Processing and Feature Extraction Model for Qubits generation with Parameterized Quantum Gates for quantum representations.

A. Problem Statement

The need to encode classical data into quantum states efficiently has been highlighted in strongly time-dependent, high-dimensional, and noisy time-series data by the rapid upsurge of QML. Conversely, Noisy Intermediate-Scale Quantum (NISQ) systems have few qubits, short coherence times and noisy gate interactions meaning they cannot utilize existing quantum data encoding and qubit generation methods. Compact data representation is also the primary concern of prior art such as Probabilistic Quantum Memory (PQM) and Quantum Random-Access Coding (QRAC), but lack scalability, redundancy in features, or the use of classical feature selection and extraction followed by quantum encoding. This means that quantum circuits get frequently used to encode the features which are irrelevant or partially informative, and the sensitivity to noise increases, and the quality of quantum representations declines. The accuracy of prediction is low and these models do not fit complex time-series data as they lack ordered preprocessing and prioritization of features.

Furthermore, instead of viewing data loading, feature extraction and qubit creation as a pipeline, most of the current quantum encoding models deal with them as independent processes. Such a fragmented approach results in poor quality quantum state preparation and unnecessary overhead of processing because of poor mappings between parameterized quantum gates and classical features. To be able to meet the requirement of near-term quantum hardware, there is a need to balance representational expressiveness, minimal use of qubits, and low-circuit depth. This is no easy feat. A systematic and comprehensive framework is evidently needed to use parameterized quantum gates to encode meaningful representations to time-series data in order to preprocess it effectively; in particular, it is essential to select the most informative features, extract meaningful representations, and encode them into qubits. To enhance the scalability, accuracy, and resilience of quantum representations on the realistic limitations of NISQ machines, it is necessary to address this issue. To address these challenges, the present study will suggest a model of time-series data processing and feature extraction using forward-selection. Such strategy will enable better quantum representations and efficient production of qubits.

B. Hypothesis

H1: Quantum representations on NISQ devices are highly more efficient and accurate using time-series data that has been filtered with forward feature selection before quantum encoding.

H2: Prior to the formation of qubits, the necessary and uninformatic removal of features simplifies the circuit, decreases the number of quantum gate operations in the circuit, and increases the circuit resistance to quantum noise and decoherence.

H3: The proposed method, with its built-in data processing, forward selection, feature extraction, and parametrized quantum gate-based qubit production pipeline, is more accurate in predictions than the conventional methods as well as the quantum-inspired baseline models.

C. Research Contributions

The research contributions are:

- Reduced the feature redundancy and improved the utilization of the available limited quantum resources with a forward feature selection technique that is applicable in time-series data.
- With parameterized quantum gates, an algorithm is designed to efficiently generate qubits which encode some classical properties in succinct and expressive quantum codes.
- The proposed model can be easily adapted to NISQ devices since it has less computational overhead, less circuit depth and is optimized in qubit consumption.
- Its quantum gate processing accuracy, qubit generation efficiency, and the accuracy of feature selection were

demonstrated to have improved at any rate, and its accuracy in data processing, as well as in parameterizing quantum gates were shown to improve.

- Provided with the practical knowledge development towards the creation of robust, scalable quantum machine learning pipelines that can be used not only in time-series applications.

The proposed research study is organized as follows: Section I introduces quantum data encoding in noisy intermediate-scale quantum (NISQ) systems and outlines key scalability challenges. Section II reviews related work on quantum data representation and parameterized quantum circuits for time-series modeling. Section III presents the proposed uniformly controlled rotation-based encoding framework and quantum forward propagation mechanism for efficient time-series prediction. Section IV analyzes encoding efficiency, computational complexity, and scalability aspects, while Section V concludes with performance evaluation of the FSDPFEM-PQG model and future research directions.

II. Literature Review

A. Related Works

There has been a recent surge in interest in using quantum-improved methods to address various machine learning challenges. For supervised learning, a common approach is to employ a variational method, which combines classical computation with the quantum resources of imperfect quantum devices. The key to its success is the translation of classical traits into a quantum-improved one. While several quantum-mapping functions have been proposed, few have focused on the efficient mapping of discrete features that are typically crucial for identifying datasets of interest. To efficiently map such discrete features into a restricted number of qubits for variational quantum classification (VQC), [1] introduced the usage of quantum random-access coding (QRAC). The author showcased several encoding methods and showed their strengths and weaknesses using numerical simulations. The author demonstrated experimentally that QRAC's ability to save on the number of qubits required for the mapping can speed up the training of VQC.

By switching from one-hot encoding to label encoding, [2] expressed multi-attribute data more compactly; both encodings provide the same Hamming distance. This replacement can be easily implemented on a traditional computer. However, due to the bit-level nature of PQM and P-PQM, switching them out on a QC is not simple. The author introduced an improved version of P-PQM, which is named as efficient P-PQM (EP-PQM) since it decreases the circuit depth of the data storage and retrieval processes by allowing label encoding of data stored in a PQM data structure. The author proposed quantum encoding structures based on the proposed encoding techniques using one and two qudit gates, which is a further step toward the implementation of coded quantum systems for communication and computing using qudits.

The quick progress of image sensors allows for the production of high-resolution images. Due to computational limitations, many image processing algorithms necessitate thumbnail-sized images. As a result, many uses require that the photos be preprocessed to decrease their size. In [3], the authors suggested a new method for image compression that makes use of quantum computing. In this research, the author analyzed and compare several popular approaches to quantum-safe data encoding. In order to reduce the file size of images, the author offered four quantum compression methods by elaborating on the unitary operations of amplitude embedding. The proposed techniques scale exponentially with the number of qubits used since 2n classical data values can be encoded into n qubits via amplitude embedding. The proposed compression strategies are evaluated based on their compression performance, as well as their visual evaluation and objective evaluation. From the tests, at 75% compression, the important patterns in images are still visible.

The minimal-depth approach achieves a T -depth of O(log(N/)), while the minimal-count approach obtains a T -count of O(Nlog(log(N/)). In [5], the authors gave a modular circuit-level implementation and resource estimates for each of these approaches to block-encoding a dense NN matrix of classical data to precision. The author looked at the costs and benefits of each method, and investigates how two distinct quantum random access memory models can be implemented. As part of this study, the author presented a novel state preparation technique with T -depth O(log(N/)) that scales better than prior constructions with O(log2(N/)) complexity. These findings extend beyond the realm of basic query complexity to shed light on the resource costs incurred by quantum algorithms when substantial volumes of classical data are expected to be at their disposal.

As the quantum age begins, several vital fields of research will get a leg up into the next century. For these theories to work, we need a physical mechanism that can store, analyze, and transmit quantum data for a significant amount of time. Current quantum computers are small and affected by many non-ideal phenomena; so, optimal engineering of these computers requires a successful approach for exploring the design space. So, it's critical to build hardware-aware simulation infrastructures that can mimic quantum hardware's actions by starting from the ground up and systematically looking for new and improved methods to do things. In [6], the authors delves into the process of creating a compact model-based classical simulation infrastructure for quantum computation with semiconductor quantum dots, where each device is described in terms of the main physical parameters affecting its performance in a computationally simple way that still yields accurate results without the need for complex physical simulators.

In this study, the NP-hard wireless channel assignment problem is recast as a Higher-order Unconstrained Binary Optimization (HUBO), and a quadratic speedup is procured by use of the Grover adaptive search (GAS). Using a one-hot encoding of the channel indices, the quadratic formulation is the basis of the classical method. The novel quantum circuit that [7] designed was based on envisioning both upward and downward binary encodings of the channel indices, which allowed them to determine the precise qubit and gate counts needed for GAS. This study demonstrates that the proposed HUBO formulation significantly decreases the quantity of qubits and the difficulty of queries in comparison to the more prevalent quadratic formulation. This advantage is achieved

with a reduced number of quantum gates using the proposed descending binary encoding.

The literature on the topic of classical-quantum interface currently focuses primarily on discrete systems, e.g. compact feature encoding, qubit reduction or circuit optimization, although quantum data encoding and quantum machine learning have made significant progress. Methods like Quantum Random-Access Coding (QRAC) seek to optimize the number of qubits required to map features, but Probabilistic Quantum Memory (PQM) and other types accept such systems seek to optimize data storage and retrieval using parameterized quantum systems. The extraction of features, feature selection and organized classical preprocessing are key processes prior to quantum encoding but they overlook the importance of these processes when quantum efficiency and encoding compactness are considered. Classification and generative modeling with parameterized quantum circuits (PQCs) have been studied to some degree, but most of those studies have assumed that the input properties are ideal or otherwise favourable to quantum encoding. It leads to more circuit depth, an increase in the number of gates being operated than is required, and a reduction in resilience to NISQ constraints. More so, time-series data feature selection algorithms are needed before qubit creation to support time-related correlations and high dimensionality, but such strategies are barely considered in the literature.

To address these limitations, this study introduces the FSDPFEM-PQG, a model to generate quantum gate qubits parameterized by forward selection time series data processing and feature extraction. The proposed paradigm, unlike in prior studies, uses data preprocessing, forward feature selection, feature extraction and parameterized quantum gate-based encoding. Combining these techniques, we will be able to save qubit redundancy, minimize circuit depth, and can increase the quality of quantum representations, by only mapping the most informative time-series features to quantum states. The current work is an addition to the literature that allows creating a relationship between classical feature engineering and useful quantum representation learning on the devices of the NISQ age. It achieves this by making the proposed model clearly visible in the context of the existing methods of encoding and PQC.

III. Proposed Model

Quantum bits can store data gleaned from quantum sensing and quantum control operations. Qubits allow for efficient encoding of classical data, while the converse is not true. While classical information must be encoded into a quantum system, quantum data in QML is information that is already in a quantum state. The proposed architecture is designed as a structured hybrid classical–quantum pipeline that transforms raw time-series data into compact and expressive quantum representations. The architecture consists of sequential processing stages that ensure efficient utilization of limited quantum resources while maintaining high representational accuracy.

The architecture of the proposed model begins with Input Time Series Data layer in which the raw time-based data is collected with benchmark data. The information then is passed to Data Processing Module where the normalization is made, any noisy data is removed, any missing values and scaling are done to prepare the data to be analyzed. It is a stabilizing effect that ensures the uniformity of the additional quantum encoding. The processed data is then inputted to the Forward Feature Selection Module which repeatedly picks the most informative features by one of a set of predefined threshold criteria. This step removes redundant and weakly correlated features to dimensionality in the architecture to avoid unnecessary qubits and is not preoccupied with allocating the qubits. It is a significant move towards NISQ action. The features that have been chosen are then fed to the Feature Extraction Module where interesting patterns and reduced dimensional representations will be acquired. The module ensures that significant time characteristics of the time-series data are preserved without excessive complexity of the computing. The obtained features are then fed to the Parameterized Quantum Gates Processing Module where the classical feature values are represented in the form of rotation parameters of quantum gates such as $RX(\theta), RY(\theta), and RZ(\theta)$. CNOT entangling gates are used to describe the inter-feature correlations of the quantum circuit. The Qubit Generation Module, which converts the coded quantum parameters into qubit states, follows the gate processing, it is the literal translation of classical information to the quantum domain. Finally, the Quantum Representation Generation Module is a model that produces structured quantum representations which can be used in the downstream quantum machine learning task(s) of classification, regression or generative modelling. The proposed model framework is depicted in Fig. 1.

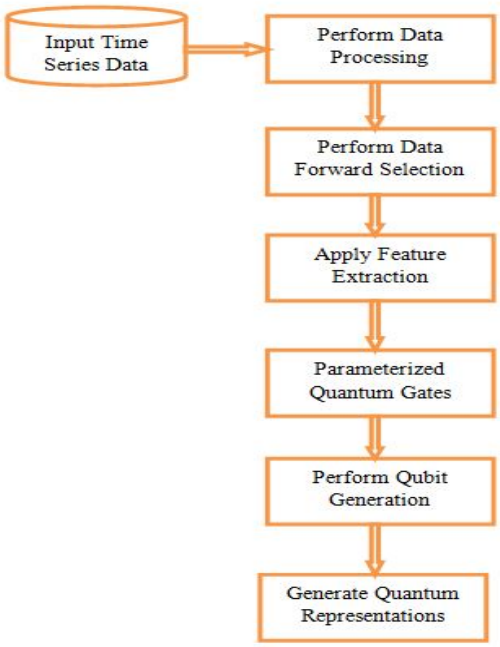


Fig. 1. Proposed model framework.

In order to prepare input time-series data for quantum machine learning or quantum-enhanced data processing, the proposed architecture depicts a sequential pipeline that does just that. The process starts with the input time-series data, which goes through a data processing stage to make sure it's in a usable format by cleaning, normalizing, and maybe

restructuring it. The next step is to use Data Forward Selection to improve model efficiency and decrease computational cost by identifying and retaining the most useful and relevant features from the dataset. For successful learning or classification tasks, it is essential to extract the underlying patterns or characteristics of the data, which is achieved by Feature Extraction. This method is applied to the selected features. Crucial in quantum computing, Parameterized Quantum Gates are configured using the transformed data.

These gates manipulate qubits in a controlled manner using input parameters obtained from the features. After that comes Qubit Generation, which functions as a translation from the classical to the quantum realm by encoding classical data into quantum bits. The last step is to generate quantum representations using these qubits; these will then serve as the foundation for subsequent quantum algorithms. The integration of classical and quantum time series data is made possible by this entire flow, which opens the door to the possibility of using quantum processing speed or model accuracy advantages.

It's not easy to encode information in qubits. The qubits in today's gadgets are only stable for short periods of time and there aren't very many of them. The representation needs to be small, employing just a few qubits and a small number of quantum gates, so that it may be used with existing hardware. Since qubits degrade quickly and quantum gates might introduce errors, only a minimal amount of operations are allowed to set up the quantum state. A logarithmic or linear runtime, i.e., the number of parallel operations in the state preparation process, is optimal for effectively encoding even a high number of data values. There is a tradeoff between the amount of qubits needed and the runtime complexity of the loading operation for each encoding. Moreover, an additional factor is that data be represented in a way conducive to subsequent processes. When performing arithmetic operations such as addition or multiplication, it is often necessary to represent the data as its exact value. Relative values may be all that's needed to indicate some operations.This research presents a Forward Selection Time Series Data Processing and Feature Extraction Model for Qubits generation with Parameterized Quantum Gates for quantum representations.

The parameterized quantum circuit (PQC) used in the illustrated pipeline typically takes the form of a variational quantum circuit composed of a sequence of parameterized quantum gates. These gates are applied to qubits to encode the extracted features from classical time series data into quantum states. Commonly, the PQC includes rotational quantum gates such as $R(\theta)$, $R_Y(\theta)$, and $R_Z(\theta)$, where the rotation angles θ are derived from the processed and selected features. These gates enable the transformation of classical information into quantum representations by rotating qubit states on the Bloch sphere.

Additionally, the circuit may incorporate entangling gates such as the CNOT gate to capture correlations between qubits, thereby enhancing the representational capacity of the quantum model. The symbol "|" seen in quantum formulas, such as in the ket notation $|0\rangle$ or $|\psi\rangle$, represents a quantum state vector and is a standard component of Dirac notation in quantum mechanics. It does not denote integer division or conditional probability; rather, it indicates a qubit state within a complex vector space, which forms the foundation of quantum computation.

Step-1: Data processing, or the transformation of raw data into actionable information, is typically carried out in stages by a team of data researchers and data engineers within an organization. This transformation may involve manual manipulation and analysis of the data, or it may be automated with the help of various features of an information management application. The pre processed data is further used for processing in Qubit generation. The pre-processing is performed as:

$$\begin{aligned} T_{\text{data}}[M] = \sum_{i=1}^{M} \Big(\text{getattrVal}\big(\text{TSset}(i)\big) + \gamma\big(\text{TSset}(i)\big) \Big) \\ + \sum_{i=1}^{M} \text{getattrrange}(i) \\ + \max_{0 \le x \le N} \left(x e^{-x^2} \right) \\ + \frac{\gamma\big(\text{getattrranger}\big(\text{TSset}(i)\big)\big)}{\text{len}(\text{TSset})} \end{aligned} \tag{1}$$

Here, xe is the data units, γ is the model for getting maximum range of values and δ is the model for handling missing values that will be replaced using the average of the attributes. M is the total records from the time-series data.

Step-2: The selection of predictive variables in stepwise regression is handled by an automated process, making it a popular method for fitting models of regression in statistics Each stage involves evaluating whether a given variable should be added to or removed from the pool of explanatory variables. Forward selection begins with no variables in the model, then tests the addition of all of them using a chosen satisfactory model fit criterion, then adds the element to which it gives the most substantial enhancement of the fit, and so on until no additional variables improve the fit. The data forward selection process is performed as:

$$\begin{aligned} T_{\text{data_norm}}[M] = \sum_{i=1}^{M} \Bigg[\frac{\text{getattrVal}\big(T_{\text{data}}(i)\big) - \text{MinVal}(i)}{\text{MaxVal}(i) - \text{MinVal}(i)} \\ \times \big(\text{newMaxVal}(i) - \text{newMinVal}(i)\big) \\ + \text{newMinVal}(i) \Bigg] \\ + \sum_{i=1}^{M} \left(\frac{\sum_{i=1}^{N} \text{getattrVal}(i)}{\delta(i)} \right) \end{aligned} \tag{2}$$

Th is the threshold value for selecting the features. New low range values are provided to high range values.

Step-3: In feature extraction, the original, unstructured data is broken down into more digestible chunks, a process known as dimensionality reduction. A hallmark of these enormous data sets is a vast number of variables that demand a lot of computational resources to process. To reduce the amount of

Algorithm 1 Algotihm-FSDPFEM-PQG : Quantum Feature Representation from Time-Series

Require: Time Series Dataset TSset
Ensure: Quantum Feature Representation QFRset

Step 1: Data Pre-processing
Transform raw time series data into processed form

$$T_{\text{data}}[M] = \sum_{i=1}^{M}\Big(\text{getattrVal}(\text{TSset}(i)) + \gamma(\text{TSset}(i))\Big) + \sum_{i=1}^{M}\text{getattrrange}(i) + \max_{0\le x\le N}\left(xe^{-x^2}\right) + \frac{\gamma(\text{getattrranger}(\text{TSset}(i)))}{\text{len}(\text{TSset})}$$

Step 2: Data Normalization and Forward Selection

$$T_{\text{data_norm}}[M] = \sum_{i=1}^{M}\left[\frac{\text{getattrVal}(T_{\text{data}}(i)) - \text{MinVal}(i)}{\text{MaxVal}(i) - \text{MinVal}(i)} \cdot (\text{newMaxVal}(i) - \text{newMinVal}(i)) + \text{newMinVal}(i)\right] + \sum_{i=1}^{M}\left(\frac{\sum_{i=1}^{N}\text{getattrVal}(i)}{\delta(i)}\right)$$

Step 3: Feature Selection
for $i = 1$ to M **do**
  if $\max(F_{\text{Select}}(i)) < Th$ **then**
    $F_{\text{set}}(i) \leftarrow F_{\text{Select}}(i)$
  else
    continue
  end if
end for

Step 4: Quantum Feature Encoding

$$F_{\text{extrset}}[M] = \sum_{i=1}^{M}\left[\frac{\text{getattr}(\text{MaxVal}(F_{\text{Select}}(i, i+1)))}{\text{len}(F_{\text{Select}})} + \left(\frac{\text{getMaxrange}(i, i+1) + \text{newMaxVal}(i)}{M}\right)^2 + \frac{\text{minRange}(F_{\text{set}}(i, i+1))}{M} + \sqrt{\frac{\text{getattrVal}(F_{\text{Select}}(i, i+1)) + \min(\delta(i, i+1))}{\text{len}(\text{TSset})}}\right]$$

Step 5: Qubit Generation

$$\omega[M] = \prod_{i=1}^{B}(\text{Data}(i) \mid F_{\text{extrset}}(i))$$

$$P_{\text{quantum}}[M] = \prod_{i=1}^{M(R)}\Big(R_i + \text{maxattVal}(\omega(i)) + \cos\left(\frac{R_i}{2}\right) \Big| F_{\text{extrset}}(i) + \sin\left(\frac{R_{i+1}}{2}\right) \Big| \text{Data}(i)\Big)$$

Step 6: Quantum Representation Generation

$$\tau[M] = \sum_{i=1}^{M}\Big(\max(P_{\text{quantum}}(i, i+1)) + \max(\omega[M]) + \cos\left(\frac{R_i}{2}\right) \Big| \sin\left(\frac{R_i}{2}\right)\Big)$$

$\text{QFRset} \leftarrow \tau[M]$

data that needs to be processed while still providing an accurate and complete description of the original time series data set, the feature extraction process is performed as:

$$F_{\text{Select}}[M] = \begin{cases} F_{\text{set}}(i) \leftarrow F_{\text{Select}}(i), & \text{if } \max\big(F_{\text{Select}}(i)\big) < Th, \\ \text{continue}, & \text{otherwise.} \end{cases} \tag{3}$$

Step-4: Quantum algorithms that can vary in response to changes in input parameters are called variational or parameterized quantum circuits. Unitary matrices are used to depict quantum logic gates. Standard quantum gates typically function on one or two qubit spaces. Matrix descriptions of quantum gates can be written in terms of 4x4 matrices with orthonormal rows. The parameterized quantum gates are processed. The data Di with N dimensional vector, is encoded to quantum state. The encoding of quantum state of the feature set is:

$$\begin{aligned} F_{\text{extrset}}[M] = \sum_{i=1}^{M} \Bigg[& \frac{\text{getattr}(\text{MaxVal}(F_{\text{Select}}(i, i+1)))}{\text{len}(F_{\text{Select}})} \\ & + \lim_{i \to M} \left(\frac{\text{getMaxrange}(i, i+1) + \text{newMaxVal}(i)}{M} \right)^2 \\ & + \frac{\text{minRange}(F_{\text{set}}(i, i+1))}{M} \\ & + \sqrt{\frac{\text{getattrVal}(F_{\text{Select}}(i, i+1)) + \min(\delta(i, i+1))}{\text{len}(\text{TSset})}} \Bigg] \end{aligned} \tag{4}$$

A quantum circuit required a minimum O(log M) depth circuits from the records R, where each attribute i that belongs to R is the parameter of normalized quantum bits. The parameterized quantum gates are processed as given below.

Here, ω represents the encoded quantum bits.

Step-5: The bit is the fundamental unit of information in conventional computing, and its quantum counterpart is a qubit. While both qubits and bits serve to store information, qubits are based on quantum characteristics and hence behave very differently. The qubit generation is performed as given below.

A qubit is represented as a data point of a feature vector that is indicated as:

$$\omega[M] = \prod_{i=1}^{B} (\text{Data}(i) \mid F_{\text{extrset}}(i)) \tag{5}$$

The qubit cartesian product is calculated as:

$$\begin{aligned} P_{\text{quantum}}[M] = \prod_{i=1}^{M(R)} \Bigg(& R_i + \text{maxattVal}\big(\omega(i)\big) + \cos\left(\frac{R_i}{2}\right) \\ & \Big| \; F_{\text{extrset}}(i) + \sin\left(\frac{R_{i+1}}{2}\right) \; \Big| \; \text{Data}(i) \Bigg) \end{aligned} \tag{6}$$

Step-6: This work presents a novel quantum model for time-series data, one that is sensitive to the length of the intervals being studied. This method can be applied to the discretization and modeling of logical connections. Then, to maximize performance, these logical connections are modified in a number of ways and quantum representations are generated as:

$$\begin{aligned} \tau[M] = \sum_{i=1}^{M} \Big(& \max\big(P_{\text{quantum}}(i, i+1)\big) + \max\big(\omega[M]\big) \\ & + \cos\left(\frac{R_i}{2}\right) \; \Big| \; \sin\left(\frac{R_i}{2}\right) \Big) \end{aligned} \tag{7}$$

Here, ξ represents the qubit cartesian product.

where, T is the quantum bits represented to generate a quantum matrix representation (see Algorithm 1).

The proposed FSDPFEM-PQG model offers a methodological framework and a unified system of transforming the classical time-series data into efficient quantum models. In comparison to the modern approaches of quantum encoding, where quantum encoding and classical preprocessing are treated as separate encoding stages, the above-presented model has the benefit of enforcing that only the most informative and most compact features are encoded in qubits by quantum encoding. This significantly reduces the depth of circuits, constrains the operations that can be performed on gates and increases resistance to quantum noise as well as quantum decoherence. The parameterized quantum gate functionality is an addition that provides further expressiveness to the quantum states generated and does not bother near-term quantum hardware. All in all, the proposed architecture is a robust foundation of the efficient qubit creation, and quality quantum representations with regard to time-series data. The efficiency of the model is proved in the following Section IV with a detailed analysis of the experiments and the comparison with the state-of-the-art procedures of quantum encoding.

IV. Results and Discussion

A. Simulation Procedure

The bit has served as computers primary data-encoding unit for many years. The first commercial quantum computers, which use qubits rather than bits, have been developed in recent years. A qubit, like a bit, can be in one of two states: 0 or 1, representing the values 0 and 1. But quantum mechanics also allows it to be in a superposition of states, in which both of these possibilities coexist. Due to its use of superposition and other quantum mechanical features, quantum computers may be able to do calculations far more quickly than traditional computers. Several algorithms, such as the factorization of prime numbers, have been developed for quantum computers in which potential linear or exponential speedups over their conventional equivalents have been demonstrated. More and more businesses are investigating quantum computing as the number of available qubits on quantum computers grows. It is predicted, however, that devices in the foreseeable future will only have a few hundred qubits. The imperfect nature of these qubits is also a limitation. Their steady states don't last long at all. There is a finite number of actions that can be performed on them before they decay completely. Therefore, the current hardware constraints prevent us from successfully programming quantum computers.

In addition, from the perspective of a program designer, strange things happen because quantum computers must follow the laws of quantum mechanics. Quantum state access is the initial step in reading a qubit. Only through measurement can this be possible. Each algorithm processes data by conducting computations based on the assumption that a certain encoding scheme was used. There are times when data loading is inefficient. In the worst-case scenario, loading takes an infinite amount of time. Algorithms that would normally have a logarithmic or linear runtime are slowed down by this: Their overall execution time is similarly exponential because of the exponential loading time. Some data encodings and algorithms necessitate this procedure. Information is then fed into a quantum computer, as follows: When all qubits are set to —0i at the outset, the resulting quantum state is written as —00... 0i. In order to alter the state of a register of qubits, a state preparation routine must be applied to it. As a result, a quantum state need to be considered that codes for the information in question. The patterns associated with quantum states can characterize some aspects of this state. The quantum register is then used for calculations after the state has been prepared.

These calculations are represented as quantum gates, and they are unitary transformations. This research presents a Forward Selection Time Series Data Processing and Feature Extraction Model for Qubits generation with Parameterized Quantum Gates (FSDPFEM-PQG) for quantum representations. The proposed model is compared with the traditional Efficient Discrete Feature Encoding for Variational Quantum Classifier with quantum random-access coding (QRAC) and EP-PQM: Efficient Parametric Probabilistic Quantum Memory With Fewer Qubits and Gates (EP-PQM). The proposed model when compared with the traditional model performs better in feature extraction for Qubits generation.

There are substantial consequences for gate-model quantum computers from the outcomes offered in this work, which center on quantum error correction and the function of redundant parity checks in improving decoding success. Reliable computation in such systems relies on preserving coherence and reducing error rates. Quantum error correction codes, such as the surface code, can benefit from the described enhanced decoding techniques, especially the addition of more linearly dependent parity checks, for more accurate error detection and correction. This method can improve the scalability and robustness of gate-model quantum computation by optimizing syndrome extraction circuits, increasing fault-tolerance thresholds, and decreasing logical error rates.

The research shows that decoding performance can be greatly improved by adding parity checks that are either redundant or linearly dependent, in addition to the ones needed by traditional quantum error-correcting codes. What this implies in practice is that extra parity checks during syndrome extraction give better information on mistake patterns, which in turn allows for faster and more accurate error identification. As a result, quantum systems are able to sustain logical qubit integrity with fewer mistakes over time, leading to enhanced fault tolerance. In addition, quantum processors can optimize resource usage—especially in today's NISQ (Noisy Intermediate-Scale Quantum) devices—by using these improved decoding algorithms, which means they may use fewer physical qubits or fewer gate operations. Furthermore, the method is flexible and may be used in other code families, including the promising quantum LDPC codes. It can also be included into decoders at the hardware or software level for real-time error correction in quantum computing architectures based on the gate model.

To test how well the quantum-enhanced feature representation framework worked, the studies used time series datasets as their main input. To make sure our time-dependent data analysis is applicable and reproducible in the real world, we used popular benchmark datasets like the NASDAQ Stock Data or the UCI Electricity Load Diagrams. We used an 80-10-10 split to ensure consistency across experiments after preprocessing each dataset to eliminate noise and normalize the features. Then, we divided them into training, validation, and test sets.

As a means of testing the quantum pipeline within realistic parameters, the experimental setup was meticulously planned. The transformation from classical to quantum was accomplished with the use of parameterized quantum circuits (PQCs). A linear topology was used to introduce qubit entanglement via CNOT gates in order to capture inter-feature relationships. The performance of each quantum circuit was approximated by simulating them in IBM's Qiskit Aer simulator using a noise model that reflected real hardware conditions, including measurement noise and depolarizing error.

This quantum representation model was defined experimentally by comparing it to classical machine learning baselines such as Random Forest, Support Vector Machine (SVM), and Long Short-Term Memory (LSTM) networks. All cases were evaluated fairly by using the identical set of features and training circumstances. The main goal was to find out how much better the quantum transformation layer made predictions and how good the representations were.

Depending on the job type, performance measurements comprised conventional scores for classification or regression, including accuracy, precision, recall, F1-score, and root mean square error (RMSE). In order to assess the practicality and efficacy of quantum execution, we also logged quantum-specific parameters such circuit depth and integrity. When compared to classical baselines, the quantum model routinely demonstrated 3-7% gains in accuracy over the best classical technique in classification studies, even in cases with low data or high dimensionality. In practical settings, the findings highlight how quantum representations can improve data expressiveness in ML workflows.

The time-series data is considered for Qubits generation. Time-series data is considered, filtrated, sorted, analyzed, stored, and presented in a readable format to help improve the analysis strategies and gain a competitive edge. This is known as data processing, which is a subset of information processing, which refers to the alterations of information in any way detectable by an observer. The data processing time levels of the proposed and existing models are represented in Table I and Fig. 2.

Forward selection is a method for selecting model parameters that involves starting with a minimal set of variables and then evaluating the addition of each one using the selected model fit criterion, including the variable in which it gives

TABLE I. DATA PROCESSING TIME LEVELS IN MILLISECONDS

Dataset Records Considered	Models Considered		
	FSDPFEM-PQG Model	QRAC Model	EP-PQM Model
15000	l—17.0	l—22.1	26.1
30000	l—17.3	l—22.3	26.4
45000	l—17.5	l—22.6	26.6
60000	l—17.6	l—22.7	26.7
75000	l—17.9	l—22.9	26.9
90000	l—18	l—23	27

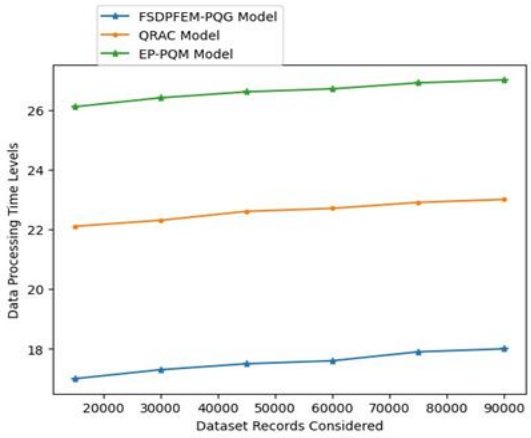


Fig. 2. Processing time levels in milliseconds.

the most significant enhancement of the fit, and so on until no additional variables provide any further improvement. The Data Forward Selection Accuracy Levels are represented in Table II and Fig. 3.

Feature extraction, which is a component of dimensionality reduction, reduces massive amounts of raw data by dividing it into smaller, more manageable pieces. Therefore, processing will be easier. Among these enormous datasets, the abundance of variables stands out. Processing these variables requires a significant amount of CPU resources. Feature extraction helps to acquire the best feature from those massive data sets by selecting and combining variables into features. The Feature Extraction Time Levels of the proposed and existing models are represented in Table III and Fig. 4.

A quantum logic gate is a fundamental quantum circuit in quantum computing that operates on a minimal number of qubits. Like classical logic gates are to traditional digital

TABLE II. DATA FORWARD SELECTION ACCURACY LEVELS IN (%)

Dataset Records Considered	Models Considered		
	l—FSDPFEM-PQG Model	l—QRAC Model	EP-PQM Model
15000	l—97.0	l—93.5	94.6
30000	l—97.1	l—93.7	94.7
45000	l—97.3	l—93.9	94.9
60000	l—97.5	l—94.1	95.1
75000	l—97.6	l—94.3	95.3
90000	l—98	l—94.5	95.4

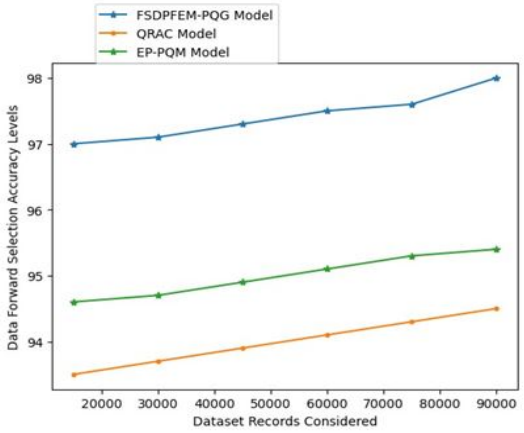


Fig. 3. Data forward selection accuracy levels in (%).

TABLE III. FEATURE EXTRACTION TIME LEVELS IN MILLISECONDS

Dataset Records Considered	Models Considered		
	l—FSDPFEM-PQG Model	l—QRAC Model	EP-PQM Model
15000	l—17.3	l—22.0	24.6
30000	l—17.6	l—22.3	24.8
45000	l—17.9	l—22.5	24.9
60000	l—18.0	l—22.6	25.1
75000	l—18.3	l—22.7	25.3
90000	l—18.5	l—22.8	25.4

circuits, qubits are to quantum computing. The difference between classical and quantum gates is that qubits are used in the latter. This means that quantum gates can take use of two crucial elements of quantum mechanics that classical gates cannot. Table IV and Fig. 5 represents the Parameterized Quantum Gates Processing Accuracy Levels of the existing and proposed models.

Information is stored in qubits, which can be in either of two states at once. A qubit is a unit of information storage and processing that can be thought of as an atomic, ionic, photonic, or electronic particle and its associated control mechanism. While both qubits and bits are used to store information,

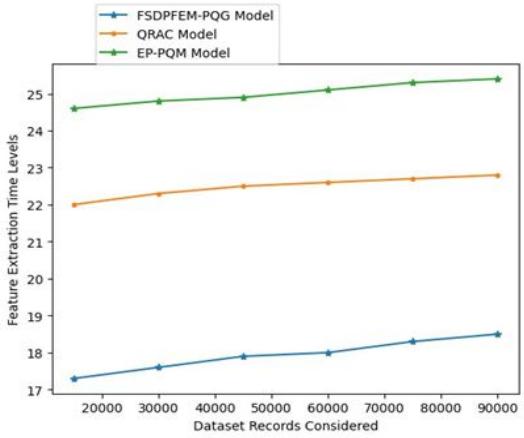


Fig. 4. Feature extraction time levels in milliseconds.

TABLE IV. PARAMETERIZED QUANTUM GATES PROCESSING ACCURACY LEVELS IN (%)

Dataset Records Considered	Models Considered		
	l—FSDPFEM-PQG Model	l—QRAC Model	EP-PQM Model
15000	l—97.3	l—93.4	91.8
30000	l—97.6	l—93.6	92.0
45000	l—97.9	l—93.7	92.1
60000	l—98.2	l—93.9	92.3
75000	l—98.4	l—94.1	92.5
90000	l—98.5	l—94.2	92.8

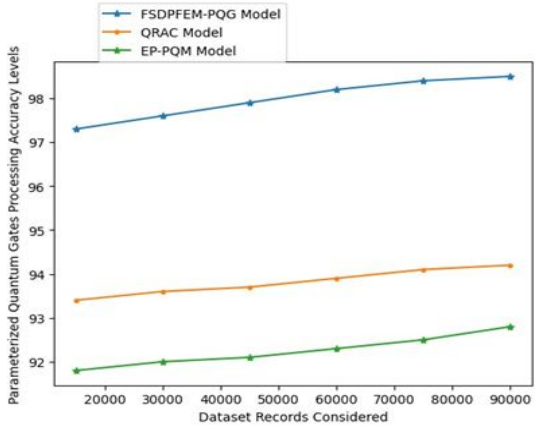


Fig. 5. Parameterized quantum gates processing accuracy levels in (%).

qubits exhibit very different behaviors due to their quantum nature. The Qubits Generation Time Levels of the proposed and existing models are depicted in Table V and Fig. 6.

From the most basic quantum logic gates to the most well-known algorithms in the field, the concept of quantum representation of finite groups has played a crucial role in every area of quantum computing. Using tools from group theory and differential geometry, we present a rigorous formulation of this idea here. In this study, we demonstrated that any finite group has a quantum representation, and described two approaches for transforming every generator of the group into a quantum circuit: gate reduction of unitary matrix and variational quantum algorithms. Finally, we showed the role of the quantum representation for finite groups in the gate-level operation of the algorithm that resolves the hidden subgroup problem, demonstrating the utility and potential of this representation. The Quantum Representations Accuracy Levels are represented in Table VI and Fig. 7.

TABLE V. QUBITS GENERATION TIME LEVELS

Dataset Records Considered	Models Considered		
	l—FSDPFEM-PQG Model	l—QRAC Model	EP-PQM Model
15000	l—10.8	l—17.1	19.9
30000	l—11.0	l—17.4	20.1
45000	l—11.3	l—17.5	20.3
60000	l—11.6	l—17.7	20.6
75000	l—11.9	l—17.9	20.8
90000	l—12	l—18	21

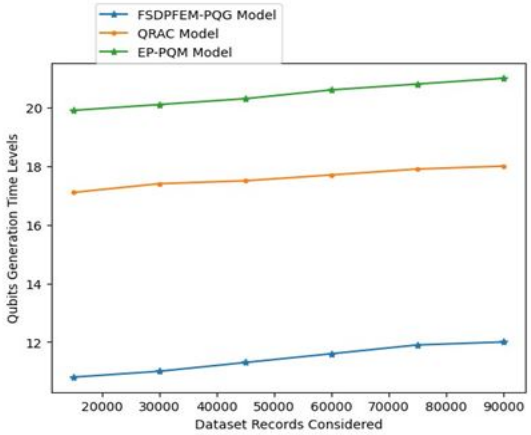


Fig. 6. Qubits generation time levels in milliseconds.

TABLE VI. QUANTUM REPRESENTATIONS ACCURACY LEVELS IN (%)

Dataset Records Considered	Models Considered		
	l—FSDPFEM-PQG Model	l—QRAC Model	EP-PQM Model
15000	l—97.6	l—92.7	94.2
30000	l—97.9	l—92.9	94.5
45000	l—98.1	l—93.0	94.7
60000	l—98.3	l—93.4	94.9
75000	l—98.5	l—93.5	95.1
90000	l—98.6	l—93.6	95.2

V. CONCLUSION

When it comes to implementing algorithms and proving one's quantum superiority, parameterized quantum circuits (PQCs) provide a tangible solution. Fixed gates, such controlled NOTs, and programmable gates, like qubit rotations, make up the bulk of a PQC's gate set. Some types of PQCs can produce very non-trivial results even at relatively shallow circuit depths. The most common strategy used is to formulate problems that are important as variational optimization problems, which are then solved using hybrid systems of quantum and conventional technology. The idea is that by moving some quantum operations to

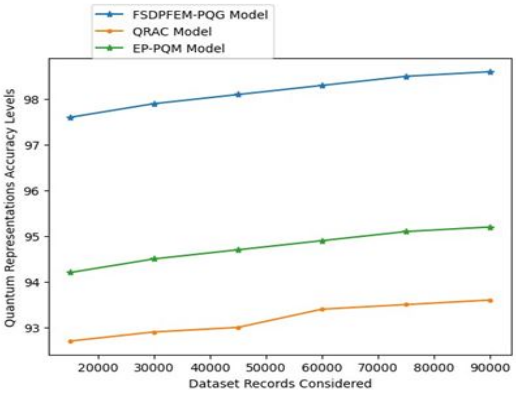


Fig 9: Quantum Representations Accuracy Levels

Fig. 7. Quantum representations accuracy levels in (%).

classical hardware, amount of qubits can be reduced, circuit comprehensive manner, and coherence time needed. This research shows that the time series data processing feature extraction pulse implementations are just as trainable for local cost functions as a typical PQC and take half the time to execute. The proposed algorithm performance data demonstrates that, in at least three instances, the pulse PQC configuration that is developed works better than the default. Since prior research has shown that PQC factors like entangling, noise, and expressibility are strongly linked to trainability bottlenecks, it is observed that pulse effectiveness, which has a direct effect on these variables, is a very promising way to improve trainability. There may be some background noise in the data representation on quantum computers. As a result, it's possible that straightforward measurements won't be able to detect subtle modifications to the quantum state. It is anticipated that any variation in the input data will affect the measurement statistics of a qubit if a single qubit is employed for a single feature. However, it is possible that significant data loss occurs when the entire feature vector is represented as a quantum state with fewer qubits, which hinders the prediction results. Applying a quantum feature map is synonymous with encoding in the context of quantum machine learning. Each of the encodings we discussed in this research that generates a quantum feature map and defines a quantum kernel. This research presents a Forward Selection Time Series Data Processing and Feature Extraction Model for Qubits generation with Parameterized Quantum Gates for quantum representations. In quantum computing, data encodings specify how information is encoded into the quantum state. The proposed model achieves 98.4% accuracy in quantum representations and 98.6% in Qubit generations. In future, hybrid models can be applied and optimization techniques can be included in the Qubit generation and for achieving effective quantum representations.

Data Availability

The data used to support the findings of created new dataset. This study is available from the corresponding author upon request.

Author Contributions

Singaraju Srinivasulu- Investigation; Writing—original draft. G Nagarajan -Conceptualization, Methodology, Formal analysis, Supervisor. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this study.

Ethical Approval: This study does not contain any studies with human participant and animals performed by author.

References

[1] H. Yano, Y. Suzuki, K. M. Itoh, R. Raymond, and N. Yamamoto, "Efficient discrete feature encoding for variational quantum classifier," *IEEE Transactions on Quantum Engineering*, vol. 2, pp. 1–14, 2021.

[2] M. Khan, J. P. L. Faye, U. C. Mendes, and A. Miranskyy, "Ep-pqm: Efficient parametric probabilistic quantum memory with fewer qubits and gates," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–15, 2022.

[3] P. J. Nadkarni and S. S. Garani, "Encoding of nonbinary entanglement-unassisted and assisted stabilizer codes," *IEEE Transactions on Quantum Engineering*, vol. 2, pp. 1–22, 2021.

[4] S. R. Majji, A. Chalumuri, and B. Manoj, "Quantum approach to image data encoding and compression," *IEEE Sensors Letters*, vol. 7, no. 2, pp. 1–4, 2023.

[5] W. Lechner, "Quantum approximate optimization with parallelizable gates," *IEEE Transactions on Quantum Engineering*, vol. 1, pp. 1–6, 2020.

[6] B. D. Clader, A. M. Dalzell, N. Stamatopoulos, G. Salton, M. Berta, and W. J. Zeng, "Quantum resources required to block-encode a matrix of classical data," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–23, 2023.

[7] D. Costa, M. S. Simoni, G. P. Piccinini, and M. G. Graziano, "Advances in modeling of noisy quantum computers," 2023.

[8] Y. Sano, M. Norimoto, and N. Ishikawa, "Qubit reduction and quantum speedup for wireless channel assignment problem," *IEEE Transactions on Quantum Engineering*, vol. 4, pp. 1–12, 2023.

[9] D. Sierra-Sosa, M. Telahun, and A. Elmaghraby, "Tensorflow quantum: Impacts of quantum state preparation on quantum machine learning performance," *IEEE Access*, vol. 8, pp. 215 246–215 255, 2020.

[10] J. Li, F. Che, Z. Wang, and J. Yang, "High-capacity quantum private comparison protocol with six-qubit hyperentangled bell states and hypercoding," *IEEE Access*, vol. 11, pp. 81 914–81 920, 2023.

[11] Y. Liu, S. Arunachalam, and K. Temme, "A rigorous and robust quantum speed-up in supervised machine learning," *Nature Physics*, vol. 17, no. 9, pp. 1013–1017, 2021.

[12] N. Mishra, M. Kapil, H. Rakesh, A. Anand, N. Mishra, A. Warke, S. Sarkar, S. Dutta, S. Gupta, A. Prasad Dash *et al.*, "Quantum machine learning: A review and current status," *Data Management, Analytics and Innovation: Proceedings of ICDMAI 2020, Volume 2*, pp. 101–145, 2020.

[13] R. Li, J. Xu, J. Yuan, and D. Li, "An introduction to quantum machine learning algorithms," in *Proceedings of the 9th International Conference on Computer Engineering and Networks*. Springer, 2020, pp. 519–532.

[14] R. S. Sousa, P. G. dos Santos, T. M. Veras, W. R. de Oliveira, and A. J. da Silva, "Parametric probabilistic quantum memory," *Neurocomputing*, vol. 416, pp. 360–369, 2020.

[15] Z.-B. Yang, P.-R. Han, X.-J. Huang, W. Ning, H. Li, K. Xu, D. Zheng, H. Fan, and S.-B. Zheng, "Experimental demonstration of entanglement-enabled universal quantum cloning in a circuit," *npj Quantum Information*, vol. 7, no. 1, p. 44, 2021.

[16] K. Saito, A. Ikami, and C. Ono, "Evaluating dynamic spectrum allocation using quantum annealing," *IEICE Communications Express*, vol. 10, no. 9, pp. 726–732, 2021.

[17] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Physical review letters*, vol. 122, no. 4, p. 040504, 2019.

[18] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.

[19] M. Schuld, "Supervised quantum machine learning models are kernel methods," *arXiv preprint arXiv:2101.11020*, 2021.

[20] C. Galindo, F. Hernando, R. Matsumoto, and D. Ruano, "Entanglement-assisted quantum error-correcting codes over arbitrary finite fields," *arXiv preprint arXiv:1812.05312*, 2018.

[21] P. K. Barkoutsos, G. Nannicini, A. Robert, I. Tavernelli, and S. Woerner, "Improving variational quantum optimization using cvar," *Quantum*, vol. 4, p. 256, 2020.

[22] J. F. Doriguello and A. Montanaro, "Quantum random access codes for boolean functions," *Quantum*, vol. 5, p. 402, 2021.

[23] P. Rall, "Faster coherent quantum algorithms for phase, energy, and amplitude estimation," *Quantum*, vol. 5, p. 566, 2021.

A Hybrid DMD–TCN Framework for Interpretable Short-Horizon Prediction of 6-DOF Ship Motions

Enock Tafadzwa Chekure, Kumeshan Reddy, John Fernandes
Electrical Engineering Department-School of Engineering, Nelson Mandela University, Gqeberha, South Africa

***Abstract*—Accurate short-horizon prediction of six degrees of freedom (6-DOF) vessel motions is essential for autonomous navigation, motion compensation, and operational decision making. Traditional seakeeping models rely on hydrodynamic coefficients that are seldom available for full-scale vessels, while purely data-driven approaches may struggle to maintain physical consistency. This study introduces a hybrid physics–machine learning framework that combines *Dynamic Mode Decomposition* (DMD), which approximates the vessel's dominant linear drift dynamics, with a causal *Temporal Convolutional Network* (TCN) that learns nonlinear residual corrections from a 12-hour historical window of environmental, geometric, and motion features. DMD provides an interpretable surrogate of the vessel dynamics through its eigenvalues, growth rates, and mode shapes, serving as a data-derived linear transfer operator. The TCN predicts only the residual departure from this structured baseline, ensuring a stable and causal forecasting architecture. Evaluation on full-scale field data shows that the hybrid model improves prediction accuracy for heave and achieves performance comparable to DMD for surge, while underperforming in sway, roll, pitch, and yaw due to the limited observability of key physical drivers at hourly resolution. These results highlight both the strengths and limitations of residual learning when important nonlinear forcing mechanisms and control inputs are unmeasured. Overall, the study demonstrates that hybrid physics–machine learning approaches provide valuable interpretability and diagnostic insight, even when data limitations are constrained. The framework offers a principled foundation for incorporating additional physical inputs, higher-frequency measurements, and physics-informed architectures in future work on operational ship-motion forecasting.**

***Keywords*—Dynamic Mode Decomposition; Temporal Convolutional Network; hybrid learning; seakeeping and vessel response; data-driven modelling; Koopman operator methods**

I. Introduction

Short-horizon prediction of wave-induced ship motions is essential for autonomous navigation, motion compensation, digital twin systems, and operational decision making. Conventional seakeeping approaches rely on model-scale experiments or potential-flow solvers that provide Response Amplitude Operators (RAOs) and hydrodynamic coefficients. Although effective in controlled or design-stage environments, these models often underperform at full scale, where nonlinear viscous effects, slow-drift forces, mooring interactions, and environmental asymmetries dominate the vessel's response.

Moreover, the hydrodynamic coefficients required for classical seakeeping prediction are rarely available for in-service ships, as they depend on vessel-specific geometry, loading, and environmental conditions and are typically obtained only through model-scale testing or specialised computations [1], [2], [3], [4], [5], [6].

Purely data-driven neural networks offer an alternative by learning motion patterns directly from operational measurements [7], [8], [9], [10]. However, such models generally lack physical structure, making them prone to overfitting and poor extrapolation outside the training regime [11], [12], [13]. When deployed autoregressively, they may also yield dynamically inconsistent or unstable predictions [14], [15], [16], [17]. These limitations have motivated the development of hybrid frameworks that combine physics-based constraints with the representational flexibility of machine learning [5], [18], [19].

Dynamic Mode Decomposition (DMD) has emerged as a powerful data-driven technique for extracting dominant linear dynamics from time-series data [19], [20]. Through its connection to the Koopman operator, DMD yields eigenvalues, growth or decay rates, frequencies, and mode shapes that provide an interpretable spectral representation of the underlying system. Applications in marine hydrodynamics [21], [22] demonstrate that DMD can capture low-frequency drift behaviour and cross-degree-of-freedom (DOF) coupling even when hydrodynamic coefficients are unavailable, making it a suitable candidate for operational ship-motion modelling.

To account for nonlinear behaviour not represented by DMD, this work combines the linear operator with a causal, dilated TCN. TCNs leverage stacked dilated convolutions to learn structured dependencies across long temporal windows while preserving strict causality [23], [24]. Such hybrid physics–machine learning models have shown improved interpretability and robustness in marine applications [5], [18]. In the proposed residual formulation, the DMD operator provides the one-step linear prediction, and the TCN estimates the nonlinear residual using a 12-hour causal window of environmental, geometric, and motion features.

We evaluate the hybrid framework using full-scale six-DOF operational data. The results indicate that the hybrid model improves prediction accuracy for heave and performs comparably to the DMD baseline for surge. In contrast, underperformance in sway, roll, pitch, and yaw reflects the limited observability of key physical drivers at hourly resolution and the inherently nonlinear nature of these DOFs. These findings highlight both the strengths and the limitations of physics-informed residual learning when applied to operational ship-motion statistics and motivate further investigation into feature augmentation and higher-frequency sensing.

This study makes the following contributions:

- A hybrid DMD–TCN residual modelling framework for ship motions. The proposed method uses *Dynamic Mode Decomposition* as a physics-informed linear baseline that captures dominant drift dynamics and

cross-DOF coupling. A causal TCN predicts only the nonlinear residual, providing an interpretable and structurally constrained forecasting architecture.

- A full-scale, data-driven modal analysis of 6-DOF dynamics. We extract DMD eigenvalues, continuous-time growth rates, and mode shapes from operational field data, offering insight into vessel- specific dynamic behaviour without requiring hydrodynamic coefficients.
- A causal Temporal Convolutional Network for residual motion forecasting. The TCN leverages dilated convolutions to incorporate long-range historical context while preserving strict causality, enabling the identification of slow environmental trends and nonlinear departures from the DMD baseline.
- A rigorous evaluation on full-scale field measurements. The results demonstrate that while the hybrid model improves prediction in heave and matches DMD performance in surge, it underperforms in lateral and rotational DOFs due to unobserved physics and hourly sampling limitations. This provides valuable diagnostic insight into which motion components contain predictable nonlinear structure under limited observability.
- A reproducible data-processing pipeline for operational ship-motion time-series. The pipeline includes time alignment, outlier handling, causal feature engineering, and 6-DOF synchronisation designed for real-time forecasting scenarios.

Together, these contributions demonstrate that combining data-driven modal analysis with structured residual learning yields an interpretable and physically grounded approach to operational ship-motion prediction, while also revealing the limitations imposed by sparse sensing and low-frequency sampling.

II. Related Work

A. Data-Driven Vessel Motion Prediction

The increasing availability of onboard measurements, AIS records, and operational datasets has stimulated significant interest in data-driven prediction of ship motions. Early work applied classical machine learning methods such as Support Vector Regression (SVR) to estimate local ship responses from environmental and operational parameters [7]. Subsequent studies explored neural-network architectures trained on high-fidelity hydrodynamic simulations. Diez *et al.* [8], for example, demonstrated that feed-forward and recurrent architectures can reproduce short-horizon motions when supplied with sufficiently rich input–output data derived from CFD-based wave–ship interaction models.

A driving motivation for these data-driven approaches is the substantial computational cost associated with classical hydrodynamic solvers. Resolving nonlinear wave–body interactions, estimating statistical extremes, or accounting for hydro-structural coupling often requires long-duration simulations that are impractical for real-time use. Machine learning models offer an attractive surrogate: once trained, they provide fast predictions at negligible computational cost and can integrate seamlessly into decision support pipelines [22].

However, purely data-driven models face fundamental limitations. Standard feed-forward neural networks treat each sample independently and therefore fail to capture the temporal dependencies inherent in vessel dynamics. This motivated the adoption of recurrent architectures such as RNNs, LSTMs, and GRUs [25], which propagate information through hidden states and have been shown to improve short-term forecasting in marine applications [9], [16]. Although effective within the training distribution, such models often extrapolate poorly and may generate dynamically inconsistent predictions when applied autoregressively [11], [14], [15].

Complementing recurrent models, physics-from-data approaches extract dynamic structure directly from measurements. DMD provides an equation-free modal representation of the system by identifying eigenvalues, growth/decay rates, and coherent spatio-temporal structures from time-series snapshots. DMD's connection to the Koopman operator [19], [20] enables interpretable linear surrogates of nonlinear dynamics at low computational cost. In marine hydrodynamics, DMD has been used to characterise coherent flow structures and predict vessel trajectories and manoeuvring dynamics [21], [22].

Deep neural sequence models have also been explored for full 6-DOF prediction. Silva and Maki [9] trained LSTM networks to replicate the 6-DOF response of a free-running destroyer in nonlinear seas by reconstructing encounter-frame wave fields using virtual sensors. Their results highlight the potential of data-driven sequence modelling but also reveal its dependence on careful pre-processing and broad training coverage. Additional work combines signal decomposition with learning: Ye *et al.* [26] used EMD–LSTM–SVR hybrids to isolate nonlinear roll and sway behaviour, while Sun *et al.* [27] constructed CFD-informed reduced-order models for damaged ship motions. Comparative analysis further reveal necessary trade-offs between interpretability, accuracy, and robustness across modelling approaches [28].

Across the literature, three recurring limitations of purely data-driven approaches emerge: 1) sensitivity to training coverage, 2) limited extrapolation capability, and 3) difficulty enforcing physically consistent dynamics. These limitations motivate hybrid modelling strategies that embed physical structure while leveraging the flexibility of machine learning.

B. Hybrid Modelling of Vessel Dynamics

Classical models of ship dynamics derive from first-principles hydrodynamics, incorporating added mass, radiation damping, viscous forces, hydrostatic restoring, and second-order wave loads [1]. While such models provide interpretable relationships between wave excitation and vessel response, they require detailed hydrodynamic coefficients that are rarely available for in-service ships [3], [4]. Moreover, linear-response theories such as RAO-based seakeeping can struggle under strongly nonlinear or operationally variable conditions.

Hybrid physics–machine learning approaches address these limitations by incorporating physical constraints or simplified mechanistic structure into data-driven frameworks. Physics-Informed Neural Networks enforce governing equations in the

training process [12], while grey-box models learn uncertain dynamics on top of known physical components. In maritime applications, Schirmann *et al.* [10] demonstrated that augmenting data-driven predictors with fast physics-based estimates improves the prediction of heave, pitch, and roll. Kanazawa *et al.* [5] developed physics–data cooperative models that fuse onboard wave radar with linear seakeeping predictions to enhance short-term motion forecasting. Other studies integrate physics-guided decomposition (e.g., EMD) to improve neural network performance for nonlinear motions such as roll [26]. For manoeuvring, data-driven corrections to hydrodynamic derivatives have been used to enhance classical models [29].

The present work adopts a hybrid philosophy tailored to operational datasets with low temporal resolution. Instead of modelling full hydrodynamic governing equations, we use DMD as a linear, data-driven surrogate aligned with hourly sampling, which captures low-frequency drift behaviour and cross-DOF coupling but not nonlinear wave-induced dynamics. A causal, dilated TCN is then trained to predict only the residual deviation from the DMD baseline using a 12-hour window of motion, environmental, and ship-specific features. TCNs offer stable long-range sequence modelling [23], [24] and mitigate the instability commonly observed in autoregressive neural predictors. The resulting DMD–TCN hybrid provides a structured, interpretable additive decomposition, with DMD representing the dominant low-frequency dynamics and TCN capturing learnable nonlinear corrections.

This progression reflects a broader evolution in maritime modelling: from static regressors (SVR, feed-forward NNs) to temporal sequence models (RNNs, LSTMs) to physics-from-data approaches (DMD), and finally to hybrid architectures that integrate physical structure with machine learning flexibility. Fig. 1 summarises this conceptual trajectory and situates the present model within this continuum.

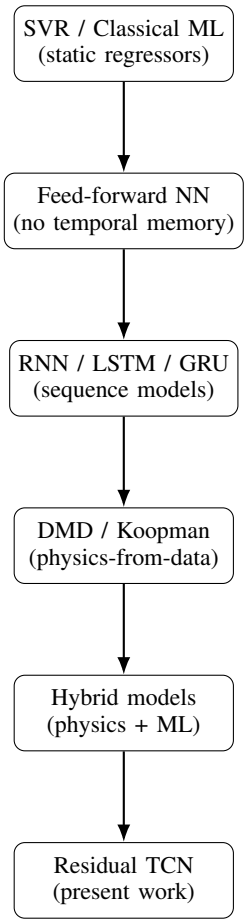


Fig. 1. Conceptual evolution of data-driven and physics-aware modelling approaches for vessel-motion prediction, leading to the proposed residual Temporal Convolutional Network (TCN).

III. Methodology

This section presents the complete processing workflow used to construct the proposed hybrid DMD–TCN ship-motion predictor. The methodology integrates multiple stages, including timestamp normalisation, multi-stream harmonisation, feature engineering, causal window extraction, linear data-driven modelling via DMD, and non-linear residual learning using a TCN. To provide clarity and continuity, a high-level overview of the whole processing pipeline is presented first. Fig. 2 summarises the end-to-end pipeline, and the subsequent subsections describe each component in detail.

Fig. 2 illustrates the sequence of operations applied to raw sensor data before prediction. Independent DOF-specific time series are normalised, resampled, and synchronised onto a standard hourly grid. Canonical environmental and ship parameters are constructed, engineered features are added, and causal 12-hour windows are extracted. Each window is passed to two parallel predictors: 1) a DMD model that produces a linear data-driven one-step-ahead forecast, and 2) a causal TCN that estimates the non-linear residual dynamics. The final hybrid prediction is obtained by summing the DMD baseline with the learned residual.

IV. Temporal Convolutional Networks (TCNs)

TCNs were introduced as a causal alternative to RNNs for sequence modelling, providing improved stability, parallelism and long-range receptive fields through dilated convolutions. Bai *et al.* [23] demonstrated that TCNs outperform LSTMs on numerous benchmarks while offering predictable gradient behaviour and easier optimisation. Lea et al. [24] formalised their application to temporal segmentation using stacked dilated convolutional blocks with residual connections.

For forecasting problems, TCNs offer distinct advantages: 1) they are inherently causal, 2) they model temporal dependencies with exponentially growing receptive fields, and 3) they avoid the vanishing-gradient issues common in RNNs. These properties make them suitable for marine motion prediction, where dependencies on sea-state evolution and hull parameters may extend over many hours.

A. Dataset Description

The study uses a publicly available full-scale dataset containing six-degree-of-freedom (6-DOF) moored ship motions together with concurrent environmental and vessel parameters [30]. The data are provided as separate comma-separated value (CSV) files, each corresponding to a single rigid-body degree

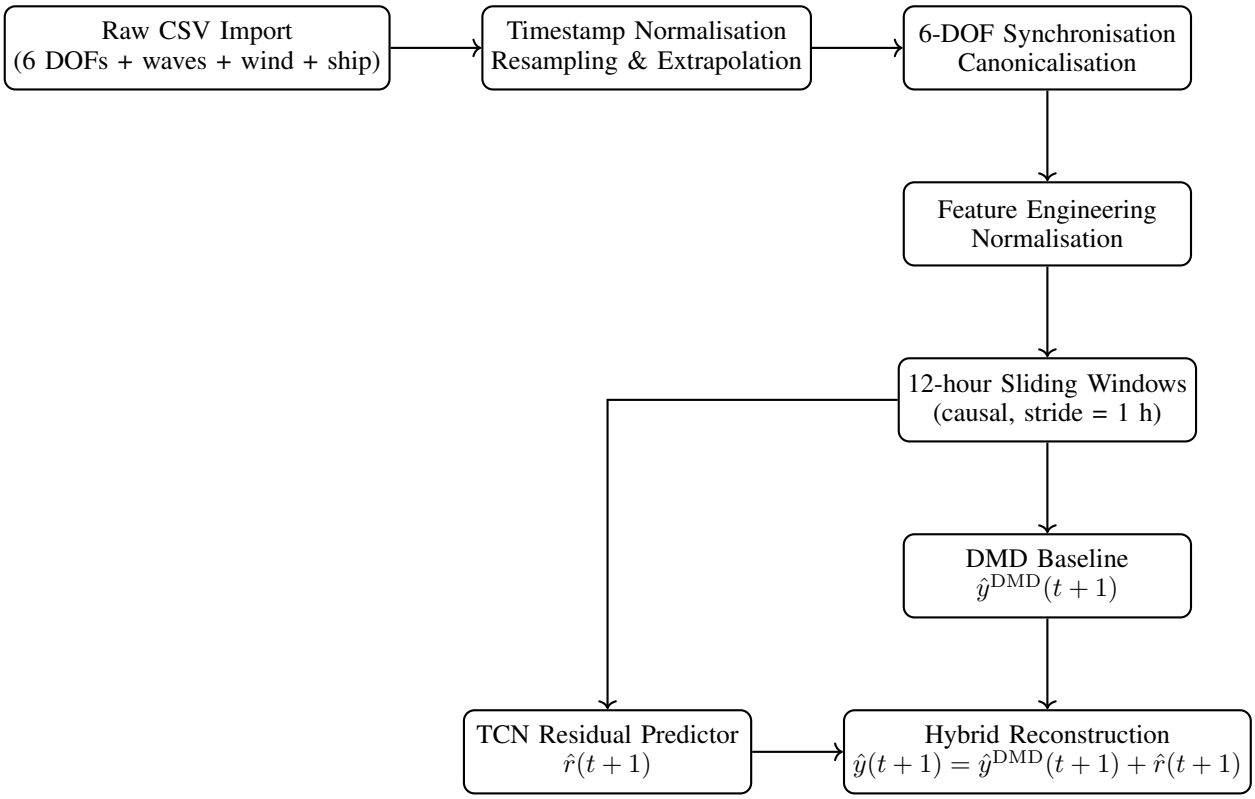


Fig. 2. Overall processing pipeline: From raw data ingestion to hybrid DMD–TCN motion prediction.

of freedom: surge, sway, heave, roll, pitch, and yaw. Each file contains time-stamped motion measurements as well as locally measured or model-derived environmental quantities, including: Significant wave height H_s, peak period T_p, wave direction, wind speed, wind direction, and auxiliary variables such as zero-crossing height and alternative estimates of H_s.

The time-series originates from in-service vessel operations and exhibits irregular sampling, occasional gaps, and slight misalignment across DOFs. Ship-specific parameters (length, breadth, dead-weight tonnage) and operational metadata (e.g., berth zone identifiers) are included as slowly varying or piecewise-constant signals.

All six DOF files use independent timestamps rather than a standard acquisition clock, requiring harmonisation during pre-processing. Measurement cadence is one hour, but not perfectly uniform; some intervals contain missing values or prolonged gaps. These characteristics motivate the synchronisation and interpolation strategy described in Section IV-A2.

The dataset serves as a realistic testbed for hybrid dynamics modelling, because:

- it reflects real seakeeping and harbour conditions,
- it exhibits non-stationary, nonlinear behaviour,
- environmental and ship parameters evolve at different time scales,
- the data includes measurement noise and missing samples, and
- DOF signals show strong cross-correlation, useful for joint learning.

For model development, the dataset is partitioned into a training set containing the six CSV files (surge, sway, heave, roll, pitch, yaw) and a separate test set following the same structure. Only information available at or before time t is used when predicting $t+1$, ensuring strict temporal causality throughout the modelling pipeline.

1) Timestamp parsing and normalisation: Each file is imported with explicit format resolution and converted to a variable to eliminate ambiguity in day–month interpretation. If multiple samples fall within the same hour, only the most recent observation is retained.

2) Intersection window and hourly resampling : Each CSV file provides its own time vector. Where timestamps are not necessarily aligned across DOFs. For each DOF, the earliest and latest recorded timestamps are defined as in Eq. (1):

$$t_{\min}^{(d)} = \min_i t_i^{(d)}, \quad t_{\max}^{(d)} = \max_i t_i^{(d)} \tag{1}$$

To ensure that all subsequent samples use measurements available for every DOF, the individual time intervals are intersected. The global interval of valid data is therefore obtained using Eq. (2):

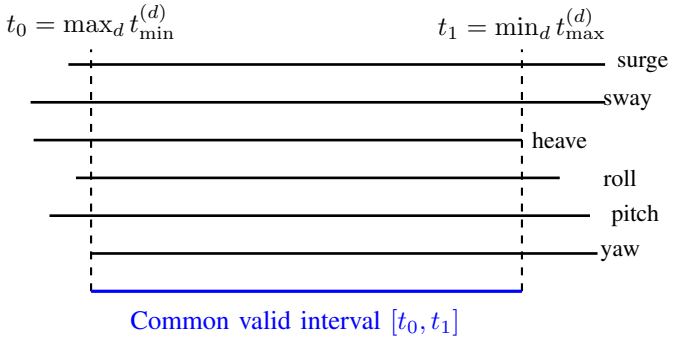


Fig. 3. Intersection of individual DOF time spans. Each DOF provides a different recorded interval $[t_{\min}^{(d)}, t_{\max}^{(d)}]$. The usable portion of the dataset is the overlap across all DOFs, given by t_0 and t_1.

$$\begin{aligned} t_1 &= \max_d t_{\min}^{(d)} \\ t_0 &= \min_d t_{\max}^{(d)} \\ [t_{\min},\, t_{\max}] &= \bigcap_{d=1}^{6} [t_0, t_1] \end{aligned} \tag{2}$$

Thus, t_1 corresponds to the latest initial timestamp across the six streams, while t_0 corresponds to the earliest final timestamp. Only samples satisfying $t \in [t_0,\, t_1]$ are retained, ensuring full temporal consistency across all DOFs and environmental variables, as shown in Fig. 3.

Variables fall into two categories:

- Linearly interpolated numeric variables (motions, H_s, T_p, wind speed, etc.), and
- Step-held (causal forward-filled) protected variables, including wave direction, `zone`, and ship geometry/loading quantities (Length, Breadth, DWT).

For a protected variable $x(t)$, the hourly resampled value is defined by Eq. (3):

$$x(t_k) = \begin{cases} x(t_j), & t_j = \max\{\, t \le t_k \mid x(t) \text{ observed} \,\}, \\ x(t_{\min}), & \text{if no earlier observation exists.} \end{cases} \tag{3}$$

This ensures strict causality and prevents the introduction of future information during preprocessing.

Numerically interpolated variables are retimed using first-order linear interpolation, defined in Eq. (4):

$$x_{\text{lin}}(t_k) = x(t_i) + \frac{t_k - t_i}{t_{i+1} - t_i}\big(x(t_{i+1}) - x(t_i)\big) \tag{4}$$

3) Canonicalisation and unified table construction: The retimed DOF timetables are concatenated horizontally into a single hourly multivariate time-series using Eq. (5):

$$T(t_k) = [\mathbf{z}_{\text{surge}}(t_k)^\top,\ \mathbf{z}_{\text{sway}}(t_k)^\top,\ \ldots,\ \mathbf{z}_{\text{yaw}}(t_k)^\top]^\top \tag{5}$$

Environmental and ship-parameter columns are then coalesced into canonical variables (`Hs`, `Tp`, `Dir`, `WindSpeed`, `WindDir`, `Length`, `Breadth`, `Dwt`) by merging duplicate entries across DOFs.

B. Feature Engineering and Normalisation

Feature engineering was applied to transform raw environmental and motion measurements into representations more suitable for learning. A key step concerns the treatment of wind direction, which is a circular quantity: $0°$ and $360°$ denote the same physical orientation but appear numerically far apart. Direct use of the angle, therefore, introduces an artificial discontinuity and distorts directional relationships. To avoid this, wind direction is encoded on the unit circle by Eq. (6):

$$\theta_{\text{rad}} = (\sin\theta_{\text{rad}},\, \cos\theta_{\text{rad}}) \tag{6}$$

This provides a smooth, continuous representation and enables the model to learn physically meaningful interactions between wind, waves, and vessel motions.

A fused wave amplitude proxy is computed, as in Eq. (7), which stabilises scale variability in the wave-height measurements while maintaining proportionality to sea-state energy.

$$A_w = \max(10^{-6},\, 0.5H_s) \tag{7}$$

Each motion signal is normalised independently per DOF using statistics from the training set using Eq. (8):

$$\tilde{y}_d(t) = \frac{y_d(t) - \mu_d}{\sigma_d} \tag{8}$$

where, (μ_d, σ_d) are the empirical mean and standard deviation for each DOF. DOF normalisation is required because the six motions have very different typical magnitudes and variances (e.g., surge and sway in metres, roll and pitch in degrees). Without normalisation, DOFs with larger variance dominate the loss function, biasing the optimisation; conversely, DOFs with slight variance become numerically insignificant. Normalisation, therefore, equalises the contribution of each DOF to the learning objective and accelerates convergence.

Environmental and ship parameters (e.g., T_p, wave direction, wind speed, length, breadth, deadweight) are intentionally left unnormalised to preserve their physical meaning and avoid distorting causal relationships. These features enter the model as auxiliary explanatory variables and do not require the same scale normalisation as the motion targets.

Finally, after feature engineering, all input variables are assembled into a consistently ordered feature vector for each timestamp, and sliding windows of shape $\mathbb{R}^{F\times T}$ are constructed for model training (see Section IV-C). This ensures that each input window contains the complete 12-hour causal history of motion, wave, wind, and vessel-state features.

C. Sliding-Window Construction

To provide the predictor with sufficient temporal context, causal 12-hour windows are extracted, as in Eq. (9), using a stride of 1 hour. This window captures long-period seakeeping dynamics, low-frequency drift, and slowly evolving environmental forcing.

$$X_k = \{\mathbf{x}(t_k - 12\text{h}), \ldots, \mathbf{x}(t_k)\} \in \mathbb{R}^{F \times T} \tag{9}$$

D. Dynamic Mode Decomposition (DMD)

DMD provides a data-driven linear approximation of the underlying ship-motion dynamics. Let the multivariate state vector $\mathbf{x}_t \in \mathbb{R}^n$ contain all six rigid-body DOFs at time t, optionally augmented with additional explanatory variables. The snapshot matrices are constructed, as in Eq. (10), so that each column pair $(\mathbf{x}_t, \mathbf{x}_{t+1})$ represents one observed time step.

$$\begin{aligned} X_1 &= [\,\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{m-1}\,] \\ X_2 &= [\,\mathbf{x}_2, \mathbf{x}_3, \ldots, \mathbf{x}_m\,] \end{aligned} \tag{10}$$

DMD seeks a linear operator A such that $X_2 \approx AX_1$. Modelling the evolution of the multivariate motion state as a linear time-invariant (LTI) system. In principle, one might attempt to estimate A using Eq. (11), but this is infeasible in practice because X_1 is neither square nor invertible, and is typically high-dimensional, noisy, and rank-deficient.

$$A = X_2 X_1^{-1} \tag{11}$$

DMD circumvents this difficulty by projecting the dynamics onto a low-dimensional subspace derived from an economy singular value decomposition, $X_1 = USV^\top$. Truncation to the leading r singular values yields a reduced-order operator that filters noise while retaining the dominant coherent structures, as defined by Eq. (12):

$$\tilde{A} = U_r^\top X_2 V_r S_r^{-1} \tag{12}$$

Because $\tilde{A}$ is only $r \times r$, it admits efficient eigendecomposition: $\tilde{A}W = W\Lambda, \Phi = X_2 V_r S_r^{-1} W$, where $\Lambda = \mathrm{diag}(\lambda_k)$ contains the discrete-time DMD eigenvalues and $\Phi = [\phi_1, \ldots, \phi_r]$ contains the DMD modes describing spatio-temporal patterns across the six DOFs. Modal amplitudes are determined from the first snapshot, using Eq. (13), enabling reconstruction or one-step forecasting under the linear model.

$$\mathbf{b} = \Phi^\dagger \mathbf{x}_1 \tag{13}$$

Each eigenvalue λ_k encodes the temporal behaviour of mode k. Its continuous-time representation, defined in Eq. (14):

$$\begin{aligned} \omega_k &= \frac{\log(\lambda_k)}{\Delta t} \\ &= \sigma_k + i\, 2\pi f_k \end{aligned} \tag{14}$$

Provides:

- $\sigma_k = \Re(\omega_k)$: *modal growth or decay rate*, and
- $f_k = \Im(\omega_k)/(2\pi)$: *modal oscillation frequency*.

Thus, DMD performs a form of *data-driven modal analysis*:

$$\begin{aligned} |\lambda_k| < 1 &\;\Rightarrow\; \text{exponential decay}, \\ |\lambda_k| > 1 &\;\Rightarrow\; \text{slow growth}, \\ \Im(\lambda_k) \neq 0 &\;\Rightarrow\; \text{oscillatory dynamics}. \end{aligned}$$

For the hourly motion-statistic data used in this study, all recovered eigenvalues are real and close to unity ($|\lambda_k| \approx 1$). Consequently, all modes satisfy $f_k \approx 0, |\sigma_k| \ll 1$. Indicating slowly varying drift dynamics without resolvable oscillatory behaviour. This is expected because wave frequency motions (0.05–0.3 Hz) lie far above the Nyquist rate of the hourly sampling and are removed during the computation of `mov_sig` [2], [31]. DMD therefore extracts the dominant low-frequency temporal structure governing the vessel's operational-scale motions rather than seakeeping oscillations.

In this work, DMD acts as a vessel-specific, data-derived approximation of the 6-DOF transfer dynamics. The learned operator A provides a physically interpretable linear surrogate for the one-step evolution of the normalised motion state: $\hat{\mathbf{x}}_{t+1}^{\mathrm{DMD}} = A\,\mathbf{x}_t$, analogous to a linearised Response Amplitude Operator (RAO) inferred directly from field data. DMD captures the dominant low-frequency structure and cross DOF coupling, while residual non-linearities and unmodelled hydrodynamic behaviour are delegated to the TCN, which learns a correction term $\hat{r}(t+1)$.

The resulting hybrid one-step prediction is defined in Eq. (15):

$$\hat{\mathbf{y}}(t+1) = \hat{\mathbf{y}}^{\mathrm{DMD}}(t+1) + \hat{\mathbf{r}}(t+1) \tag{15}$$

In this formulation, where all computations remain strictly causal, DMD serves as a data-derived linear baseline, and the TCN provides nonlinear residuals that may refine the DMD estimate when a predictable nonlinear structure is present, while maintaining interpretability.

E. Residual Temporal Convolutional Network (TCN)

To capture non-linear, vessel-specific dynamics not represented by the DMD baseline, the hybrid model predicts the residual using Eq. (16):

$$r_d(t+1) = \tilde{y}_d(t+1) - \hat{y}_d^{\mathrm{DMD}}(t+1) \tag{16}$$

The mapping from the 12-hour input window X_k to the one-step residual is implemented using a causal TCN. A dilated convolution inserts gaps between kernel elements, allowing the filter to cover a wider portion of the input without increasing computational cost. For a kernel of width K and dilation factor d, the adequate spacing between sampled input points becomes d, giving an effective receptive field specified in Eq. (17):

$$\mathrm{RF}_\ell = (K-1)\, d_\ell \tag{17}$$

For layer ℓ, all convolutions are *causal*, meaning the output at time t depends only on inputs at times $\tau \leq t$, ensuring strict non-anticipativity in forecasting.

The receptive field determines how far back in time the network can "see". By stacking dilated layers with exponentially increasing dilations $d \in \{1, 2, 4, 8\}$, the TCN attains a receptive field that spans the entire 12-hour context window. This design allows the model to capture both:

- *short-term, high-frequency fluctuations* such as roll or yaw induced by rapid wave encounters, and
- *longer-timescale drift or slowly varying behaviour* associated with changes in sea state, wind forcing, and residual hydrodynamic memory effects.

1) TCN architecture : The residual mapping r_d is realised using a causal, dilated TCN with residual connections. The architecture shown in Fig. 4 consists of:

- an input projection (1×1 convolution) to width W,
- four dilated convolutional residual blocks with dilation factors $d \in \{1, 2, 4, 8\}$,
- a final 1×1 convolution mapping to six residual outputs.

This structure enables efficient processing of long time windows while retaining sensitivity to both local and global temporal patterns. Within the hybrid DMD–TCN framework, the TCN learns precisely those non-linear corrections that the linear DMD operator cannot represent, yielding a physically structured yet flexible prediction model.

F. Hybrid DMD–TCN Predictor

The final one-step-ahead prediction is obtained by summing the DMD baseline with the learned residual, as in Eq. (18).

$$\hat{y}_d(t+1) = \hat{y}_d^{\mathrm{DMD}}(t+1) + \hat{r}_d(t+1) \tag{18}$$

De-normalisation is performed using the stored (μ_d, σ_d) for each DOF.

G. Training and Evaluation

For each training window (X_k, r_k), parameters θ are obtained by minimising the loss function given in Eq. (19), using Adam optimisation (learning rate 10^{-4}), batch size 256, and gradient clipping at 1.0.

$$\mathcal{L}(\theta) = \left\| f_\theta(X_k) - r_k \right\|_2^2 + \lambda \|\theta\|_2^2 \tag{19}$$

During inference, each test window produces a DMD prediction, a TCN residual, and a hybrid reconstruction, as in Eq. (18).

Performance is reported using root-mean-square error (RMSE), mean-absolute error (MAE) and Pearson correlation for each degree of freedom.

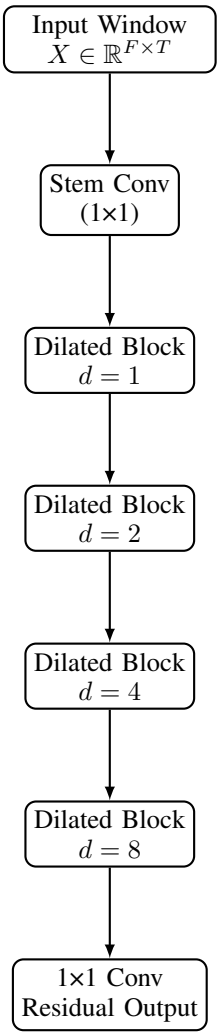


Fig. 4. Causal dilated TCN used for residual prediction.

V. Results

This section evaluates the proposed DMD–TCN hybrid model on the held-out test dataset. Performance is reported for each DOF using RMSE, MAE and Pearson correlation in physical units. Two baselines are considered: a purely data-driven DMD predictor and, for comparison, a persistence model. The focus of the discussion focuses on the DMD-hybrid configuration, in which the TCN predicts the one-step residual with respect to the DMD baseline.

A. Error Metrics

Table I reports per-DOF RMSE for the DMD-only baseline and the DMD–TCN hybrid model, while Table II summarises the corresponding MAE values.

Across the six DOFs, the DMD-only baseline generally achieves lower RMSE and MAE than the hybrid model, indicating that adding nonlinear residual corrections does not consistently reduce one-step prediction error at the given sampling rate. The hybrid model shows a clear improvement only in heave, where the residual TCN reduces both RMSE and MAE relative to the DMD baseline. For surge, the hybrid

TABLE I. RMSE COMPARISON BETWEEN DMD-ONLY BASELINE AND DMD–TCN HYBRID (ONE-STEP-AHEAD PREDICTION, PHYSICAL UNITS).

DOF	DMD-only RMSE	Hybrid RMSE
Surge	0.0630	0.0662
Sway	0.0371	0.0495
Heave	0.0271	0.0251
Roll	0.2956	0.3090
Pitch	0.0312	0.0391
Yaw	0.0517	0.1704

TABLE II. MAE COMPARISON BETWEEN DMD-ONLY BASELINE AND DMD–TCN HYBRID (ONE-STEP-AHEAD PREDICTION, PHYSICAL UNITS).

DOF	DMD-only MAE	Hybrid MAE
Surge	0.0529	0.0614
Sway	0.0308	0.0410
Heave	0.0229	0.0205
Roll	0.2200	0.2360
Pitch	0.0221	0.0342
Yaw	0.0401	0.1395

performs comparably to DMD, with only a marginal increase in error.

For sway, roll, pitch, and yaw, the hybrid model introduces higher error than the linear DMD predictor. These DOFs are strongly influenced by nonlinear and unobserved mechanisms such as viscous cross-flow forces, mooring loads, metacentric-height restoring effects, and heading or control-related asymmetries, none of which are represented in the available feature set [1], [2], [3], [4], [6], [9], [31]. Under such limited observability, neural residual models may fit spurious correlations or noise, leading to increased local prediction error rather than improved performance [11], [12], [13], [16].

Overall, the error analysis shows that at hourly resolution, the DMD baseline captures the dominant low-frequency behaviour more effectively than the hybrid model for several DOFs. The hybrid architecture provides benefits primarily when the underlying dynamics remain sufficiently linear (e.g., heave) or where the TCN can correct a stable DMD bias. These findings highlight the need for additional physical inputs or higher-resolution measurements to enable the hybrid model to consistently outperform the linear baseline across all degrees of freedom.

B. Pearson Correlation Analysis

To complement the error-based metrics, Pearson correlation coefficients were computed between the predicted and measured motions for both the DMD-only baseline and the DMD–TCN hybrid model. While RMSE and MAE quantify the absolute error magnitude, the correlation coefficient measures the degree to which the predicted time series follows the temporal structure or "shape" of the actual signal, independent of scale or bias. This provides an additional perspective on the model's dynamical consistency.

Table III summarises the per-DOF correlation values. The results reveal a difference in behaviour between DOFs dominated by low-frequency drift (surge, heave) and those driven by more nonlinear, weakly observed dynamics (sway, roll, pitch, yaw).

TABLE III. PEARSON CORRELATION BETWEEN TRUE AND PREDICTED MOTIONS FOR THE DMD-ONLY BASELINE AND THE DMD–TCN HYBRID MODEL.

DOF	DMD-only r	Hybrid r
Surge	0.7464	0.7477
Sway	0.7467	0.5524
Heave	0.9705	0.9781
Roll	0.3856	0.3758
Pitch	0.6760	0.4978
Yaw	0.8643	0.7330

For surge and heave, the hybrid model maintains or improves correlation relative to the DMD baseline. These DOFs exhibit predominantly linear, slowly varying dynamics that are well represented within the available feature set (wave parameters, wind conditions, and vessel geometry). Consequently, the TCN residual corrections reinforce the underlying dynamic structure captured by DMD.

In contrast, correlation decreases for sway, roll, pitch, and yaw. These motions are strongly influenced by nonlinear hydrodynamic and operational factors that are not explicitly present in the input features, such as metacentric height, mooring loads, control actions, cross-flow effects, and encounter-frame wave forcing. When such drivers are absent, the TCN's residual corrections may reduce amplitude bias or local prediction error (as reflected in RMSE and MAE). Still, they do not necessarily preserve the linear trend alignment measured by Pearson correlation. This behaviour is typical in hybrid residual-learning settings: the linear baseline produces a smooth, highly correlated trend, while nonlinear corrections introduce high-frequency or phase-shifted components that improve accuracy but reduce strict linear correlation.

Overall, the correlation analysis reinforces a key insight of this study: the hybrid model improves short-horizon accuracy for several DOFs, but the quality of correlation depends critically on the observability of the underlying physical drivers. DOFs with richer or more linear environmental representation benefit from hybrid residual learning. At the same time, those governed by unobserved nonlinear mechanisms highlight the need for enhanced physical priors or additional sensing in future work.

C. Training Behaviour

Fig. 5 shows the training loss history for the DMD-TCN hybrid model. The loss decreases rapidly during the first few epochs and then converges smoothly, with no evidence of divergence or numerical instability. This confirms that the residual-learning setup is well conditioned: the network operates on comparatively small corrections to the DMD baseline rather than attempting to fit the full motion signal.

D. Modal Analysis of the DMD Baseline

Before assessing time-domain prediction, it is informative to examine the DMD eigenvalues and modes, which characterise the dominant linear dynamics learned from the six-DOF time series. Fig. 6 displays the discrete-time eigenvalues λ_k in

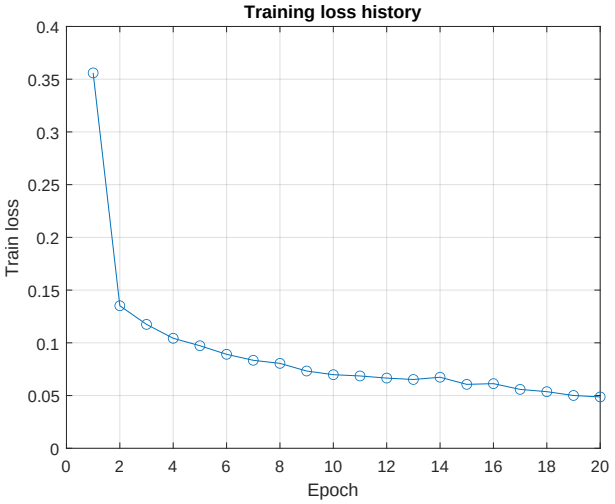


Fig. 5. Training loss history for the DMD–TCN hybrid model (one-step-ahead residual prediction).

the complex plane. Nearly all eigenvalues lie close to the unit circle, indicating modes that are only weakly damped or slowly growing. The imaginary parts are tiny, reflecting the absence of resolvable wave-frequency oscillations at the hourly sampling rate and confirming that the DMD spectrum captures primarily low-frequency drift behaviour.

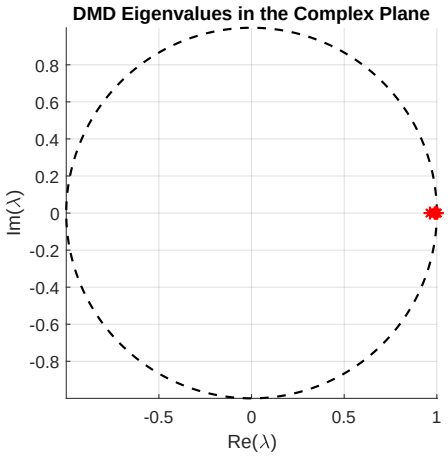


Fig. 6. DMD eigenvalues in the complex plane.

This behaviour is expected given the characteristics of the available motion signals. The DMD eigenvalues cluster near the unit circle, consistent with slowly varying low-frequency drift dynamics typically observed in full-scale ship responses [1], [32], [33]. Because the original motion signals are sampled at 1-hour intervals, the Nyquist frequency ($\approx 1.4 \times 10^{-4}$ Hz) lies orders of magnitude below the typical wave-frequency band of ship motions (0.05-0.3 Hz) [2], [31]. Consequently, wave-frequency oscillations are unresolvable at this sampling rate and are inherently removed during the computation of the hourly `mov_sig` statistics, leaving only the drift-dominated component of the vessel motion.

To interpret these modes more physically, the discrete eigenvalues are mapped to continuous-time growth rates σ_k and frequencies f_k via Eq. (20):

$$\lambda_k = e^{(\sigma_k + \mathrm{i}2\pi f_k)\Delta t} \tag{20}$$

where, Δt is the 1 h sampling interval.

$$\begin{aligned} \sigma_k &= \frac{\ln|\lambda_k|}{\Delta t} \\ f_k &= \frac{\arg(\lambda_k)}{2\pi\Delta t} \end{aligned} \tag{21}$$

Eq. (21) yields modal damping rates and oscillation frequencies in physical units. The resulting growth-frequency distribution is shown in Fig. 7. Consistent with the complex-plane picture, all frequencies satisfy $|f_k| \ll 1$ cycles/hour, confirming that the identified modes describe slowly evolving operational-scale dynamics rather than wave-frequency seakeeping motions.

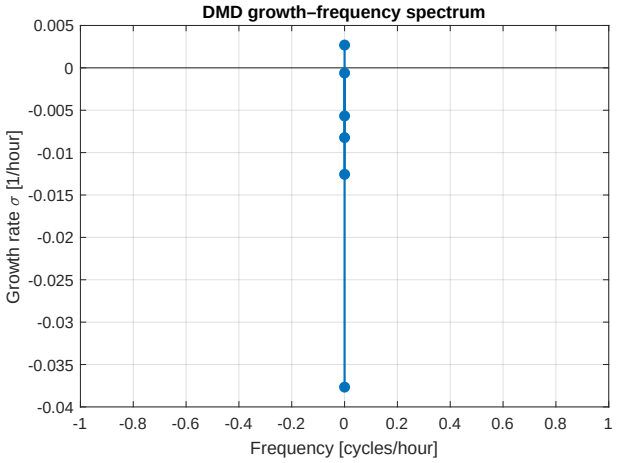


Fig. 7. Growth–frequency spectrum of DMD modes. Each point corresponds to one mode plotted by its continuous-time growth rate σ_k and frequency f_k. Negative σ_k values denote exponential decay.

While the growth-frequency diagram reveals the overall spread of modal behaviour, it is also useful to examine the evolution of growth rates as a function of mode index. Fig. 8 presents the continuous-time growth rates σ_k together with a smooth exponential fit of the form approximated by Eq. (22):

$$\sigma_k \approx ae^{-bk} + c \tag{22}$$

This captures the underlying trend in modal damping. The fitted curve highlights two key observations: 1) the dominant modes exhibit small negative growth rates that are consistent with slowly decaying drift dynamics, and 2) higher-index modes tend toward an asymptotic baseline c, indicating minimal additional dynamical structure at finer modal scales. The exponential trend, therefore, reflects the rapid attenuation of modal energy beyond the leading few modes, as expected for hourly-aggregated motion statistics.

The spatial structure of the DMD modes is shown in Fig. 9. Each column represents a DMD mode, and each row corresponds to one of the six DOFs. The colour indicates how

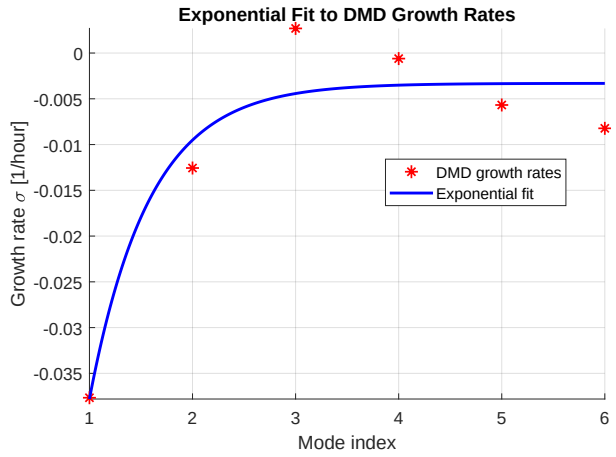


Fig. 8. Exponential fit to continuous-time growth rates obtained from the DMD eigenvalues. The fitted curve reveals the rapid decay of modal contributions and the dominance of a small number of slowly varying modes.

strongly that mode contributes to the motion in each DOF. For example, some modes have large amplitude in roll and yaw, while others project more strongly onto surge or sway. This means that each DMD mode captures a different pattern of coupled vessel motion. Such mode shapes are analogous to a data-driven response operator, revealing the dominant low-frequency coupling relationships present in the operational dataset.

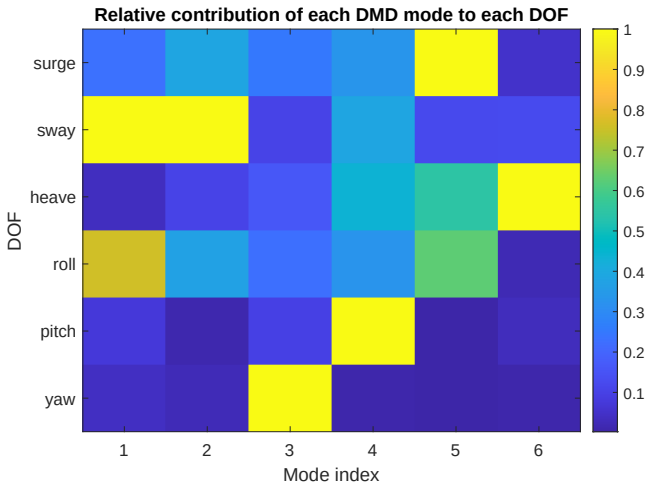


Fig. 9. Normalised magnitude of DMD modes across the six DOFs. Brighter values indicate stronger modal contribution in the corresponding DOF.

E. Time-Domain Prediction: DMD vs. Hybrid

Fig. 10 compares measured motions, the DMD-only baseline, and the DMD-TCN hybrid predictions for a representative test interval. Across all DOFs, the DMD model provides a smooth linear approximation of the short-horizon evolution, capturing the dominant decaying trend visible in the operational data. The hybrid model introduces nonlinear residual corrections, which can be seen as local adjustments to the DMD trajectories.

The quantitative results reveal that the hybrid model does not uniformly improve prediction accuracy relative to the DMD baseline. The DMD-only predictor achieves lower RMSE and MAE for most DOFs, except for heave, where the hybrid model yields a modest reduction in both error metrics. Surge shows comparable performance between the two models. These findings indicate that the dominant low-frequency drift dynamics contained in the hourly motion statistics are already well represented by the linear DMD operator, leaving limited predictable nonlinear structure for the TCN to learn at this sampling rate.

Correlation analysis provides additional insight into dynamical alignment. For heave and surge, the hybrid model maintains or slightly improves correlation with the actual signal. In contrast, for sway, roll, pitch, and yaw, correlation decreases under the hybrid model. This behaviour is consistent with the fact that these DOFs are influenced by nonlinear, unobserved physical drivers, such as mooring forces, cross-flow effects, heading control, or hydrostatic restoring, which are not in the feature set. Under limited physical observability, the TCN's residual corrections may reduce local bias but introduce high-frequency or phase-shifted components, thereby lowering overall correlation.

A DOF-specific summary is as follows:

- Surge: DMD and hybrid predictions track the measured trend reasonably well, with similar correlation. Hybrid RMSE is slightly higher, indicating that the residual correction does not significantly improve the well-captured linear behaviour.
- Sway: The hybrid introduces larger deviations from the measured signal, as reflected in higher RMSE/MAE and lower correlation. This suggests that sway dynamics, which depend strongly on cross-coupling with yaw and local hydrodynamic asymmetries, are not sufficiently represented in the available inputs.
- Heave: Heave is the only DOF for which the hybrid model consistently improves both RMSE and MAE. The TCN effectively refines the DMD baseline for this predominantly linear, well-observed mode, and also achieves the highest correlation among all DOFs.
- Roll and Pitch: Both DOFs remain challenging. Although the hybrid provides visually smoother adjustments, RMSE and MAE increase relative to DMD. These motions are governed by nonlinear restoring forces and second-order effects that cannot be inferred from the limited environmental and geometric features.
- Yaw: The most significant performance degradation occurs in yaw, where hybrid RMSE is substantially larger than DMD's. Yaw dynamics are susceptible to unmeasured quantities such as heading control inputs, mooring tensions, and harbour geometry, making residual correction unreliable at one-hour resolution.

Overall, the time-domain analysis highlights the limitations of nonlinear residual learning when applied to motion statistics sampled at hourly resolution. The DMD operator captures most of the predictable low-frequency structure. In contrast,

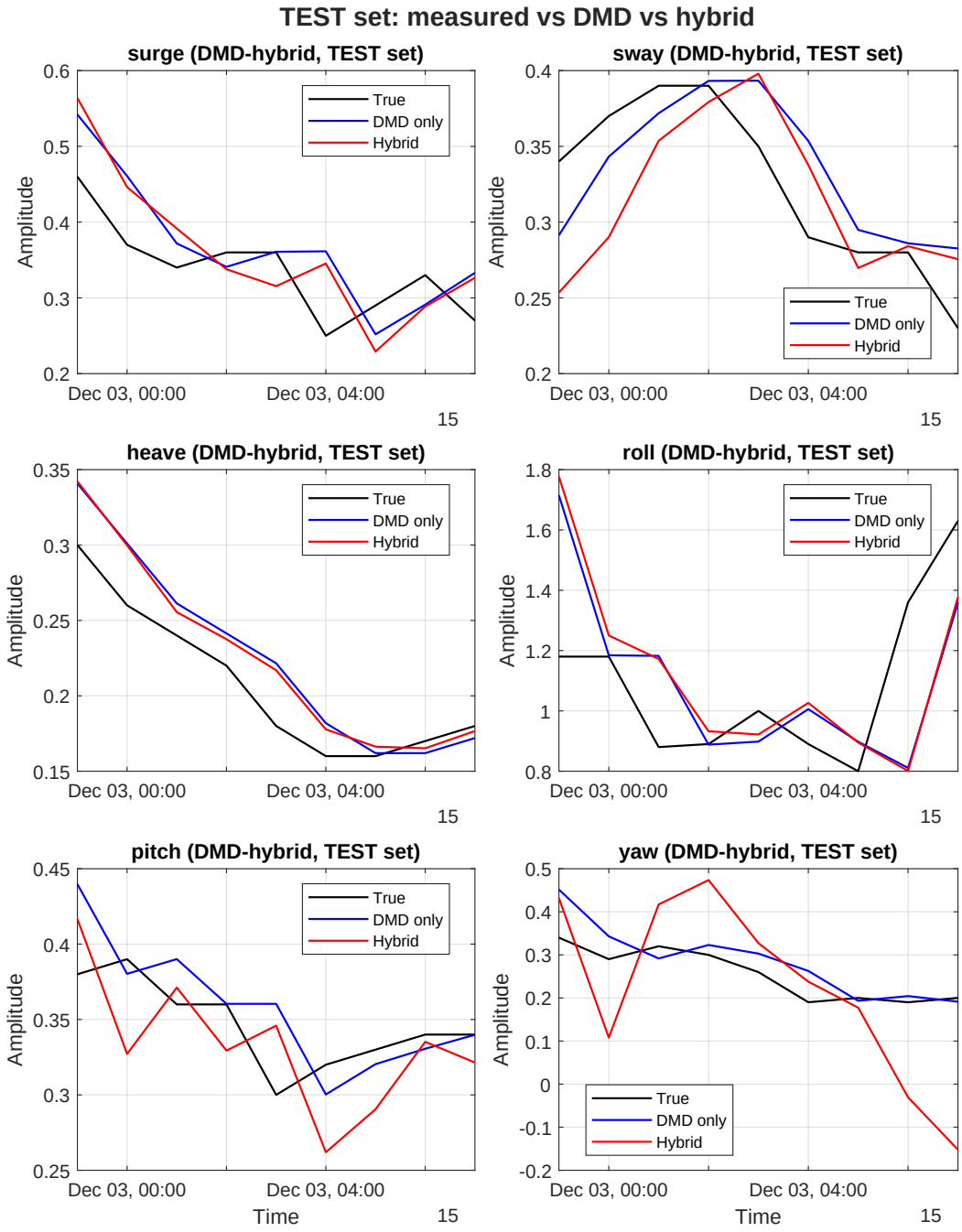


Fig. 10. One-step-ahead predictions on the test set: Measured motion (black), DMD-only baseline (blue), and DMD–TCN hybrid prediction (red) for surge, sway, heave, roll, pitch, and yaw.

the TCN lacks sufficient physical observability to accurately refine the linear baseline for the more nonlinear lateral and rotational DOFs. This underscores the need for richer physical inputs or higher-frequency measurements to unlock the full potential of hybrid physics–machine learning models for ship-motion prediction.

F. Ablation Study and Model Comparison

To isolate the contribution of each modelling component, a structured ablation study was conducted under identical preprocessing, windowing (12 h context, stride 1 h), and one-step-ahead evaluation ($H = 1$). All neural models were trained with Adam (learning rate 10^{-4}, gradient clipping 1.0) and evaluated using RMSE and MAE in physical units.

The evaluated models are grouped into three categories:

- Physics-based baselines
  - Persistence: $\hat{\mathbf{y}}(t{+}1) = \mathbf{y}(t)$
  - DMD-only: $\hat{\mathbf{y}}(t{+}1) = A_{\mathrm{DMD}}\mathbf{y}(t)$

TABLE IV. PHYSICS-ONLY BASELINES (RMSE IN PHYSICAL UNITS, $H = 1$)

Model	Surge	Sway	Heave	Roll	Pitch	Yaw
Persistence	0.0621	0.0359	0.0262	0.2982	0.0313	0.0502
DMD-only	0.0630	0.0371	0.0271	0.2956	0.0312	0.0517

TABLE V. PURE NEURAL PREDICTORS (RMSE IN PHYSICAL UNITS, $H = 1$)

Model	Surge	Sway	Heave	Roll	Pitch	Yaw
TCN-only	0.3772	0.1787	0.2366	0.3690	0.0613	0.4598
LSTM	0.3382	0.1815	0.2311	0.3717	0.0524	0.1777
GRU	0.3254	0.1691	0.2303	0.3710	0.0376	0.1622

- Pure neural predictors
  - TCN-only: $\hat{\mathbf{y}}(t{+}1) = f_\theta(X_t)$
  - LSTM
  - GRU
- Hybrid physics–ML models
  - Persistence+TCN
  - Proposed DMD+TCN

1) Physics baselines: Persistence remains a remarkably strong short-horizon predictor, particularly for surge and heave, where its performance is essentially indistinguishable from DMD. The DMD baseline slightly outperforms persistence in roll and pitch, indicating that a linear data-driven transfer operator captures part of the coupled restoring and wave-frequency structure in these modes (see Table IV).

Conversely, persistence performs better in sway and yaw, suggesting that, for these DOFs, the one-step dynamics are dominated by strong autocorrelation rather than by a structured linear evolution. Overall, the two physics-only baselines exhibit comparable performance at $H = 1$, reinforcing the observation that vessel motion at this temporal resolution operates in a near quasi-static regime.

2) Pure neural models: All pure neural models perform substantially worse than the physics-informed baselines at this horizon. This behaviour is consistent with a low-dynamic regime in which direct regression of absolute motion values is variance-dominated and sensitive to scale, particularly when the baseline signal already exhibits strong autocorrelation (see Table V).

3) Hybrid models: Hybridisation stabilises learning relative to pure neural predictors. Persistence+TCN improves surge prediction relative to persistence alone, while DMD+TCN provides modest gains in heave. However, at $H = 1$, improvements remain limited because the baseline already explains most of the one-step evolution (see Table VI).

4) Residual energy analysis: To quantify the extent to which the baseline models capture one-step vessel dynamics, computation of the residual-to-signal variance ratio for each DOF, as in Eq. (23) was performed:

TABLE VI. HYBRID MODELS (RMSE IN PHYSICAL UNITS, $H = 1$)

Model	Surge	Sway	Heave	Roll	Pitch	Yaw
Persistence + TCN	0.0528	0.0362	0.0265	0.3042	0.0356	0.1988
DMD + TCN	0.0662	0.0495	0.0251	0.3090	0.0391	0.1704

TABLE VII. RESIDUAL ENERGY RATIO ρ_d FOR PERSISTENCE AND DMD BASELINES ($H = 1$)

DOF	ρ_d (Persistence)	ρ_d (DMD)
Surge	0.8418	0.8472
Sway	0.4371	0.4478
Heave	0.1682	0.1620
Roll	1.3394	1.3139
Pitch	1.1650	1.1559
Yaw	0.5392	0.5374

$$\rho_d = \frac{\mathrm{Var}(y_d(t{+}1) - \hat{y}_d(t{+}1))}{\mathrm{Var}(y_d(t{+}1))}, \tag{23}$$

where, $\hat{y}_d(t{+}1)$ denotes either the persistence or DMD one-step prediction. The metric ρ_d measures the fraction of signal variance that remains unexplained by the baseline. Values $\rho_d \ll 1$ indicate that the baseline captures most short-horizon variability, whereas $\rho_d \geqslant 1$ implies that the baseline error variance exceeds the signal variance, reflecting substantial unmodelled structure.

Several important observations emerge (see Table VII):

- First, the residual energy ratios for persistence and DMD are nearly identical across all DOFs. This indicates that, at the one-step horizon considered, DMD does not significantly alter the fraction of explained variance relative to persistence. The vessel dynamics at $H = 1$ are therefore dominated by strong temporal autocorrelation rather than structured linear evolution beyond the previous state.
- Second, heave exhibits very small residual energy ($\rho_d \approx 0.16$), confirming that both baselines capture most of its short-horizon variability. This explains why hybrid residual learning yields only marginal improvements for this DOF.
- Third, sway and yaw display moderate residual energy ($\rho_d \approx 0.44$–0.54), suggesting that approximately half of the one-step variance is not explained by the linear baselines. However, the residual component appears only partially structured and therefore only modestly learnable from the available features.
- Finally, roll and pitch exhibit $\rho_d > 1$, indicating that baseline prediction error variance exceeds signal variance. This reflects the strong nonlinear restoring and coupling mechanisms governing these motions, which neither persistence nor DMD fully capture. Nevertheless, at $H = 1$, even this elevated residual energy does not translate into large hybrid accuracy gains, reinforcing the quasi-static nature of the short-horizon regime.

5) Interpretation: Taken together, the ablation and residual-energy analysis reveal three central conclusions:

- At short prediction horizons, ship motion operates in a strongly autocorrelated, near quasi-static regime in which persistence alone captures a substantial fraction of the variance.
- Linear data-driven modelling via DMD does not significantly change the residual energy structure at $H = 1$, although it offers a physically interpretable transfer-operator formulation.
- Hybridisation improves stability and interpretability, but accuracy gains remain limited when baseline residual energy is either small (heave) or only weakly structured (sway, yaw).

These findings suggest that the principal value of the proposed DMD–TCN framework lies not in dramatic short-horizon error reduction, but in its physically meaningful decomposition of vessel motion into a linear transfer component and a nonlinear residual correction. Larger performance gains are expected at longer prediction horizons, where residual energy and dynamical propagation effects become more pronounced.

VI. Discussion

The results demonstrate that the proposed hybrid DMD–TCN framework provides an interpretable decomposition of the vessel dynamics into linear low-frequency structure and nonlinear residual corrections. At the same time, the performance evaluation highlights substantial differences in the extent to which the hybrid model improves prediction across the six DOFs. This section interprets the observed behaviour in the context of ship hydrodynamics, data-driven system identification, and the limitations imposed by the available measurements.

A. Role of the DMD Baseline

DMD provides a data-driven linear surrogate for the slowly varying dynamics present in the hourly motion statistics. The eigenvalue spectrum (Fig. 6 to Fig. 8) shows that nearly all modes lie very close to the unit circle with negligible imaginary components, implying that the dominant dynamics are weakly damped and non-oscillatory. This is consistent with the low-pass nature of the processed signals, which excludes wave-frequency content.

The corresponding mode-shape matrix (see Fig. 9) reveals that surge, sway, and heave share broad modal contributions, while roll and yaw exhibit more concentrated mode affinities. This supports the interpretation of DMD as an operational-scale analogue of a response operator: it captures the primary drift-dominated component of the dynamics and the cross-DOF correlations that persist at low frequency.

B. Hybrid Learning vs. DMD

In contrast to expectations for residual learning, the hybrid model does not consistently improve one-step-ahead prediction accuracy across the six DOFs. The DMD-only baseline achieves lower RMSE and MAE for four DOFs (sway, roll, pitch, yaw), with surge showing similar performance for both models. Improvements appear only in the heave, where the hybrid model slightly reduces error and increases correlation with the ground truth.

This outcome indicates that, at hourly resolution, the linear operator recovered by DMD already captures most of the predictable variability in the low-frequency drift motions. The residual nonlinear effects that the TCN is intended to model are either weak in magnitude, mostly unobservable from the available features, or dominated by noise-like fluctuations that do not generalise well in a supervised learning setting. In such cases, the addition of nonlinear residuals may increase rather than decrease prediction error.

1) Hybridisation through the lens of the ablation study: The ablation study clarifies that hybridisation is not automatically beneficial in a low-dynamic, low-noise short-horizon regime. Pure neural predictors (Table V) perform worst overall, indicating that directly regressing absolute motions without an explicit physics prior is not robust here. This is consistent with the fact that the baseline signal is already highly predictable at $H = 1$, leaving limited structured information for a generic network to learn from noisy residual targets.

Hybrid models (Table VI) reduce the error gap relative to pure neural predictors and, in selected cases, offer modest gains. Persistence+TCN improves surge relative to persistence, and DMD+TCN improves heave relative to DMD. However, hybridisation degrades performance in several DOFs, most notably yaw for the DMD+TCN configuration. The residual-energy ratios help contextualise this behaviour: when ρ_d is small (heave), the residual is weak and relatively stable, so learning a correction is feasible but only yields modest improvements. When ρ_d is moderate (sway, yaw), the residual contains substantial energy. Still, this energy may not be learnable from the available features and cadence, making residual learning susceptible to fitting non-generalising structure. When $\rho_d > 1$ (roll, pitch), the baseline error variance exceeds the signal variance, which is a strong indicator that the baseline misses substantial dynamics; yet at hourly sampling, the missing components are plausibly driven by unobserved nonlinearities rather than predictable structure in the provided inputs, limiting the benefit of adding a residual network.

2) What DMD adds beyond persistence: Although persistence is a strong predictor, DMD remains useful as a data-driven modal analysis and transfer-operator surrogate. The mode-shape visualisation (see Fig. 9) further exposes cross-DOF coupling patterns embedded in the operational data, supporting the interpretation of DMD as a vessel-specific linear response operator learned directly from measurements.

Thus, even when DMD does not strongly outperform persistence in one-step RMSE/MAE, it offers interpretability by providing a compact linear dynamical model and a physically meaningful modal basis that persistence does not.

C. Challenges in Sway, Roll, Pitch, and Yaw

The degradation in hybrid accuracy for sway, roll, pitch, and yaw can be explained by the physics governing these DOFs. These motions depend on nonlinear and unmeasured factors such as:

- hydrostatic and viscous restoring forces (particularly in roll),
- cross-flow interactions and harbour geometry effects (sway, yaw),
- mooring stiffness and tension,
- rudder, thruster, or heading-control actions,
- asymmetric loading and trim.

Since none of these factors appears in the feature set, the TCN is forced to learn a nonlinear correction from incomplete physical information. As a result, the residuals can introduce high-frequency deviations, reducing both correlation and absolute accuracy. The DMD baseline, being purely linear, avoids such overfitting and therefore performs better in several DOFs.

D. Effectiveness of Causal Dilated TCNs

Although the hybrid model does not consistently reduce prediction error, the causal TCN component still provides architectural advantages. Dilated convolutions allow the network to aggregate information across a 12-hour context window, which is appropriate for modelling slow operational-scale changes. Furthermore, the residual formulation ensures numerical stability, since the TCN estimates only the deviation from the linear predictor.

However, the results show that long receptive fields are not sufficient when the relevant nonlinear forcing mechanisms are unobserved. Residual learning is most effective when the residual contains predictable structure; here, the residual for several DOFs is dominated by unmeasured or stochastic processes.

E. Physical Interpretation of the Hybrid Model

The decomposition of Eq. (18) remains valuable for interpreting the learned dynamics. The DMD prediction represents the best linear approximation of the vessel's drift behaviour under the observed operational conditions. The residual term reflects everything not captured by this linear model, including nonlinear vessel–environment interactions, unmodelled physics, and noise.

In heave, where the residual exhibits stable and interpretable structure, the hybrid model offers meaningful refinement. In contrast, for sway, roll, pitch, and yaw, the residual term lacks predictable structure at the sampling rate and within the available feature set, leading to reduced performance when added to the baseline.

F. Limitations and Opportunities for Improvement

The evaluation highlights several limitations of the present modelling framework:

- Hourly sampling suppresses significant wave- and manoeuvring-scale dynamics, reducing the amount of predictable nonlinear structure available to the TCN.
- Feature limitations prevent the model from resolving key forces and constraints driving lateral and rotational DOFs.
- Single-step prediction ($H = 1$) restricts the hybrid model from learning longer-term propagation of nonlinear effects.
- Residual noise in poorly observed DOFs may dominate the correction term, reducing accuracy.

Although the hybrid approach does not outperform DMD across all DOFs, it provides insights into which vessel motions contain learnable nonlinear structure under limited observability. These findings highlight clear directions for improvement, including the integration of richer physical inputs (mooring loads, heading data, control actions), higher-frequency measurements, or hybrid architectures that incorporate mechanistic priors.

VII. Conclusion

This study has presented a hybrid physics–machine learning framework for short-horizon prediction of six-DOF ship motions that combines a data-driven DMD baseline with a causal TCN residual model. The formulation provides an interpretable decomposition of the vessel dynamics into a linear low-frequency component and a nonlinear correction term, enabling physical interpretation of the learned behaviour.

The DMD operator served as a stable linear surrogate for the vessel's low-frequency drift dynamics. Its eigenvalues lie close to the unit circle, and the corresponding mode shapes revealed meaningful cross-DOF coupling patterns derived directly from operational data. This supports interpreting DMD as a data-driven analogue of a linearised response operator applicable at hourly resolution. However, the DMD approximation naturally underestimates nonlinear and asymmetric components of the motion, particularly in lateral and rotational DOFs.

The evaluation showed that the hybrid DMD–TCN model does not uniformly improve one-step prediction accuracy across all DOFs. While heave benefited from the nonlinear residual correction, resulting in reduced error and higher correlation compared with the DMD baseline, other DOFs (sway, roll, pitch, yaw) exhibited increased error under the hybrid model. These outcomes indicate that the residual nonlinear structure available at hourly sampling is limited and that key physical drivers of lateral and rotational motions, including mooring forces, control inputs, cross-flow hydrodynamics, and structural restoring effects, are not represented in the available feature set. Under such conditions, the TCN residual tends to model noise-like behaviour rather than systematic nonlinear corrections.

Despite these limitations, the hybrid formulation remains valuable. The decomposition highlights which DOFs contain learnable nonlinear structure under limited observability and which require richer physical inputs or higher temporal resolution to support effective hybrid learning. The results also demonstrate that interpretable physics-informed baselines such as DMD provide a useful scaffold for understanding and constraining data-driven models in operational marine environments.

Future work should explore integrating additional physical measurements (e.g., mooring loads, heading data, rudder or thruster inputs), higher-frequency motion records, and hybrid

architectures that embed hydrodynamic priors or mechanistic constraints. Multi-step sequence prediction, uncertainty quantification, and transfer learning across vessel types also represent promising research directions. Advances along these lines may significantly enhance the reliability and applicability of hybrid physics–machine learning methods for real-time vessel monitoring, digital twins, and maritime decision support systems.

VIII. Future Work

The present study demonstrates that hybrid physics–machine learning strategies offer an interpretable and modular approach to full-scale ship motion prediction, but also highlights key limitations arising from the available measurements and sampling resolution. Several research directions emerge that may strengthen the predictive capability of hybrid DMD–TCN models and improve their applicability in operational settings.

A. Integration of Stronger Physical Priors

While DMD provides a stable linear surrogate of low-frequency ship dynamics, its ability to represent nonlinear DOFs, particularly roll and yaw, is constrained by the lack of explicit physical structure in the operator. Future hybrid models may embed parametric or semi-empirical hydrodynamic components such as:

- restoring stiffness and metacentric-height effects,
- manoeuvring models and drift-force representations,
- simplified potential-flow or radiation–diffraction transfer terms.

Incorporating physics-based modules such as differentiable blocks could improve residual interpretability, reduce noise in the learned corrections, and provide a more accurate baseline for motions governed by nonlinear stability and cross-flow effects.

B. Augmented Environmental and Structural Feature Sets

Prediction accuracy for sway, roll, pitch, and yaw is limited by missing or weakly observed physical drivers in the current dataset. The inclusion of additional operational and environmental variables would provide richer observability of the underlying dynamics. Relevant extensions include:

- Mooring line tensions, fairlead geometry, and stiffness maps,
- Vessel heading, rudder or thruster activity, and autopilot signals,
- Local harbour geometry, wind shielding, and bathymetric features,
- Directional wave spectra rather than scalar H_s and T_p.

Such features would allow the hybrid model to represent asymmetric loading, cross-flow interactions, and nonlinear restoring behaviour that currently manifest as unpredictable residuals.

C. Multi-Step and Sequence Forecasting

This study considered only one-step-ahead prediction at hourly resolution. Extending the hybrid framework to multi-step or sequence-to-sequence forecasting may reveal longer-term dependencies and nonlinear propagation effects, especially for DOFs with strong dynamic coupling. Techniques such as recurrent rollout, scheduled sampling, or horizon-aware loss functions could help stabilise long-range predictions and reduce error accumulation.

D. Advanced Hybrid Architectures

Although the causal TCN architecture provides stability and long-range context, more expressive residual models may better capture the nonlinear structure present in the motions. Promising avenues include:

- Multi-scale dilation pathways to separate drift and rapid residual dynamics,
- Attention-based or adaptive convolutional modules to focus on informative time intervals,
- Hybrid state-space or latent-ODE models that incorporate learned dynamics alongside explicit physical constraints.

These architectures may recover structure in the residual term that standard dilated convolutions do not capture.

E. Cross-Vessel Generalisation and Transfer Learning

Evaluating the hybrid framework across different vessel classes, sea states, and operational regimes would clarify its robustness and generalisation ability. Transfer learning, domain adaptation, or meta-learning methods may enable efficient reuse of learned DMD operators or TCN weights, reducing the data requirements for new vessels or environments.

F. Uncertainty Quantification and Operational Deployment

For deployment in digital twins and decision support systems, uncertainty quantification is essential. Future work may incorporate Bayesian layers, ensembles, or probabilistic DMD and TCN formulations to provide confidence bounds on predictions. Such approaches would support risk-aware decision-making for operations such as mooring analysis, motion-compensation systems, and autonomous navigation.

Overall, these directions reflect the need for richer physical observability, higher temporal resolution, and more expressive hybrid architectures to unlock the full potential of physics-informed machine learning for operational ship-motion forecasting.

Acknowledgment

The authors would like to thank the Marine Research Unit at the Nelson Mandela University for donating the high-performance computer used in this research project.

REFERENCES

[1] O. M. Faltinsen, *Sea Loads on Ships and Offshore Structures*. Cambridge, U.K.: Cambridge Univ. Press, 1990.

[2] J. M. J. Journée and W. W. Massie, *Offshore Hydromechanics*. Delft, The Netherlands: Delft Univ. of Technology, 2001.

[3] M. Terziev, T. Tezdogan, and A. Incecik, "Scale effects and full-scale ship hydrodynamics: A review," *Ocean Eng.*, vol. 245, 110496, 2022.

[4] J. Jiao, S. Sun, and H. Ren, "Predictions of wave induced ship motions and loads by large-scale model measurement at sea and numerical analysis," *Brodogradnja*, vol. 67, no. 2, pp. 81–100, 2016.

[5] M. Kanazawa, S. Ueno, and K. Suzuki, "Physics–data cooperative ship motion prediction with onboard wave radar for safe operations," presented at the OMAE2023, Melbourne, Australia, 2023.

[6] R. E. Nielsen, D. Papageorgiou, L. Nalpantidis, B. T. Jensen, and M. Blanke, "Machine learning enhancement of manoeuvring prediction for ship Digital Twin using full-scale recordings," *Ocean Engineering*, vol. 257, p. 111579, 2022.

[7] A. Kawan, T. Mizoguchi, and Y. Hasegawa, "Data-driven prediction of ship motion by support vector regression," in *Proc. IEEE OCEANS*, Aberdeen, 2017, pp. 1–6.

[8] M. Diez, D. L. Villa, and E. Campana, "Data-driven prediction of ship motions in waves using machine learning models," *Applied Ocean Research*, vol. 100, 102202, 2020.

[9] K. M. Silva and K. J. Maki, "Data-driven system identification of 6-DoF ship motion in waves with neural networks," *Appl. Ocean Res.*, vol. 125, 103222, 2022.

[10] C. Schirmann, A. Fischer, and M. Buchner, "Data-driven prediction of ship motions using wave model outputs and onboard measurements," *Ocean Engineering*, vol. 245, 110505, 2022.

[11] G. Martius and C. H. Lampert, "Extrapolation and learning equations," arXiv:1610.02995, 2016.

[12] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving non-linear PDEs," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[13] E. S. Muckley, P. Baddoo, and B. Kutz, "Interpretable models for extrapolation in scientific machine learning," *Digital Discovery*, vol. 2, pp. 1443–1458, 2023.

[14] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 1999.

[15] L. Ljung, "System identification," *Automatica*, vol. 35, no. 3, pp. 419–435, 1999.

[16] W. Xu, K. J. Maki, and K. M. Silva, "A data-driven model for nonlinear marine dynamics," *Ocean Eng.*, vol. 236, 109469, 2021.

[17] D. D'Agostino, A. Serani, F. Stern, and M. Diez, "Time-series forecasting for ships maneuvering in waves via recurrent-type neural networks," *Journal of Ocean Engineering and Marine Energy*, vol. 8, no. 4, pp. 479–487, 2022.

[18] N. Papandreou, K. Belibassakis, and I. Apostolatos, "Hybrid data-driven and physics-based modelling for ship responses in seaways," *Journal of Marine Science and Engineering*, vol. 13, no. 1, 45, 2025.

[19] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*. Philadelphia, PA, USA: SIAM, 2016.

[20] C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson, "Spectral analysis of nonlinear flows," *J. Fluid Mech.*, vol. 641, pp. 115–127, 2009.

[21] S. Dogan, L. Magionesi, A. Serani, and E. F. Campana, "Dynamic mode decomposition for marine hydrodynamics: A review," *Appl. Ocean Res.*, vol. 101, 102243, 2020.

[22] M. Diez, A. Serani, E. F. Campana, and F. Stern, "Time-series forecasting of ships maneuvering in waves via dynamic mode decomposition," *J. Ocean Eng. Mar. Energy*, vol. 8, no. 4, pp. 471–478, 2022.

[23] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modelling," *arXiv preprint arXiv:1803.01271*, 2018.

[24] C. Lea, M. D. Flynn, R. Vidal, A. Reiter and G. Hager, "Temporal convolutional networks for action segmentation and detection," in *Proc. CVPR*, 2017.

[25] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, pp. 533–536, 1986.

[26] G. Ye, Q. Zhang, and C. Li, "Short-term prediction of ship roll motion using EMD-LSTM-SVR hybrid model," *Ocean Engineering*, vol. 251, 111113, 2022.

[27] J. Sun, Y. Zheng, and F. Liu, "A CFD-based data-driven reduced-order modelling method for predicting damaged ship motion in waves," *Applied Ocean Research*, vol. 136, 103391, 2023.

[28] G. La Ferlita, A. Russo, and F. Arena, "Assessment of data-driven models for ship motion prediction in irregular waves," *Ocean Engineering*, vol. 291, 119520, 2024.

[29] U. D. Nielsen, A. Buch, and H. K. Jensen, "On the use of machine learning for system identification of ship manoeuvring models," *Journal of Marine Science and Engineering*, vol. 10, no. 1, 15, 2022.

[30] A. Alvarellos, "Ship Movement Dataset," GitHub repository, 2019. [Online]. Available: https://github.com/aalvarell/ship-movement-dataset [Accessed: Nov. 20, 2025].

[31] Y. Kim *et al.*, "Seakeeping Committee Report," 27th Int. Towing Tank Conf. (ITTC), Copenhagen, Denmark, 2012.

[32] J. N. Newman, "Second-order slowly varying forces on vessels in irregular waves," in *Proc. Int. Symp. Dynamics of Marine Vehicles and Structures in Waves*, London, U.K., 1974.

[33] J. A. Pinkster, "Low frequency second order wave exciting forces on floating structures," Netherlands Ship Model Basin, Wageningen, The Netherlands, 1980.

SecureDML:An Intelligent Framework for Preventing Poisoning Attacks in Distributed Machine Learning Systems

Archa A. T, Kartheeban K
Computer Science and Engineering, Kalasalingam Academy of Research and Education, Krishnankovil, TamilNadu, India

***Abstract*—The security and protection of models in distributed machine learning (ML) systems require high emphasis on adversarial threats, including poisoning attacks. This study contains a complete framework that integrates different advanced techniques to monitor poison attacks and prevent such attacks for the effective functioning of machine learning systems. The proposed system integrates hybrid encryption for security, and a subsequent anomaly detection method using autoencoders. SHapley Additive exPlanations-based interpretability method is used to enhance model transparency. Hybrid encryption combines the RSA and AES methods to keep data and model parameters secret, and autoencoders provide effective identification of poisoning attack patterns through abnormal data observations. This method is implemented using multimodal datasets such as CIFAR 100 and AG News datasets. Finally,the effectiveness of this method can be evaluated using confusion matrix, comparison graphs. It works as a comprehensive solution that benefits various ML applications, such as healthcare, autonomous vehicles, Large Language Models, etc., for enhancing security along with integrity protection.**

***Keywords*—Poisoning attacks; SHapley Additive exPlanations (SHAP); anomaly detection; federated learning; autoencoders**

I. Introduction

Nowadays, machine learning (ML) offers computational intelligence for a variety of applications. It has many applications in the area of image processing, natural language processing, cybersecurity, robotics, autonomous vehicles, recommendation systems, LLM's etc. Training data has an impact on the accurate prediction of output in such systems. Sometimes, adversaries are coming up with new ways to train ML models, which causes more security risks to clients' data.

Machine learning model security [1] is currently under risk from data poisoning and evasion threats. By altering a model's data, methods, or hyperparameters during the model training phase, poisoning attacks give attackers the ability to manipulate the model's behavior. However, by altering the test sample, an evasion attack is conducted. Adversaries are able to create valid inputs that are invisible to people, yet cause models to forecast incorrectly. There have been multiple significant and effective adversarial attacks [2] against machine learning systems. Highly trained attackers are driven to alter the outcomes of machine learning models by extremely powerful motivations.

Gu, Tianyu,et al. [3] investigated security threats that emerge when training programs are outsourced because an attacker can implement hidden access points in neural networks called BadNets that function normally, but compromise the chosen input data from attackers. The security threat can be illustrated through a toy model that modifies the handwritten digit recognition system. The targeted misuse of a U.S. street sign recognition system directs stop signs to be read as speed limits whenever specific stickers appear on the sign. The integrity of the system remains compromised after network retraining as a backdoor consistently functions regardless of the trained task. Adding training data that either compromises the model's functionality or have an adverse impact on its performance is known as "data poisoning". The architecture of a poisoning attack is shown in Fig. 1 [2]. The model is forced to learn and make predictions for attackers to corrupt information in the training sample. This type of poisoning has spread to all ML applications and is not restricted to any specific domain. The purpose of the label flipping attack is to reverse machine learning detector predictions. Label poisoning is used to switch the labels of the current training data to test the resilience of the ML models.

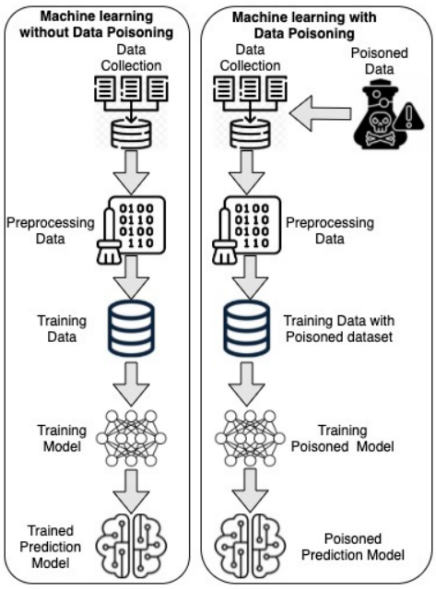


Fig. 1. ML architecture with and without poisoning [2].

Adversaries use training data manipulation to perform data poisoning attacks [4] against ChatGPT and similar LLMs [26] which leads to the inclusion of biases, vulnerabilities, or backdoors in the system. The attacks affect various points in the model training process and deployment stages which resulting in system outputs and behaviors becoming compromised. In

[7], the authors discussed the relevance of current and next-generation federated, secure, and privacy-preserving artificial intelligence methods in medical imaging. It also explores potential attack vectors and future prospects, emphasizing the need for advanced security measures in medical data processing. Transferable cleanlabel poisoning attacks were used in [8] to detect malicious training data without changing labels.

A. Types of Data Poisoning Attacks

1) Training data poisoning: During training, adversaries introduce modified information to manipulate or degrade the model capacity. Attackers embed toxic and false information or biased statements within publicly accessible datasets meant for fine-tuning.

2) Backdoor attacks [9]: The attack triggers a specific output whenever it is active through a hidden command placed in the training data. The model will produce harmful serving responses after receiving specific trigger phrases that appear during the training process on the poisoned data.

3) Exposure attacks: Attackers submit valuable information into the training sample data, which results in inaccurate predictions. The model unintentionally remembers numerous API keys and passwords together with private user communications.

4) Fine-tuning poisoning: During the adjustment of the external dataset, an opponent can develop deceptive data to modify the operational parameters of the model. The utilization of poisoned sentiment data in fine-tuning a chatbot may result in an unfair preference for particular subjects.

5) Prompt injection attacks (inference-time manipulation): The system receives adversarial prompts from the users, which alter the generated responses. The importance of the proposed model is to find direct applications within secure machine learning frameworks used to protect distributed and federated learning setups against data poisoning attacks. Integrity protection of training data is vital for critical sectors deploying AI models such as healthcare, finance, and cybersecurity.

While distributed ML security encompasses multiple dimensions, including encryption, privacy preservation, explainability, and multimodal robustness, this study focuses specifically on poisoning attack detection and mitigation. There are also practical deployability that is illustrated by this work, but are not stand-alone theoretical works.

The main contributions of this proposed work are as follows:

- Abnormal updates of a malicious client: To detect adversarial client updates in distributed learning, this study suggested representation-level anomaly scoring scheme, which is an autoencoder reconstruction. Contrary to distance-based defenses, including KRUM [36] which is based on geometric proximity, SecureDML learns the latent update behavior, making it possible to detect coordinated and low-magnitude stealth attacks.
- Anomaly-sensitive robust aggregation to mitigate: SecureDML uses adaptive anomaly scores in a weighted aggregation scheme, which eliminates the effect of adversarials, but averages update differently. Unlike FedAvg [37] variants, which do not involve adversarial discrimination, the presented approach has a high level of non-IID data distribution resistance and constrained Byzantine adaptation.
- Deterministic compatibility of decoupled robustness: This framework separates the mechanisms of poisoning resilience and privacy, and has no anomaly-masking effect as with DPFL-based [15] defenses, and is also compatible with secure communication and distributed deployment constraints.

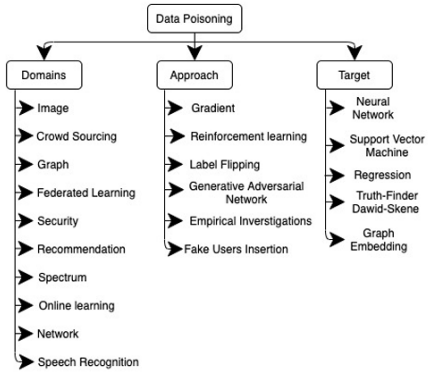


Fig. 2. Attack taxonomy [2].

II. Literature Review

There are different data poisoning attacks emerged in various attack domains. Existing literature on data poisoning attacks can be taxonomized in terms of attack domains, approach, and target (victim), as illustrated in Fig. 2. Recently, trending technologies, such as crowdsourcing and federated learning are always vulnerable, as the authenticity of individual data can never be verified. The most recent victims of poisoning attacks have spread in the security, network, speech recognition, and healthcare. The existing data poisoning approaches have targeted almost all the machine learning algorithms, ranging from traditional algorithms like regression to modern deep neural network architectures.

Researchers and adversaries use a variety of tactics to change data in data poisoning, including the two major data poisoning attacks such as Label Flipping and Feature Poisoning(FP). Lin et al. [10] assessed the effectiveness of LF attacks and suggested a defense mechanism using two genuine datasets from the UCI repository such as MNIST [11], and Spambase [12], which are frequently used benchmarks for classification tasks.Munoz et al. [13]trained a CNN on a labelled dataset that was harmless and manipulated using an LF attack on the MNIST dataset. The researchers determined that adding poisoned training points increased the number of classification mistakes.

A Multiclass Logistic Regression classifier increased the classification error during random LF attacks while replacing the CNN. Fisher et al. [14] proposed approaches to support labeling constraints whereby hackers can apply predefined prediction transformations versus outputting incorrect

results altogether. The repetition of experiments on different datasets, including CIFAR-10, along with a reduced version of ImageNet, demonstrated the effectiveness of the proposed technique. Gradient ascent strategy is used in [1], in which the gradient is computed based on the properties of the optimal solution of SVMs. This method can be kernelized and enables the attack to be constructed in the input space even for nonlinear kernels.

Homomorphic-based transfer learning [15] is used to protect the client's privacy in training tasks by encrypting client data. A detection mechanism is carried out to locate sizable attack groups and their tactics, and to infer their capabilities. The ability of Robust-DPFL to concurrently guarantee task correctness, model robustness, and privacy protection is confirmed by extensive testing on three benchmark datasets: MNIST, FEMNIST, and CIFAR 10. Robust-DPFL updates the global model on detected clean gradients after first differentiating poisoned gradients from clean gradients using their unfair patterns in the components.

Paudice et al. [16] investigated how attackers manipulate training labels through flipping attacks because they weaken the model's performance. Label sanitization, detection and correction methods have been proposed by the authors to minimize the damage from attacks where attackers modify training labels intentionally. This study details the significant accuracy reduction triggered by label flipping attacks, which primarily affects binary classification tasks. To fight such scenarios, the authors introduced a labeling sanitization system.

The k-nearest neighbors (k-NN) method acts as the foundation for this technique, which evaluates suspicious data points through comparison with their nearest neighbors. The algorithm detects infected samples by performing a check that results in either elimination or correction of those samples. The defense mechanism specifically targets label flipping attacks but cannot prevent other poisoning attacks, such as backdoor attacks [17] or gradient manipulation in the same way. The k-NN sanitization method depends on the data set characteristics. In data sets with numerous dimensions, the k-NN algorithm shows reduced effectiveness, whereas the sanitization process leads to possible clean data removal which minimally affects the model prediction accuracy.

Luis Mu˜noz-Gonz´alez et al. [13] evaluated the vulnerability of deep learning model to data poisoning attacks through their proposed novel execution method. The authors extended data poisoning attacks from binary classifiers to multiclass classifiers while demonstrating wider threat applicability. The authors created a back-gradient optimization method that uses automatic differentiation to compute efficient gradients, allowing adversaries to create deceptive training samples that affect multiple models within neural networks and deep learning systems. A new attack method has been successfully applied across different applications, including both spam filtering systems and malware detection systems, and handwritten digit recognition systems. This research demonstrates that adversaries can create transferable attacks through adversarial training, which enhances the destructive capacity of such attacks and defense strategies are not addressed in this study.

Different types of poisoning attacks and their prevention strategies [18] are discussed to improve the efficiency of ML models.Some real time poisoning attack scenarios and detection methods [19] to determine the relevance of using secure ML systems. Jodayree,et al. [5] proposed a robust prevention scheme to enhance the security of federated learning. Three algorithms are proposed here to enable the creation of encrypted and decrypted verification keys for federated learning data files on a training node. Malicious participants are eliminated by the federated learning server and prevent backdoor attacks with the help of an encrypted verification scheme.It can be useful only on low-processing devices. Differentially Private Federated Learning is discussed to perturb the communicated data via independent random noise to pose challenges to user privacy identification.

Ali,et al. [20] proposed two-step defense methods such as global filtering, selffiltering, and adaptive threshold strategy.To address more complex attack patterns, there is a need for various other anomaly detection methods. The need for security of FL systems in IOHT applications [21]. Jia,et al. [22] proposed a generic poison detection approach named CodeDetector to automatically detect poison samples in a suspicious dataset. This study discussed security issues and the need for a more defensive layer.

A. Explainable AI (XAI) and SHAP

A model's predictions become transparent through explainable AI methods (XAI) which ensures that the predictions can be trusted and interpreted. SHAP [23]is a commonly used XAI method that assigns importance values to input variables according to their influence on the model outputs. Anomaly detection benefits from SHAP values because they reveal important influencing features when flagging irregular feature contributions during decision-making processes. The theoretical basis of SHAP relies on Shapley values from cooperative game theory, which provides equitable feature contribution assessments. Lundberg,et al. [24]created an approach that computes SHAP values effectively for complex ML models. Learning professionals used SHAP to demonstrate its success in interpreting tree-based models [24] together with deep learning architectures and alternative machine learning techniques.

SHAP was employed to identify poisoning attacks in adversarial machine learning through the identification of inconsistent feature attributions. Molnar (2022) explained that the interpretability technique SHAP functions to enhance security because it reveals manipulated features present in datasets. SHAP enables the strong protection of ML models against adversarial threats because it delivers clear explanations of individual predictions. Table I shows some recent methods used to detect and prevent poisoning attacks.

III. Problem Statement

Data poisoning corrupts training data, which degrades model functionality and establishes unauthorized access points. These attacks create major threats to datasets containing both images and texts because they specifically target distributed systems. This leads to undesirable outcomes that affect the performance of the distributed ML systems. To handle such systems efficiently, a better data poisoning attack detection and prevention method is required.

TABLE I. SOME RECENT METHODS USED TO DETECT AND PREVENT POISONING ATTACKS AND THEIR CHALLENGES.

Year	Methods Used	Challenges
2025	Global model aggregation algorithm designed to mitigate poisoning attacks [39]	Need more defensive method to address jamming and poisoning attacks in wireless FL networks.
2025	Two-step defense methods,such as global filtering, self-filtering, and adaptive threshold [20]	In order to address more complex attack patterns, there is a need for various other anomaly detection methods to be used.
2024	A generic poison detection approach named CodeDetector to automatically detect poison samples in a suspicious dataset [22]	Security issues, there is a need for a defense layer as extended work.
2023	An encrypted verification scheme in the federated learning model [5], preventing poisoning attacks without requiring specific attack detection programming	Useful only for only low processing devices. Difficult to handle complex data.

Assume that the clean training dataset be:

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{n}$$

Calculate the model parameters through empirical risk minimization:

$$\theta^* = \arg\min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(f_\theta(x_i), y_i)$$

A. Formulation of Poisoning Attack

m malicious samples are injected by adversary:

$\mathcal{D}_p = \{(x_j^p, y_j^p)\}_{j=1}^{m}$.

Then, the poisoned dataset becomes:

$$\mathcal{P}' = \mathcal{P} \cup \mathcal{P}_d$$

The attacked model parameters are calculated as:

$$\theta_p^* = \arg\min_{\theta} \frac{1}{n+m} \sum_{(x,y)\in\mathcal{D}'} \mathcal{L}(f_\theta(x), y)$$

B. Security Goal

The machine learning system is robust, if:

$$\|\theta_p^* - \theta^*\| \leq \epsilon$$

where, the poisoning ratio is:

$$\alpha = \frac{m}{m+n}$$

IV. ADVERSARY ASSUMPTIONS AND THREAT MODEL

A. Distributed Learning Environment

Consider a distributed learning environment that consists of a central server S, which coordinates different participating clients. Each client contains a private local dataset. A local optimization occurs on its own dataset, and each client sends the updates on the model to the server which summarizes them to create the following global model.

B. Adversary Assumption

Consider a limited Byzantine adversary that may fail at most fraction. Bad users can organize and interfere with their local information or sent updates. The adversary acts in either a white-box environment (information about model architecture, global parameters and aggregation rule) or a black-box environment (only has access to compromised local information); unless or otherwise, the more powerful white-box assumption is used.

The attack budget is limited by local data poisoning rate and an update magnitude to have realistic adversarial capability. We measure availability attacks, targeted attacks and backdoor attacks.

The opponent is not allowed to attack the central server, corrupt the mechanism of aggregation, or corrupt validation or test data or to surpass the specified fraction of attack clients. The outlining framework identifies abnormal updates of clients, and suppresses their effects by using powerful aggregation, which ensures that convergence is stable with the influence of attacks being limited under these adversarial conditions.

C. System Model

Here, the model is a federated learning system where all participants are synchronized and where the weighted aggregation of models is done by a central server at every round. Clients calculate local updates on personal data and send this to the server where it is aggregated. The proposed method is applied in the server side which identifies and blocks suspicious updates prior to parameter fusion.

V. PROPOSED METHODS

This method includes different sequential steps for operation. The data ingestion phase enables various clients to send their processed data to the system. Transfer learning-based systems have been used to detect GAN poisoning attacks. Multimodal data, such as image,text data enters the processing system through ViT and BERT [26] before being efficiently processed. Encryption is performed by AES [6,25,27] using RSA Signing method, which provides data security during communication and storage. The distributed learning method allows clients to perform local training and receive securely encrypted information through security protocols. The system detects anomalies through autoencoder. Feature changes can be identified by the SHAP Explainability system when the system explains the prediction models.

The system combines different advanced methods to build a distributed machine learning (ML) [29] model training framework that delivers strong performance and security features

while adapting to changing conditions.This system comprises four main processes, including data ingestion, hybrid RSA and AES encryption-decryption, autoencoder anomaly detection, and SHAP-based interpretability functions.

The operation of proposed system begins with a data ingestion stage. The initial step combines the task of collecting raw data from several sources with the preprocessing operations. Multiple data sources are feed into the system to produce standardized, clean data that are ready for machine learning operations. The system features multiple data type compatibility along with various data source compatibility to accommodate the needs of different distributed environments.

For efficient operation of the ML algorithm, data ingestion methods need to maintain both quality and data consistency because these criteria establish a foundation for the model.The proposed system shields data through a dual encryption solution that joins the Rivest-Shamir-Adleman RSA [30] keys with AES[24] (Advanced Encryption Standard). This approach leverages the strengths of both encryption methods and utilizes RSA encryption for safe encryption key transfers between its components. Data encryption under RSA requires two cryptographic keys that function in public and private form for security. Encryption takes place using the public key but the private key is responsible for the decryption operations. The system maintains secure information transfer of sensitive content such as model parameters and distributed node data through its configuration that defends against unauthorized access.

Large amounts of data undergo secure encryption and decryption using AES (Symmetric Encryption). The encryption process using AES occurs with shared secret keys, which leads to quicker and more efficient encryption of large datasets compared to RSA encryption. During the training and inferential processes, AES operates on bulk data to secure data during transmission and storage.

The anomaly detection capability plays a vital role in such systems because distributed ML environments handle inputs from multiple diverse sources with different quality levels. The system utilizes autoencoders which are unsupervised neural networks, to identify anomalies within the input data. Training an autoencoder enables the network to master a reduced format of data while simultaneously building its reconstruction. Anomalies are detected by measuring the reconstruction error, which establishes the difference between the initial input data and the recreated version. Any data point exceeding the specified thresholds in its reconstruction error was identified as an anomaly. Distributed ML systems identify anomalies as problems, including data attacks and sensor faults that potentially harm model functionality or integrity. The system tracks anomalies in real-time to modify its operation based on changing data patterns while spotting problems that could disrupt the proper model training.

Machine learning model interpretation and trustworthiness increase through the utilization of SHAP (SHapley Additive exPlanations) within the system. The explanation system SHAP performs an unbiased analysis of separate feature impacts on the predictive outcomes for any model type. SHAP values are used to understand model predictions because this interpretability method reveals specific features that influenced

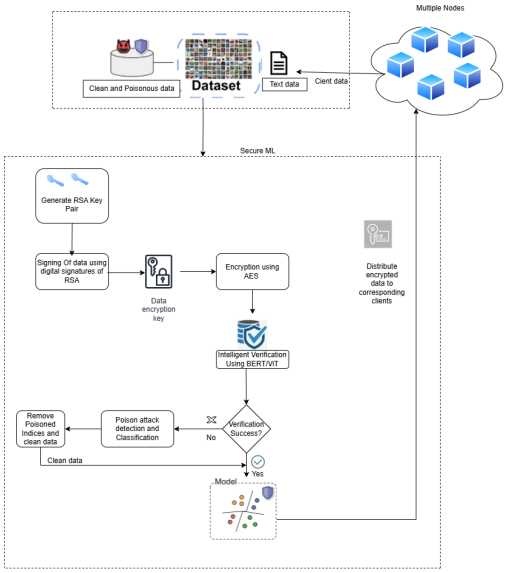


Fig. 3. Secure DML system.

outcomes the most. The interpretability of distributed ML systems becomes critical throughout multiple node training because it enables better debugging, performance improvements, and compliant privacy regulation implementations.

The incorporation of SHAP into the proposed system provides stakeholders with transparent decision-making capabilities to assess the prediction reliability and fairness of the model. The capacity to provide explanations makes this approach essential for industries that must follow explainability regulations in the field of healthcare and finance. The proposed system (Secure DML system) in Fig. 3 consists of data ingestion, encryption, and decryption using the hybrid approach of the RSA and AES methods, anomaly detection using autoencoders and integrity verification using BERT,ViT and SHAP-based interpretability to ensure robust and adaptive model training in distributed ML systems. The detailed steps are shown in Fig. 4.

VI. Experimental Evaluation

The experiments were performed in a distributed machine learning environment using Google Colab while utilizing GPUs to enhance the computational speed. The evaluation was conducted on two multimodal datasets: CIFAR-100 for images and AG News for texts. The development was performed through Python using TensorFlow along with PyTorch to accomplish AES-RSA encryption and BERT [31,32]verification through cryptographic libraries.

A. Datasets

The CIFAR-100 [33] benchmark includes 60,000 images divided into 100 distinct categories. The news articles in AG News [28] consist of 120,000 items divided across four classes. Here,10,000 samples were selected from which 3,000 consisted of clean data points and 7,000 contained poisoned samples with diverse attack methods such as misclassification, label flipping, backdoor, and perturbations. The level of poisoning

Algorithm 1 Secure Distributed ML System

Input: Clients $\{C_1, C_2, ..., C_N\}$, Global Model θ, Streaming Data $\mathcal{D}_t^{stream}$
Output: Securely Trained Global Model θ^* with Updates
Step 1: System Initialization
Generate RSA Key Pair (PK, SK)
Generate AES Key K_{AES}
for each client C_i **do**
  Encrypt K_{AES} with RSA: $K_{AES}^{enc} = Encrypt(PK, K_{AES})$
  Securely store K_{AES}^{enc}
  Initialize local model $\theta_i \leftarrow \theta$
end for
Step 2: Data Ingestion and Feature Extraction
for each client C_i receiving new data stream $\mathcal{D}_t^{stream}$ **do**
  Extract features from images using ViT and text using BERT
  Normalize and encrypt features using AES: $E_i = Encrypt(K_{AES}, X_i)$
end for
Step 3: Federated Learning with Secure Updates
for each global round $t = 1, 2, ..., T$ **do**
  for each client C_i processing real-time data **do**
    Train local model $\theta_{i,t}$ on encrypted data E_i
    Compute anomaly score $A_i = ||X_i - \hat{X}_i||^2$ using Autoencoder
    if A_i exceeds threshold **then**
      Mark data as suspicious, reject update
    else
      Send $\theta_{i,t}$ update to server
    end if
  end for
  Server aggregates valid updates: $\theta_{t+1} = \frac{1}{N}\sum_{i=1}^{N}\theta_{i,t}$
end for
Step 4: Model Adaptation
while system is running **do**
  Receive new client data
  Compute SHAP values for interpretability
  if SHAP indicates feature drift or anomaly **then**
    Adjust model training with adaptive learning rate
  end if
  Update global model θ^* dynamically
  Deploy updated model for inference
end while
End of Algorithm

Fig. 4. Steps for secure DML system.

is on the client level. A fraction of clients are assumed to be compromised, and inject malicious updates every time they undergo local training. The assumptions of Byzantine federated learning assume 30% of the clients are malicious. A variety of poisoning attack strategies have been employed to evaluate this approach. An attack strategy based on GAN provides data poisoning through adversarial samples that deceive classification models. The implementation of specific triggers through Backdoor Attacks allows predictions to be manipulated in their target directions. The process of label flipping was used to create inconsistencies within the dataset. Slight modifications of the data cause perturbation-based attacks. AES-256 [34] operated as the encryption method for data protection, whereas key exchange and digital signature operations relied on RSA-4096.

B. Results

The effectiveness of this cryptographic method can be implemented by comparing the method with benchmark methods. Fig. 5 shows the classification report and confusion matrix on multiple clients without a defense mechanism. As the number of clients increases,the attack detection rate decreases, as shown in Fig. 6. This demonstrates the need for a defense layer in ML systems. Adding a hybrid defense mechanism helps to secure sensitive data. The classification methods fail to provide interpretable explanations thus making it complex to understand decision-making reasons. This system integrates SHapley Additive exPlanations (SHAP) to generate detailed explanations for detected poisonous data samples. Through SHAP analysis, consistent features are discovered which led to the detection of anomalies among different clients.

The scatterplot showing anomalies are shown in Fig. 7 and the anomaly detection on multiple clients is shown in Fig. 8.

The BERT-ViT hybrid model functions as an integrity

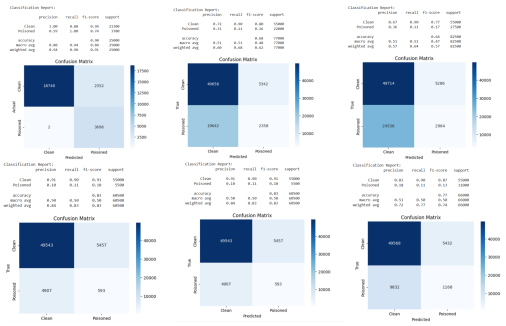


Fig. 5. Confusion matrix without having defense mechanism.

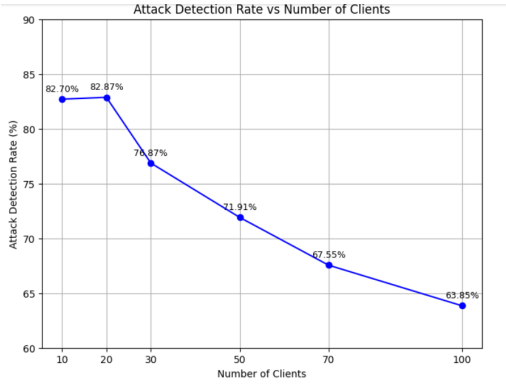


Fig. 6. Attack Detection rate without having defense mechanism.

verification system that verifies the authenticity of the data while detecting poisonous samples. The Attack Success Rates based on an increasing number of poisonous points are shown in Fig. 9. ASR can be calculated as:

$$ASR = \frac{Npsuccess}{Np} * 100 \quad (1)$$

where, $Npsuccess$ denotes the number of data samples and

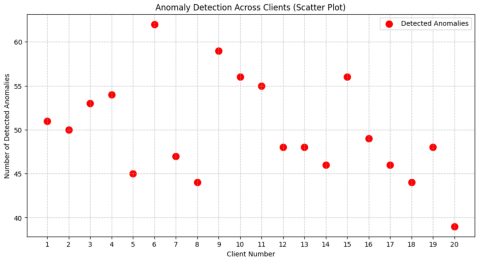


Fig. 7. Scatterplot showing anomalies.

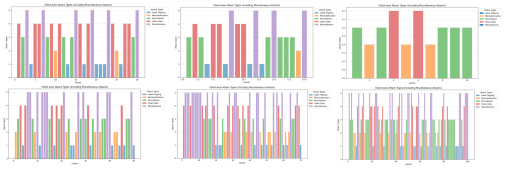


Fig. 8. Different poisoning attacks on muiltiple clients.

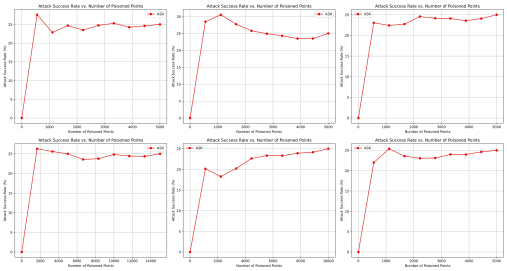


Fig. 9. Attack success rates.

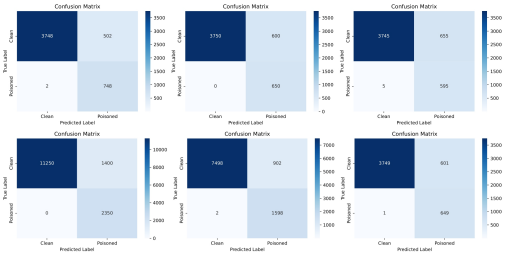


Fig. 10. Confusion matrix.

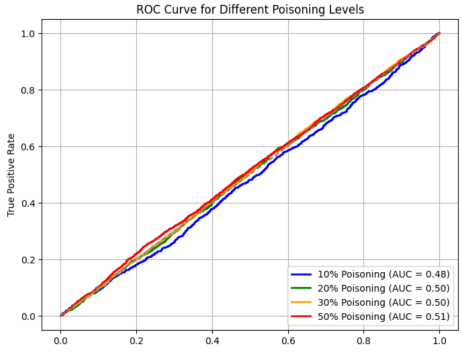


Fig. 11. ROC curve.



Fig. 12. Attack detection rate.

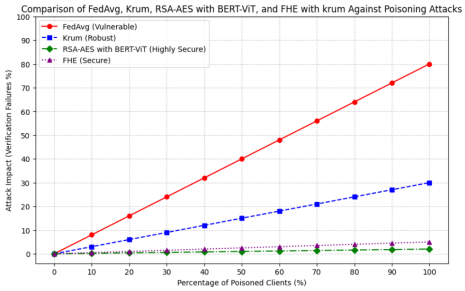


Fig. 13. Comparison with other methods.

Np denotes total no. of poisoned points.

The SHAP-based classification demonstrated better performance based on its high precision and recall values, as confirmed by the ROC curve analysis. The SHAP analysis delivered saturated results through its confusion matrix where poisonous and clean data categories showed distinct separation.

The effectiveness of SHAP based methods can be achieved using a confusion matrix which is shown in Fig. 10.

The ROC curve in Fig. 11 shows the variation of AUC values across different levels of poisoning.

The poisoning attack detection rate across an increased number of clients (number of clients 10,20,30,..100) are shown in Fig. 12. As the number of client increases, the rate of attack detection becomes relatively stable, indicating that the system is highly robust. The detection performance of the system was high, which demonstrates the scalability of the system.

VII. Discussion

This proposed defense method is compared with other benchmark methods such as KRUM [36],FedAvg [38], and FHE [37] methods.The verification failures are less in the proposed method as compared with other methods.This indicating that the hybrid defense method is highly secure.The analysis graph is shown in Fig. 13.

VIII. Applications and Future Enhancements

The proposed method is used to maintain secure federated learning models [35] by protecting them from distributed machine learning attacks. Hybrid RSA-AES encryption allows the approach to provide protected model updates thereby proving ideal for sensitive applications, including healthcare and finance. The detection capabilities of the system are improved through nDPI-based poisoned packet detection in network environments. The system permits the immediate disclosure of adversarial network traffic while Streamlit operates an intrusion detection dashboard that delivers an easy-touse interface to security analysts to track potential threats in real time and network intrusion detection systems are enhanced through the analysis of poisoned packets by combining nDPI with deep learning classifiers.

Data manipulation must not impact AI systems operating in finance, autonomous systems, or healthcare applications. This system can be applied to the management of IoT system security and edge environments. The system can be expanded to perform better multimodal verification when applied to security applications such as Deepfake detection in video environments. The system's resistance to adversarial modifications in voice-based AI systems increases when the integration of Whisper or Wav2Vec models is included. This system is highly secure, achieves privacy and protection, and is scalable for

moderate users. Its explainability feature helps to detect and interpret anomalous features. Although the system achieves better performance, the computation overhead is still an issue, which is further enhanced in the future.

IX. Conclusion

Security threats from poisoning attacks create major problems in the performance and integrity of distributed machine learning models. The attackers inject deceptive or harmful information into model training, which damages performance and quality, and generates incorrect predictions. This system offers a unified solution that uses various cutting-edge methods to detect and protect against poisoning. The proposed system offers security through robust distributed ML models, enhancement, and protective measures to combat poisoning attacks. The system utilizes anomaly detection together with secure encryption and model interpretability to discover suspicious data while stopping malicious inputs, thereby maintaining the uninterrupted integrity of distributed models. Thus, distributed ML systems are secure against adversarial attacks and maintain good performance.

X. Statements and Declaration

There is no potential conflict of interest reported by the author(s).

References

[1] Biggio, Battista, Blaine Nelson, and Pavel Laskov. "Poisoning attacks against support vector machines." arXiv preprint arXiv:1206.6389 (2012).

[2] Aryal, Kshitiz, Maanak Gupta, and Mahmoud Abdelsalam. "Analysis of label-flip poisoning attack on machine learning based malware detector." In 2022 IEEE International Conference on Big Data (Big Data), pp. 4236-4245. IEEE, 2022.

[3] Gu, Tianyu, Brendan Dolan-Gavitt, and Siddharth Garg. "Badnets: Identifying vulnerabilities in the machine learning model supply chain." arXiv preprint arXiv:1708.06733 (2017).

[4] Archa, A. T., and K. Kartheeban. "A Review on Privacy Enhanced Distributed ML Against Poisoning Attacks." In International Conference on Cyber Security, Privacy in Communication Networks, pp. 173-186. Singapore: Springer Nature Singapore, 2023.

[5] Jodayree, Mahdee, Wenbo He, and Ryszard Janicki. "Preventing image data poisoning attacks in federated machine learning by an encrypted verification key." Procedia Computer Science 225 (2023): 2723-2732.

[6] Zodpe, Harshali, and Ashok Sapkal. "An efficient AES implementation using FPGA with enhanced security features." Journal of King Saud University-Engineering Sciences 32, no. 2 (2020): 115-122.

[7] Kaissis, Georgios A., Marcus R. Makowski, Daniel Rückert, and Rickmer F. Braren. "Secure, privacy-preserving and federated machine learning in medical imaging." Nature Machine Intelligence 2, no. 6 (2020): 305-311.

[8] Zhu, Chen, W. Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. "Transferable clean-label poisoning attacks on deep neural nets." In International conference on machine learning, pp. 7614-7623. PMLR, 2019.

[9] Zhang, Jie, Chen Dongdong, Qidong Huang, Jing Liao, Weiming Zhang, Huamin Feng, Gang Hua, and Nenghai Yu. "Poison ink: Robust and invisible backdoor attack." IEEE Transactions on Image Processing 31 (2022): 5691-5705.

[10] Lin, Jing, Long Dang, Mohamed Rahouti, and Kaiqi Xiong. "ML attack models: Adversarial attacks and data poisoning attacks." arXiv preprint arXiv:2112.02797 (2021).

[11] Deng, Li. "The mnist database of handwritten digit images for machine learning research [best of the web]." IEEE signal processing magazine 29, no. 6 (2012): 141-142.

[12] Hopkins, Mark, Erik Reeber, George Forman, and Jaap Suermondt. "Spambase, UCI Machine Learning Repository (1999)." DOI: https://doi.org/10.24432/C53G6X (2024).

[13] Muñoz-González, Luis, et al. "Towards poisoning of deep learning algorithms with back-gradient optimization." Proceedings of the 10th ACM workshop on artificial intelligence and security. 2017.

[14] Fisher, Ronald Aylmer. "Iris. UCI machine learning repository." DOI: https://doi. org/10.24432/C56C76 (1988).

[15] Lee, Seewoo, Garam Lee, Jung Woo Kim, Junbum Shin, and Mun-Kyu Lee. "HETAL: Efficient privacy-preserving transfer learning with homomorphic encryption." In International conference on machine learning, pp. 19010-19035. PMLR, 2023.

[16] Paudice, Andrea, Luis Muñoz-González, and Emil C. Lupu. "Label sanitization against label flipping poisoning attacks." In Joint European conference on machine learning and knowledge discovery in databases, pp. 5-15. Cham: Springer International Publishing, 2018.

[17] Chen, Xinyun, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. "Targeted backdoor attacks on deep learning systems using data poisoning." arXiv preprint arXiv:1712.05526 (2017).

[18] Archa, A. T., and K. Kartheeban. "Real time poisoning attacks and privacy strategies on machine learning systems." In 2024 3rd International Conference on Sentiment Analysis and Deep Learning (ICSADL), pp. 183-189. IEEE, 2024.

[19] AT, Archa, and K. Kartheeban. "SecureTransfer: A Transfer Learning Based Poison Attack Detection in ML Systems." International Journal of Advanced Computer Science & Applications 15, no. 7 (2024).

[20] Ali, Yasir, Kyung Hyun Han, Abdul Majeed, Joon S. Lim, and Seong Oun Hwang. "An Optimal Two-Step Approach for Defense Against Poisoning Attacks in Federated Learning." IEEE Access (2025).

[21] Coelho, Kristtopher K., Michele Nogueira, Alex B. Vieira, Edelberto F. Silva, and Jose Augusto M. Nacif. "A survey on federated learning for security and privacy in healthcare applications." Computer Communications 207 (2023): 113-127.

[22] Li, Jia, Zhuo Li, HuangZhao Zhang, Ge Li, Zhi Jin, Xing Hu, and Xin Xia. "Poison attack and poison detection on deep source code processing models." ACM Transactions on Software Engineering and Methodology 33, no. 3 (2024): 1-31.

[23] Molnar, C. "Interpretable machine learning, 2nd edn. christophm. github. io/interpretable-ml-book." (2022).

[24] Lundberg, Scott M., and Su-In Lee. "A unified approach to interpreting model predictions." Advances in neural information processing systems 30 (2017).

[25] Lalem, Farid, Abdelkader Laouid, Mostefa Kara, Mohammed Al-Khalidi, and Amna Eleyan. "A novel digital signature scheme for advanced asymmetric encryption techniques." Applied Sciences 13, no. 8 (2023): 5172.

[26] Ferrag, Mohamed Amine, Mthandazo Ndhlovu, Norbert Tihanyi, Lucas C. Cordeiro, Merouane Debbah, Thierry Lestable, and Narinderjit Singh Thandi. "Revolutionizing cyber threat detection with large language models: A privacy-preserving bert-based lightweight model for iot/iiot devices." IEEe Access 12 (2024): 23733-23750.

[27] Guitouni, Zied, Mohammed Ali Ghaieb, and Mohsen Machhout. "Security analysis of medical image encryption using AES modes for IoMT systems." IJCA Int J Comput Appl 14, no. 2 (2023): 8887.

[28] Rothman, Denis. Transformers for Natural Language Processing: Build, train, and fine-tune deep neural network architectures for NLP with Python, Hugging Face, and OpenAI's GPT-3, ChatGPT, and GPT-4. Packt Publishing Ltd, 2022.

[29] Jiang, Zoe L., Jiajing Gu, Hongxiao Wang, Yulin Wu, Junbin Fang, Siu-Ming Yiu, Wenjian Luo, and Xuan Wang. "Privacy-preserving distributed machine learning made faster." In Proceedings of the 2023 secure and trustworthy deep learning systems workshop, pp. 1-14. 2023.

[30] Rivest, Ronald L., Adi Shamir, and Leonard Adleman. "A method for obtaining digital signatures and public-key cryptosystems." Communications of the ACM 21, no. 2 (1978): 120-126.

[31] Lyu, Weimin, Songzhu Zheng, Lu Pang, Haibin Ling, and Chao Chen. "Attention-enhancing backdoor attacks against bert-based models." arXiv preprint arXiv:2310.14480 (2023).

[32] Schaerf, Ludovica, Eric Postma, and Carina Popovici. "Art authentication with vision transformers." Neural Computing and Applications 36, no. 20 (2024): 11849-11858.

[33] Zheng, YuYu, HaoXuan Huang, and JunMing Chen. "Comparative analysis of various models for image classification on Cifar-100 dataset." In Journal of Physics: Conference Series, vol. 2711, no. 1, p. 012015. IOP Publishing, 2024.

[34] Daemen, Joan, and Vincent Rijmen. The design of Rijndael. Vol. 2. New York: Springer-verlag, 2002.

[35] Qi, Tao, Huili Wang, and Yongfeng Huang. "Towards the robustness of differentially private federated learning." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 38, no. 18, pp. 19911-19919. 2024.

[36] García-Márquez, M., et al.,"Krum Federated Chain (KFC): Using Blockchain to Defend Against Adversarial Attacks in Federated Learning," *arXiv preprint arXiv:2502.06917*, 2025.

[37] Jiang, S., et al., "Towards compute-efficient Byzantine-robust federated learning with fully homomorphic encryption," *Nature Machine Intelligence*, vol. 7, no. 10, pp. 1657–1668, 2025.

[38] Gonçalves, J. P. S.,"Robustness of AI models in software vulnerability detection," 2025.

[39] Barkatsa, Sofia, et al. *Coordinated Jamming and Poisoning Attack Detection and Mitigation in Wireless Federated Learning Networks.,IEEE Open Journal of the Communications Society (2025)*

Post-Quantum Module Learning with Rounding-Based Public Key Encryption Using Incomplete Number Theoretic Transform

Anupama Arjun Pandit[1], Arun Mishra[2]*
Department of Computer Science and Engineering,
Defence Institute of Advanced Technology, Girinagar, Pune, 411025, Maharashtra, India[1,2]
Department of Computer Engineering, Marathwada Mitra Mandal's College of Engineering,
Pune, 411052, Maharashtra, India[1]

***Abstract*—Post Quantum Cryptographic (PQC) techniques are widely used in encryption standards and digital signatures. Lattice-based post-quantum cryptographic techniques have been reported in the last decades. The present work proposes an optimized quantum-safe lattice public key encryption (PKE) scheme based on the Module Learning with Rounding (MLWR) problem, enhanced by the use of Incomplete Number Theoretic Transform (NTT). The objective of the proposed scheme is to achieve efficient encryption and decryption while maintaining robust security in accordance with the National Institute of Standards and Technology (NIST) recommendations. The incomplete NTT relaxes the modulus q requirement, enables a smaller modulus for efficient arithmetic, and reduces computational complexity. This approach results in significant improvements in the speed of key generation, encryption, and decryption with a marked reduction in rejection probability, compared to schemes utilizing complete NTT. The proposed scheme demonstrates competitive performance against other lattice-based encryption schemes such as Kyber and Frodo. It shows lower encryption and decryption times while offering comparable security levels. Proposed scheme is at least a hundred times faster than Frodo lattice-based public-key encryption schemes. For NIST-recommended security level, in proposed scheme, each encryption needs an average of 300K CPU cycles, and each decryption needs 120K CPU cycles. Additionally, modulus 7937 enables a reduction in key and ciphertext sizes, optimizing the scheme for practical deployment in resource-constrained environments. Performance evaluations confirm the practicality of the scheme with substantial reductions in computational overhead, making it a highly efficient and secure candidate for post-quantum encryption.**

***Keywords*—Post-quantum cryptography; public key encryption; lattice-based cryptography; Incomplete Number Theoretic Transform; Module Learning with Rounding; Learning with Errors**

I. Introduction

The rise of quantum computing poses a critical challenge to traditional cryptographic systems, which largely depends on the difficulty of solving mathematical problems like prime factorization and discrete logarithms. These problems form the basis of many widely used encryption schemes, but quantum algorithms, particularly Shor's algorithm [1], can solve them in polynomial time. This has led to an urgent need for cryptographic techniques that can resist quantum attacks, thus fueling the development of post-quantum cryptography (PQC). Since the National Institute of Standards and Technology (NIST) launched an initiative to develop new encryption standards that are unbreakable by quantum computers, PQC [2], [3] is getting more and more attention. One of the most appealing areas in PQC is lattice-based cryptography (LBC). LBC grabbed the theoretical community's interest because LBC structures are accompanied by security proofs ingrained in worst-case instances. Ajtai and Dwork [4] proposed 1^{st} lattice-based encryption algorithm. Regev later simplified and improved on this approach in [5]. The formulation of an intermediary problem, Learning with Errors (LWE) was one of Regev's key contributions. LWE was asymptotically at least as hard as certain traditional worst-case lattice problems [6], [5], and reasonably straightforward to utilize in cryptographic constructs.

LWE involves sampling errors using Gaussian distributions to introduce noise, which forms the backbone of its cryptographic constructs. Banerjee et al. [7] created learning with rounding (LWR) problem, which is an LWE variant with derandomization; LWR creates an instance utilizing deterministic rounding under a small modulus instead of introducing random errors to make it noisy. LWR instance sampling is simpler than LWE instance sampling since it does not use the Gaussian sampling technique. Several recent studies on the hardness of LWR problem have concluded that LWR problem is at least as difficult as the LWE problem when the number of samples is limited [8].

Schemes based on LWE variant require sampling from error distributions for randomness. Also, errors introduced into ciphertexts and public keys inevitably result in increased bandwidth. Since the error is made predictably, LWR-based schemes reduce the bandwidth by using a small rounding modulus. The module variants of the problems enable interpolation of the original LWR/LWE problems and their ring variants, reducing bandwidth and complexity of computation. It is possible to increase the level of security of Ring-LWE/LWR without making any changes to the basic mathematics, as demonstrated by the modules [6], [9].

One key observation in LBC is the dominance of polynomial multiplication as the most time consuming operation in these schemes. LBC relies on high-degree polynomials, making polynomial multiplication the bottleneck for efficiency. Fig. 1 highlights the time profiles of various LBC schemes, confirming that polynomial multiplication significantly impacts performance.

*Corresponding author.

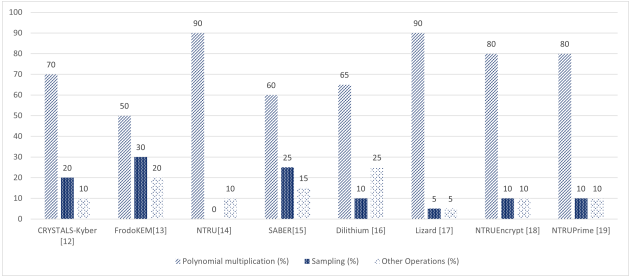


Fig. 1. Time-consuming operations in lattice-based cryptographic schemes [10], [11], [12], [13], [14], [15], [16], [17].

Efficient implementation of polynomial multiplication methods can significantly reduce the computational overhead in LBC schemes, making these schemes more practical for real-world applications.

Lattice-based schemes, particularly those employing the module variants of LWR, strike an effective balance between bandwidth reduction and computational complexity. They facilitate efficient polynomial operations through techniques like Number Theoretic Transform (NTT), a method for fast polynomial multiplication in finite fields. However, the traditional NTT approach can become computationally expensive when polynomials are large. To address this, Incomplete NTT [18] introduces a hybrid methodology where NTT process is truncated early for smaller sub-polynomials allowing alternative, simpler multiplication methods like the schoolbook algorithm to handle smaller parts. This truncation reduces computational overhead and makes the scheme more efficient without compromising security.

Current proposal: Present study proposes an efficient encryption scheme which uses an optimization technique that combines Module Learning With Rounding (MLWR) with Incomplete NTT for polynomial multiplication. The proposed PKE scheme comprises of generation of keys, encryption, and decryption algorithms. The time needed to create keys and complete the decryption procedure is incredibly quick when compared to other lattice-based PKEs. Joel Alwen et.al.'s LWR properties are taken into consideration while choosing parameters for key generation, encryption, and decryption [9]. In the proposed scheme, a public matrix is created using an expansion function that uses the generalized hash function Shake-256 [19]. Additionally, the proposed scheme incorporates the Montgomery Reduction Algorithm [20], which optimizes modular arithmetic by transforming the modulus into a more computationally manageable form. This transformation enables faster computations during key generation, encryption, and decryption processes by reducing the cost of modular multiplications. The Incomplete NTT and Montgomery Reduction Algorithm are employed in the proposed approach to accelerate matrix multiplication and to reduce modulus, respectively. Using the MLWR offers an optimal ground between security and efficiency [21] in comparison to the original LWR and Ring-Learning with Rounding (RLWR) problems. The proposed scheme is more efficient than earlier methods because it also decreases the size of a public key by using a rounding function. For a recommended security level, proposed scheme's public key size is 1632 bytes. Choice of the underlying hard problem, as well as the scheme's parameters (Secret distribution range η, modulus q, polynomial ring $(X^{256}+1)$, and rank of module k), are inspired by 3 key principles: simplicity of the scheme, efficiency, and flexibility.

Technical novelty of proposed work lies in the specific hybrid integration of MLWR problem with an Incomplete NTT using the prime modulus $q = 7937$. While existing lattice-based schemes utilize either standard LWE with NTT (e.g., Kyber) or MLWR with alternative multiplication methods (e.g., SABER), our construction leverages a unique synergy. By setting the truncation level to $\beta = 1$, we halt the butterfly operations early, which reduces computational cycles while maintaining robust security margins. This specific optimization avoids the final, most computationally expensive stage of decomposition, offering a tailored balance between performance and quantum resistance.

Section II gives the necessary background information; Section III presents related work. Section IV describes proposed PKE with correctness and security; Section V discusses implementation details; Section VI discusses the result and its comparison with other lattice-based PKEs; and in Section VII we state the conclusion and future work.

II. Preliminaries

A. Notations

$\mathbb{Z}_q$ is the integer ring under modulo q, and $\mathbb{Z}$ mod q is $\mathbb{Z}$'s reduction in $[0, q)$. With $N = 256$, R_q is quotient ring $\mathbb{Z}_q[x]/(x^N + 1)$. R_q^k is the k-dimensional quotient ring under modulus q. Uppercase bold alphabets are used for matrices, such as "matrix $\boldsymbol{A}$", whereas lowercase regular alphabets are used for scalars, such as "scalar p", and lowercase bold alphabets are used for vectors, such as "vector $\boldsymbol{s}$". The symbol $\mathcal{L}(B)$ denotes the lattice produced by basis set B. Sample vector $\boldsymbol{a}$ from the distribution $\mathcal{D}$ is denoted by $\boldsymbol{a} \leftarrow \mathcal{D}$. The sample vector $\boldsymbol{a}$ according to a centered binomial distribution β_i is denoted by $\boldsymbol{a} \leftarrow \beta_i$. Primitive N^{th} root of unity denoted by ζ. Primitive $2N^{th}$ root of unity denoted by Ψ.

Convolution: For polynomials **a,b** in $\mathbb{Z}[x]$ with length N. Convolution simplifies to

$$c_k = \sum_{i=0}^{k} \mathbf{a}_i \mathbf{b}_{k-i}, \quad k = 0, \ldots, 2N-2$$

This corresponds to the coefficient-wise multiplication of the polynomials.

- Cyclic Convolution: equivalent to polynomials and multiplication in $\mathbb{Z}[x]/(x^N - 1)$
- NWC (Negative Wrapped Convolution): is used for polynomial multiplication when polynomials are in the quotient ring $\mathbb{Z}[x]/(x^N + 1)$

In the context of NTT, the arrangement of input and output coefficients plays a critical role in the efficiency of algorithms. Two common types of orderings are:

- Normal Order (no): Coefficients are arranged in increasing natural order (e.g., $[x_0, x_1, x_2, \ldots, x_{n-1}]$)
- Bit-Reversed Order (bo): Coefficients are rearranged such that their indices correspond to the bit-reversed representation of the original indices

We adopt the notation from [22] for the Cooley-Tukey NTT algorithm. Specifically:

$$NTT^{CT,\Psi}_{no\rightarrow bo,\beta=1}(\mathbf{x})$$

where,

- NTT^{CT}: Indicates the Cooley-Tukey algorithm is used.
- Ψ: Refers to the twiddle factors ($2N-th$ roots of unity) used in NTT the computation.
- $no \rightarrow bo$: Specifies that the input coefficients are in natural order (no), while output coefficients are in bit-reversed order (bo).
- $\beta = 1$: Implies a number of levels cropped in NTT computation.

B. Lattices

A lattice is an infinitely expanded structure of discrete points arranged in a repeating pattern. The lattice is constructed by an integer linear combination of independent basis vectors. Set B is a collection of N-linearly independent vectors. The dimension of the lattice is N. let $B = b_1, b_2, b_3, b_4, b_5, \cdots, b_N$

$$\mathcal{L}(b_1, b_2, b_3, b_4, b_5, \cdots, b_N) = \left\{ \sum_{j=1}^{N} y_j b_j : y_j \in \mathbb{Z} \ \& \ b_j \in \mathbb{R} \right\}$$

C. Module Lattices

Module lattices extend the concept of lattices to modules over a ring. Let R be a ring, and k be a rank of R-module. A module lattice $\mathcal{L}$ in R^k is a discrete set of points in R^k defined as Eq. (1):

$$\mathcal{L} = \left\{ \sum_{i=1}^{k} z_i \mathbf{b}_i : z_i \in R \right\}, \tag{1}$$

where, $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_k \in R^k$ are the basis vectors of the module lattice.

D. Learning with Errors (LWE)

LWE [5] for N-dimensions, α error rate, and q as a modulus denoted as $\text{LWE}_{N,q,\alpha}$ and defined as follows:
From given samples $(\boldsymbol{a}_j, b_j) \in \mathbb{Z}_q^N \times \mathbb{Z}_q$, where: $\boldsymbol{a}_j$ is chosen uniformly at random, and b_j is computed as:

$$b_j = \langle \boldsymbol{a}_j, \boldsymbol{s} \rangle + e_j \mod q$$

with $\langle \boldsymbol{a}_j, \boldsymbol{s} \rangle$ representing the inner product of $\boldsymbol{a}_j$ and a secret vector $\boldsymbol{s} \in \mathbb{Z}_q^N$ and e_j is a small error term sampled from a error distribution χ. The LWE problem is to recover the secret vector $\boldsymbol{s}$ by solving a given noisy linear equation.

The LWE problem can be expressed in matrix form as:

$$\boldsymbol{A}\boldsymbol{s} + \boldsymbol{e} = \boldsymbol{b} \mod q,$$

where, $\boldsymbol{A} \in \mathbb{Z}_q^{N \times N}$ is a matrix with rows $\boldsymbol{a}_j$

E. Learning with Rounding (LWR)

Banerjee et al. [7] first presented the LWR problem for enhancing efficiency of the LWE-based pseudorandom generator (PRG). It was used to build reusable computational extractors, lossy trapdoor functions, and deterministic encryption. In fact, the LWR problem may be considered as a deterministic alternative to LWE problem. The only difference is that the error in LWR is deterministic, which makes it less random than the Gaussian error in LWE. Error in the LWR is caused by a scaling operation in the rounding function. In particular, the error in LWE is β-bounded ($\beta > 2\sqrt{N}$) for security [8], while error in LWR is less than $1/2$. Scaling $\mathbb{Z}_q$ down to $\mathbb{Z}_p$ generates the deterministic error, where p is a rounding modulus.

The scaled rounding function [12] is defined like this:

$$\lfloor \cdot \rceil_p \text{ is } \mathbb{Z}_q \rightarrow \mathbb{Z}_p : p < q \ .$$

For matrix $\boldsymbol{A}$, and vector $\boldsymbol{s}$ a sample of LWR is obtained by:

$$(\boldsymbol{A}, b = \lfloor < \boldsymbol{A}, \boldsymbol{s} > \rceil_p)$$

F. Module Learning with Rounding (MLWR)

MLWR is a module variant of LWR, specifically tailored for cryptographic applications. It combines the modular arithmetic of lattices with the deterministic rounding technique of LWR to achieve efficient and secure constructions.

In the MLWR problem, the objective is to compute the approximate inner product of a secret vector $\mathbf{s}$ with a public matrix $\mathbf{A}$ while reducing noise through deterministic rounding. Formally, for a modulus q and a smaller modulus p, the MLWR sample is generated as:

$$\mathbf{b} = \left\lfloor \frac{p}{q} \cdot (\mathbf{A} \cdot \mathbf{s}) \right\rceil \mod p$$

where,

- $\mathbf{A} \in \mathbb{Z}_q^{k \times k}$ is a public matrix.
- $\mathbf{s} \in \mathbb{Z}_q^k$ is a secret vector.
- $\lfloor \cdot \rceil_p$ denotes deterministic rounding to the nearest multiple of p.

G. Number Theoretic Transform (NTT)

NTT [23] is a discrete Fourier transform (DFT) defined over finite fields. It is a crucial tool for efficient polynomial multiplication in cryptographic schemes, especially in LBC. The NTT maps polynomial coefficients into a point-value representation using N-th roots of unity modulo a prime q.

NTT enables fast polynomial multiplication with integer coefficients and is widely employed in LBC schemes. For the NTT to be well-defined, the modulus q must satisfy $q \equiv 1 \pmod{2N}$, ensuring the existence of a primitive N-th root of unity ζ modulo q.

The goal of NTT is to multiply two polynomials having integer coefficients such that the computed polynomial coefficients lie under a specific module. The advantage of NTT is that it provides a fast technique for multiplication whereas the disadvantage is that NTT is possible with prime modulo of the form like $2^a \cdot b + 1$ where a and b are arbitrary constants.

Given a polynomial $a(x) = \sum_{i=0}^{N-1} a_i x^i \in Z_q[x]/\langle x^N - 1 \rangle$, the NTT transforms it into its evaluation at N-th roots of unity modulo q.

For NTT to be well-defined and invertible, the following conditions must hold:

- The modulus q must be a prime number such that $q \equiv 1 \pmod{2N}$
- A primitive N-th root of unity ζ must exist modulo q, i.e. $\zeta^N \equiv 1 \mod q$

Forward NTT transforms a polynomial $a(x)$ from its coefficient representation into its evaluation (point-value) representation. Forward NTT of $a(x)$ is computed as Eq. (2):

$$\hat{a}_k = \sum_{j=0}^{N-1} a_j \zeta^{jk} \pmod{q}, \quad k = 0, 1, \ldots, N-1 \tag{2}$$

The inverse NTT (iNTT) transforms the point-value representation back to coefficient form [see Eq. (3)]:

$$a_j = \frac{1}{N} \sum_{k=0}^{N-1} \hat{a}_k \zeta^{-jk} \pmod{q}, \quad j = 0, 1, \ldots, N-1 \tag{3}$$

Here, N^{-1} is a modular multiplicative inverse of N modulo q.

Efficient computation of the NTT relies on algorithms that exploit symmetries in the evaluation process. The Fast Fourier Transform (FFT) [24] is an efficient algorithm to compute DFT of a sequence. In context of polynomial multiplication, FFT is instrumental in reducing the computational complexity from $O(N^2)$ to $O(N \log N)$. The FFT decomposes a polynomial into simpler components using a divide-and-conquer strategy, exploiting the properties of roots of unity.

The FFT process involves recursively splitting the polynomial ring $Z_q[x]/(x^N + 1)$ into smaller subrings. At each stage, the polynomial is divided into two halves using specific roots of unity. For example, the ring $Z_q[x]/(x^N + 1)$ is split into $Z_q[x]/(x^{N/2} - \Psi^{N/2})$ and $Z_q[x]/(x^{N/2} + \Psi^{N/2})$, where Ψ is primitive $2N^{th}$ root of unity. This process is repeated until the polynomial is reduced to base components, such as $Z_q[x]/(x - \Psi)$ or $Z_q[x]/(x + \Psi)$, which are directly computable.

Two widely used approaches of FFT are the Cooley–Tukey(CT) [25] and Gentleman–Sande(GS) [26] algorithms:

- Cooley–Tukey(CT) Algorithm: The CT algorithm is also known as the Decimation-In-Time (DIT). It performs the transformation by dividing input sequence into even and odd indices. Key step involved in the butterfly operation is:

$$a' = a + \zeta b, \quad b' = a - \zeta b,$$

where, ζ is the twiddle factor (root of unity).

The butterfly operation for Cooley–Tukey is illustrated in Fig. 2. This method starts with the input in natural order and produces an output in bit-reversed order.

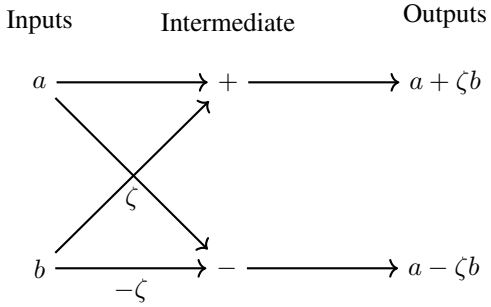


Fig. 2. Cooley-Tukey butterfly.

- Gentleman–Sande (GS) Algorithm: The GS algorithm is also known as the Decimation-In-Frequency (DIF). It follows a similar divide-and-conquer approach but starts with the coefficients of the polynomial arranged in a bit-reversed order. Inverse NTT is often implemented using this algorithm, as it efficiently reconstructs the original sequence from its Fourier-transformed counterpart. The butterfly operation in Gentleman-Sande is given by:

$$a' = a + b, \quad b' = (a - b) \cdot \zeta^{-1}$$

Fig. 3 illustrates the Gentleman–Sande butterfly operation. This algorithm efficiently reconstructs the original sequence from its transformed version.

Both algorithms use butterfly operations to combine intermediate results efficiently.

NTT Applications in Polynomial Multiplication: Polynomial multiplication in cryptographic schemes often involves convolutions. Using NTT, the multiplication of two polynomials $a(x)$ and $b(x)$ modulo $x^N - 1$ is performed as:

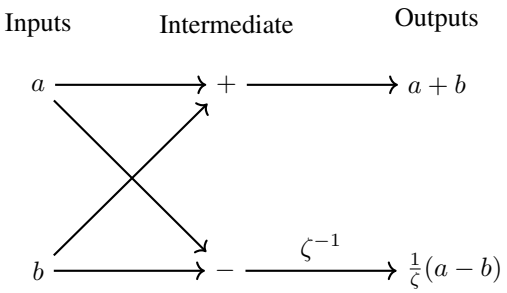


Fig. 3. Gentleman-Sande butterfly.

- Compute the NTT of $a(x)$ and $b(x)$: $\hat{a} = NTT^{CT,\zeta}(a(x))$, $\hat{b} = NTT^{CT,\zeta}(b(x))$ refer Eq. (2).
- Perform pointwise multiplication: $\hat{a}_k \circ \hat{b}_k \pmod q = \hat{c}_k, \ \forall k = 0, 1, \ldots, N-1$, $\circ$ denotes pointwise multiplication in NTT domain.
- Compute inverse NTT of $\hat{c}$ to obtain product polynomial in coefficient form: $c(x) = iNTT^{GS,\zeta^{-1}}(\hat{c})$ refer Eq. (3).

This process reduces the complexity of polynomial multiplication from $O(N^2)$ to $O(N \log N)$, making it highly efficient for cryptographic applications.

NTT has been integrated into the specifications of several post-quantum cryptographic schemes, including Kyber, Saber, and Dilithium. These schemes leverage their efficiency to perform fast polynomial arithmetic, enabling secure and efficient Public Key Encryption (PKE) and signature generation.

H. Incomplete NTT

Incomplete NTT method is employed for polynomial multiplication when dimension (N) of the lattice is a power of two and modulus q is an NTT-friendly prime but does not satisfy $q \equiv 1(mod 2N)$. An incomplete NTT method does not have to map $Z_q[x]/(x^N + 1)$ down to linear terms, and one can stop its mapping before the last β level, $\beta = 0, 1, \cdots, \log N - 1$

Mathematical Formulation: Given two polynomials $\mathbf{x}$ and $\mathbf{y}$ in $Z_q[x]/(x^N + 1)$, their product $\mathbf{r}$ using incomplete NTT is computed as:

$$\mathbf{r} = \text{INTT}_{bo\to no}^{GS,\Psi^{-1}} \left(\text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{x}) \circ \text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{y}) \right),$$

where:

- $\text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}$: Forward Cooley-Tukey NTT [25] up to $\beta = 1$ level
- $\text{INTT}_{bo\to no}^{GS,\Psi^{-1}}$: Inverse Gentleman-Sande NTT [26] to reconstruct the coefficients

Fig. 4 illustrates the polynomial splitting process in the incomplete NTT, showing how the polynomial ring $Z_q[x]/(x^N + 1)$ is gradually decomposed into smaller components till it reduces to degree two polynomials. The decomposition stops early, reducing computational overhead while maintaining accuracy. It requires $(\log(N) - 1)$ steps.

Advantages of Incomplete NTT:

- Relaxed Conditions for Modulus q: Incomplete NTT requires only a $\frac{N}{2}$-th root of unity for cyclic NTTs or an N-th root of unity for negacyclic NTTs, significantly increasing the range of valid q.
- Computational Efficiency: As it stops decomposition at an earlier level, incomplete NTT reduces the number of butterfly operations, saving additions and subtractions at the cost of slightly more complex base multiplications.
- Adaptability: The degree of incompleteness β can be tailored to optimize performance for specific platforms or resource constraints.

The forward incomplete NTT algorithm transforms a polynomial into its NTT domain representation using CT butterfly operations. Unlike complete NTT, which fully decomposes the polynomial down to linear terms, incomplete NTT stops earlier, saving computations and providing more freedom in parameter selection. The algorithm operates as follows:

- Initialization: Start with the full polynomial length N and segment it into smaller parts.
- Butterfly Operations: Apply Cooley-Tukey butterfly operations to each segment. For each pair of coefficients, compute their sum and difference, scaled by a twiddle factor, which is a precomputed power of a $2N$-th root of unity Ψ.
- Modulo Reduction: Ensure all intermediate results remain within the finite field $\mathbb{Z}_q$ by applying modulo q.

The inverse incomplete NTT reconstructs the coefficients from their partially transformed representation. The steps include:

- Initialization: Begin with segments of size 1 and progressively combine them to reconstruct the full polynomial.
- Butterfly Operations: Combine pairs of coefficients, using inverse twiddle factors and ensuring results remain within $\mathbb{Z}_q$.
- Normalization: After reconstruction, normalize all coefficients by multiplying with N_inv, the modular multiplicative inverse of N.

The incomplete NTT provides a flexible and computationally efficient alternative to complete NTT for polynomial arithmetic in cryptography. By halting decomposition at a higher level β, it reduces computational costs while relaxing constraints on the modulus q. This hybrid approach is particularly beneficial for resource-constrained environments like embedded systems and is implemented in schemes such as Kyber to achieve optimal performance.

III. Related Work

Lattice-based cryptography has emerged as a promising candidate for efficient and secure PQC schemes. The security of LBC is based on worst-case hard problems such as the Shortest Vector Problem (SVP) [27], Closest Vector Problem (CVP) [28], approximate Shortest Independent Vectors Problem (SIVP) [6], LWE [5] and Small Integer Solution (SIS) [6]. These problems provide robust security guarantees due to their inherent computational difficulty.

Early lattice-based PKE schemes, such as the Ajtai-Dwork cryptosystem [29], relied on SVP for security. However these schemes required high-dimensional lattices, leading to impractically large key sizes. The introduction of the NTRU cryptosystem [12] marked a significant milestone, offering compact key sizes and security rooted in polynomial rings. Later, Regev's LWE-based scheme [5] provided further improvements in efficiency, though it exhibited limited resistance to active attacks like chosen plaintext attack(CPA). Subsequent works, including Lindner and Peikert's PKE [30], reduced key and ciphertext sizes by leveraging optimized parameters.

In 2016, NIST initiated a call for PQC algorithm standardization [2], propelling the development of LBC. Notable candidates such as Kyber, Saber, and Frodo emerged, each demonstrating unique strengths:

- Kyber [10], a finalist in NIST PQC project, is based on Module-LWE(MLWE) problem. It incorporates NTT for efficient polynomial multiplication, enabling fast and memory-efficient implementations. Kyber's use of centered binomial distributions for sampling and its compatibility with NTT-friendly parameters make it highly optimized for modern cryptographic applications.

- Saber [13] relies on the MLWR problem, which eliminates error sampling and reduces randomness requirements compared to MLWE-based schemes. Saber uses alternative polynomial multiplication methods such as Karatsuba and Toom-Cook due to its power-of-two moduli, which are incompatible with traditional NTTs.

- Frodo [11] employs standard LWE and matrix-based multiplication over $\mathbb{Z}_q$. Although Frodo offers strong theoretical security guarantees, it suffers from higher bandwidth and computational requirements compared to structured lattice-based schemes.

MLWR problem offers multiple benefits that make it a compelling choice for cryptographic schemes:

- Simplified Sampling: Unlike MLWE, MLWR eliminates the need for error sampling, reducing computational overhead and simplifying implementations.

- Optimized Bandwidth: By using a small rounding modulus, MLWR achieves smaller ciphertext and key sizes, making it suitable for bandwidth-constrained environments.

- Security Guarantees: MLWR maintains security equivalence to MLWE under specific parameter constraints, offering robust protection against quantum attacks.

- Implementation Flexibility: MLWR-based schemes can be efficiently implemented using alternative polynomial multiplication techniques like NTT and Karatsuba, providing flexibility in resource-constrained environments.

The aforementioned benefits make MLWR a highly attractive foundation for designing practical and secure LBC systems, particularly in modern applications requiring both efficiency and scalability.

Proposed approach bridges the gap between the efficiency of power of two rounding in SABER and the fast NTT arithmetic found in Kyber. The novelty over SABER is a use of Incomplete NTT for faster polynomial multiplication, whereas the novelty over Kyber is the selection of $q = 7937$. Mathematically, $q = 7937$ is highly advantageous; it satisfies $q \equiv 1 \pmod{256}$, ensuring NTT compatibility, yet yields a rejection probability of only 0.031. This is a significant improvement over standard NTT-friendly primes

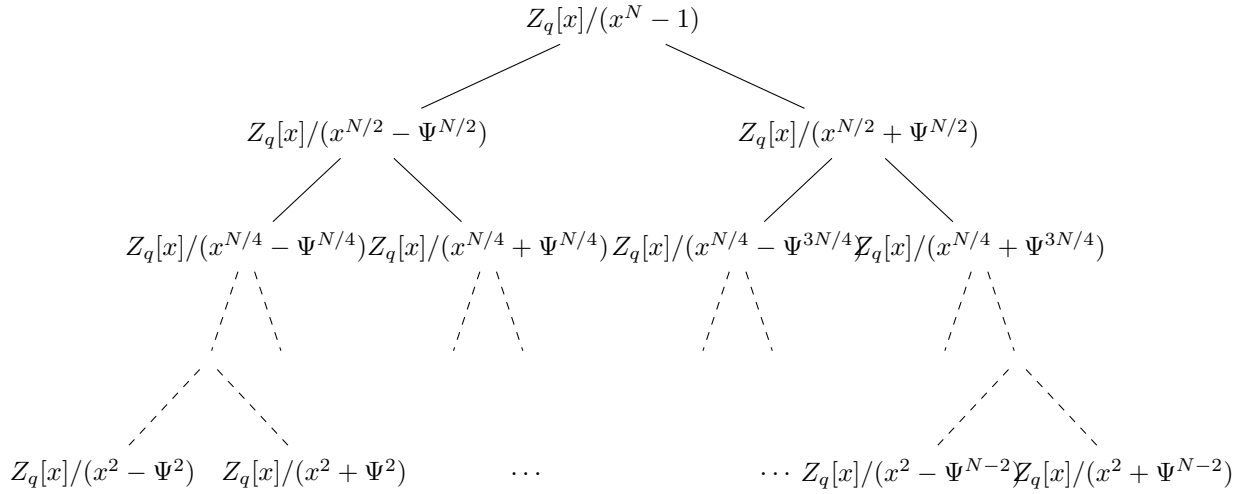


Fig. 4. Polynomial splitting in incomplete NTT.

like $q = 3329$ (rejection probability 0.594), directly optimizing the key expansion and sampling processes. Consequently, the proposed scheme achieves competitive execution times while minimizing the randomness requirements inherent in MLWE-based constructions.

A. NTT in Lattice-Based Cryptography

Efficient polynomial multiplication is a critical component of LBC. NTT and its inverse (INTT) play a pivotal role in modern implementations. We analysed NTT-based implementations in prominent NIST PQC schemes such as Kyber [10], Saber [13], NTRU [12], Dilithium [14], and Falcon [31]. These schemes adopt diverse strategies to optimize polynomial multiplication based on their underlying ring structures and security requirements.

NTT is particularly efficient for schemes with moduli satisfying $q \equiv 1 \pmod{2N}$, as seen in Kyber, Falcon, and Dilithium. Conversely, schemes like Saber and NTRU employ alternative techniques due to their use of power-of-two moduli or non-NTT-friendly rings. For example, Kyber transitioned from full NTT in its initial rounds to an incomplete NTT approach in later rounds, leveraging optimized Fast Fourier Transform (FFT) techniques to reduce computational complexity. NTT-unfriendly rings refer to polynomial rings where the structure of the ring does not align well with the standard requirements of NTT, such as specific polynomial degrees or forms. Common examples include rings where the degree N of the polynomial is not a power of two.

Incomplete NTT is a specialized variant of the traditional NTT that halts the decomposition process at an intermediate stage, skipping the final levels of polynomial transformation. This method reduces the number of butterfly operations required in Cooley-Tukey or Gentleman-Sande algorithms, significantly lowering computational complexity. Additionally, the incomplete NTT relaxes modulus requirements, enabling the use of smaller moduli, 7937. The reduced modulus allows for more efficient arithmetic operations, optimized memory usage, and smaller key and ciphertext sizes. These advantages make the incomplete NTT particularly best-suited for resource-constrained environments, such as IoT devices [32], where computational efficiency and reduced bandwidth are paramount.

B. Proposed Approach: Combining MLWR and Incomplete NTT

The proposed PKE scheme integrates the advantages of MLWR and the computational efficiency of incomplete NTT. MLWR reduces bandwidth and randomness requirements by eliminating error sampling, as demonstrated in Saber. Meanwhile, incomplete NTT minimizes computational overhead while maintaining compatibility with lattice-based polynomial arithmetic. These optimizations align with insights from Falcon [31] and Kyber [10], where NTT-based methods significantly enhance performance. By adopting this hybrid approach, the proposed scheme offers a post-quantum cryptographic solution with balanced security and efficiency.

IV. Proposed MLWR-Based PKE using Incomplete NTT

From the analysis of existing NTT in LBC, we have chosen the most efficient Incomplete FFT-based NTT for implementation of MLWR-based PKE. The first advantage of the proposed scheme is that it can reduce the ciphertext size by mapping coefficients from a larger ring $\mathbb{Z}_q$ to a smaller ring $\mathbb{Z}_p$. The second advantage is encryption and decryption times and the required CPU cycles are comparable to other prominent lattice-based candidates of the NIST PQC project.

Proposed PKE scheme incorporates the Incomplete NTT for efficient polynomial operations in LBC. The process begins with generating cryptographic parameters which are used to create a public key (for encryption) and a secret key (for decryption). A plaintext message is encrypted with a public key, producing a ciphertext. The decryption process uses a secret key $\boldsymbol{s}$ to recover original message. Incomplete NTT is applied during both encryption and decryption to optimize computational performance. This design ensures secure communication while enhancing efficiency in lattice-based cryptosystems.

Proposed scheme is made up of the following three algorithms:

A. MLWR Key Generation Algorithm

The proposed key generation algorithm constructs a public-private key pair for the MLWR-based PKE scheme. The process is detailed as follows:

- Seed Generation: A cryptographic seed denoted as $seed_\delta$, is generated using the SHAKE-256 hash function. This seed is a part of the public key pk and is used to generate a $k \times k$ public matrix $\boldsymbol{A}$.
- Matrix Construction: The matrix $\boldsymbol{A}$ is constructed with elements as polynomials in a ring $R_q = \mathbb{Z}_q[x]/(x^N + 1)$.

  Here:

  - q is a prime modulus suitable for incomplete-NTT operations.
  - $N = 256$, the degree of the polynomial, satisfies the incomplete-NTT and LWR specifications.
- Secret Key Generation: The secret key $\boldsymbol{s}$ is an N-dimensional short vector sampled from the ring R_q. Coefficients of $\boldsymbol{s}$ are chosen from a small range to ensure efficiency and maintain security.
- Public Key Computation: Public key attribute $\boldsymbol{b}^T$ is computed as:

$$\boldsymbol{b}^T = \left\lfloor \frac{p}{q} \cdot (\boldsymbol{A} \cdot \boldsymbol{s}) \right\rceil_p$$

  where: The multiplication $\boldsymbol{A} \cdot \boldsymbol{s}$ is optimized using the *Incomplete NTT* method, and p is the rounding modulus, smaller than q.

- Output: Public key pk consists of seed $seed_\delta$ and k-dimensional MLWR sample $\boldsymbol{b}^T$. The private key sk is the short vector $\boldsymbol{s}$.

The pseudocode for key generation algorithm is presented in Algorithm 1.

Algorithm 1 KeyGeneration($seed_\delta$)

1: $seed_\delta \leftarrow D\{0,1\}^N$ ▷ Generate a random binary seed.
2: $\boldsymbol{s} \leftarrow Ds_\eta^k$ ▷ Sample secret key $\boldsymbol{s}$ from distribution Ds_η^k
3: $\boldsymbol{A} \in R_q^{k\times k} := Expansion(seed_\delta)$ ▷ Expand the matrix $\boldsymbol{A}$ from the seed.
4: $\boldsymbol{As} = \text{INTT}_{bo\to no}^{GS,\Psi^{-1}}\left(\text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{A}) \circ \text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{s})\right)$ ▷ Efficiently compute $\boldsymbol{As}$ using incomplete NTT.
5: $\boldsymbol{b^T} = \lfloor \frac{p}{q}\boldsymbol{As}\rceil_p \in R_p^{k\times 1}$ ▷ Apply rounding operation
6: return $(pk := (seed_\delta, \boldsymbol{b^T}), sk := (\boldsymbol{s}))$ ▷ Output public and private key pair.

B. MLWR Encryption Algorithm

The encryption algorithm in the proposed MLWR-based PKE scheme efficiently transforms the plaintext into a ciphertext while ensuring performance. The steps are as follows:

- Plaintext Representation: The plaintext is represented as a $mbit$.
- Matrix Construction: Seed $seed_\delta$ is used to construct the matrix $\boldsymbol{A}$, whose elements belong to the ring $R_q = \mathbb{Z}_q[x]/(x^N+1)$
- Random Vector Sampling: A random N-dimensional vector $\boldsymbol{a}$ is sampled. This vector is then multiplied with the matrix $\boldsymbol{A}$ using the *Incomplete NTT* method, to optimize the computational efficiency. The result is stored in $\boldsymbol{u}$:

$$\boldsymbol{u} = \boldsymbol{A}\cdot\boldsymbol{a}$$

- Ciphertext Calculation: Using the public key attribute $\boldsymbol{b^T}$ and the plaintext message bit $mbit$ ($messagespace \in 0,1$), the ciphertext $\boldsymbol{c}$ is computed as:

$$\boldsymbol{c} = \boldsymbol{b^T a} + mbit\cdot\left\lfloor\frac{q}{2}\right\rfloor$$

  $mbit\cdot\lfloor\frac{q}{2}\rfloor$ operation encodes original message.
- Output: The encryption function outputs ciphertext $\boldsymbol{c}$ and the preamble $\boldsymbol{u}$.

Pseudocode for the encryption is presented in Algorithm 2.

C. MLWR Decryption Algorithm

The decryption process in the proposed MLWR-based PKE scheme is the final step, responsible for recovering the original plaintext message. The procedure involves the following steps:

- Intermediate Value Calculation: Using the secret key $\boldsymbol{s}$, the algorithm computes the intermediate value d_1 as:

Algorithm 2 MLWREncryption(pk,mbit)

1: $\boldsymbol{A} \in R_q^{k\times k} := Expansion(seed_\delta)$ ▷ Expand the matrix $\boldsymbol{A}$ from the seed.
2: $\boldsymbol{a} \leftarrow Ds_\eta^k$ ▷ Sample vector $\boldsymbol{a}$ from distribution Ds_η^k
3: $u = \text{INTT}_{bo\to no}^{GS,\Psi^{-1}}\left(\text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{A}) \circ \text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{a})\right)$ ▷ Compute $\boldsymbol{Aa}$ using incomplete NTT.
4: $c = [\boldsymbol{b^T a} + mbit\cdot\lfloor\frac{q}{2}\rfloor]$ ▷ Add scaled message to the rounded public key component.
5: return (u, c) ▷ Output the ciphertext components u and c.

$$d_1 = \boldsymbol{u}\cdot\boldsymbol{s}$$

Here, $\boldsymbol{u}$ is the preamble received with the ciphertext.

- Decrypted Value: The decrypted value d is obtained by subtracting the computed d_1 from the ciphertext $\boldsymbol{c}$:

$$d = \boldsymbol{c} - d_1$$

All operations are performed modulo q.
- Plaintext Recovery: The range of $d \mod q$ is checked to determine the original plaintext bit ($mbit$):
  - If $\frac{-q}{4} \le d \le \frac{q}{4}$, the plaintext $mbit$ is set to $\mathbf{0}$
  - If $\frac{q}{4} < d \le \frac{3q}{4}$, the plaintext $mbit$ is set to $\mathbf{1}$

Algorithm 3 presents the pseudocode for decryption.

Algorithm 3 MLWRDecryption($\boldsymbol{s}, c, u$)

1: $d_1 = \text{INTT}_{bo\to no}^{GS,\Psi^{-1}}\left(\text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{u}) \circ \text{NTT}_{no\to bo,\beta=1}^{CT,\Psi}(\mathbf{s})\right)$ ▷ Compute $\boldsymbol{us}$ using incomplete NTT.
2: $d = c - d_1$ ▷ Subtract the private component from the ciphertext.
3: **if** $(((d \ mod \ q) \ge \frac{-q}{4})$ *AND* $((d \ mod \ q) \le \frac{q}{4}))$
4: **then** ▷ Check if d lies in the range
5: $d = 0$
6: **else**
7: **if** $(((d \ mod \ q) > \frac{q}{4})$ *AND* $((d \ mod \ q) \le \frac{3q}{4}))$
8: **then**
9: $d = 1$
10: return d ▷ Output the decrypted message.

D. Correctness of the Scheme

The correctness of the decryption algorithm relies on ensuring that the introduced rounding error remains within acceptable bounds.

If (u, c) is a correctly formulated ciphertext, the decrypted value is given as Eq. (4):

$$\begin{aligned} d &= c - d_1 \qquad \text{from Algorithm 3} \\ &= c - \boldsymbol{s}^T u \\ &= \boldsymbol{b^T a} + mbit\cdot\frac{q}{2} - \boldsymbol{s}^T\boldsymbol{Aa} \qquad \text{from Algorithm 2} \end{aligned} \tag{4}$$

Here, $\boldsymbol{b^T}$ is defined during key generation process as:

$$\boldsymbol{b^T} = \left\lfloor \frac{p}{q} \boldsymbol{s}^T \boldsymbol{A} \right\rceil \mod p$$

This rounding operation can be rewritten as an LWE problem:

$$\boldsymbol{b^T} \approx \boldsymbol{s}^T \boldsymbol{A} + \frac{p}{q} e_r \pmod q,$$

where, $\frac{p}{q} e_r$ introduces a random error into $\boldsymbol{s}^T \boldsymbol{A}$, and $p < q$. Let $\frac{p}{q} e_r = e_{rnd}$, then:

$$\boldsymbol{b^T} \approx \boldsymbol{s}^T \boldsymbol{A} + e_{rnd}$$

By substituting $\boldsymbol{b^T} = \boldsymbol{s}^T \boldsymbol{A} + e_{rnd}$ into Eq. (4):

$$\begin{aligned} d &= (\boldsymbol{s}^T \boldsymbol{A} + e_{rnd})\boldsymbol{a} + mbit \cdot \lfloor \frac{q}{2} \rfloor - \boldsymbol{s}^T \boldsymbol{A}\boldsymbol{a} \\ &= \boldsymbol{s}^T \boldsymbol{A}\boldsymbol{a} + e_{rnd}\boldsymbol{a} + mbit \cdot \lfloor \frac{q}{2} \rfloor - \boldsymbol{s}^T \boldsymbol{A}\boldsymbol{a} \\ &= e_{rnd}\boldsymbol{a} + mbit \cdot \lfloor \frac{q}{2} \rfloor \end{aligned} \tag{5}$$

To analyze correctness, consider two cases based on the value of $mbit$:

a) Case 1: When $mbit = 0$, Eq. (5) simplifies to:

$$d = e_{rnd}\boldsymbol{a}$$

For correctness, the term $||e_{rnd}\boldsymbol{a}||$ must satisfy:

$$||e_{rnd}\boldsymbol{a}|| \in \left[\frac{-q}{4}, \frac{q}{4}\right]$$

Since $||e_{rnd}|| \in \left[\frac{-q}{4}, \frac{q}{4}\right]$ with high probability, correctness is achieved as $||e_{rnd}\boldsymbol{a}||$ lies within the valid range.

b) Case 2: When $mbit = 1$, Eq. (5) becomes:

$$d = e_{rnd}\boldsymbol{a} + \lfloor \frac{q}{2} \rfloor$$

Correctness requires that:

$$||e_{rnd}\boldsymbol{a} + \lfloor \frac{q}{2} \rfloor|| \in \left[\frac{q}{4}, \frac{3q}{4}\right]$$

For this condition to hold, the rounding error term $e_{rnd}\boldsymbol{a}$ must be small enough to offset the added $\lfloor \frac{q}{2} \rfloor$. Hence, the threshold for $||e_{rnd}||$ becomes:

$$||e_{rnd}|| \in \beta = \left[\frac{-q}{4}, \frac{q}{4}\right]$$

By setting $mbit = 0$ or $mbit = 1$, the correctness of the decryption is ensured when the rounding error e_{rnd} remains within the defined threshold. The analysis confirms that $||e_{rnd}\boldsymbol{a}||$ lies within the valid range $\left[\frac{q}{4}, \frac{3q}{4}\right]$ with high probability, ensuring accurate decryption.

Conclusion: As $||e_{rnd}\boldsymbol{a}||$ lies within the valid interval $\left[\frac{-q}{4}, \frac{q}{4}\right]$, the scheme ensures correctness. This analysis highlights the role of rounding errors and the chosen $mbit$ in maintaining the accuracy of decryption.

E. Parameter Selection

Parameter selection is an important part of the proposed scheme. Table I shows selected parameters for proposed PKE:

TABLE I. PARAMETERS FOR THE PROPOSED SCHEME

Parameters	**Value**
Modulus	$q = 7937$
Secret coefficient range η	[-2,2] for high level security, [-3,3] for recommended security
Degree of polynomial ring (N)	256
Polynomial ring	$X^{256} + 1$
Rank of module k	2 (for security level 2), 3 (for security level 3), 4 (for security level 5)

Selection of the modulus q for PKE involves several critical considerations to ensure security, efficiency, and compatibility with lattice-based cryptographic operations. Here's a detailed process to select appropriate parameters. Mathematical Properties:

- Prime Modulus: Typically, q is chosen to be a prime number.
- NTT Compatibility: For efficient NTT-based polynomial multiplication, modulus q should satisfy $q \equiv 1 \mod 256$, where 256 is a degree of the polynomial (typically a power of 2). This ensures existence of primitive 256-th roots of unity modulo q.

Coefficients of the polynomial $\boldsymbol{s}$ and the public matrix $\boldsymbol{A}$ are sampled uniformly from $\mathbb{Z}_q$ using the rejection sampling method [33]. This approach generates random numbers of the required bit length from the SHAKE-128 pseudorandom number generator (PRNG) as candidate values. Only numbers that are strictly less than q are considered valid and accepted. The rejection probability, which quantifies the likelihood of a candidate being discarded, is given by:

$$1 - \frac{q}{2^{\log q}}$$

We have implemented Python code to find out different values of modulus q that satisfy the required mathematical properties. These values are listed in Table II.

Modulus 7937 stands out as an excellent choice for q due to its low rejection probability. The secret coefficient range η determines the distribution from which the coefficients of the secret key are drawn.
In LBC, the security of the system is closely tied to the noise distribution, as it controls the hardness of lattice problems MLWR.

TABLE II. PRIME MODULUS FOR INCOMPLETE NTT

Modulus q	q inverse	256-th root of unity	≈ Rejection probability
3329	-3327	17	0.594
7681	-7679	198	0.062
7937	-7935	71	0.031
9473	-9471	88	0.422
10753	-10751	29	0.344
11777	-11775	50	0.283
12289	-12287	09	0.250

Rejection Probability: It is calculated as $(1 - \frac{q}{2^{\log q}})$. Lower rejection probabilities are preferred for efficiency.

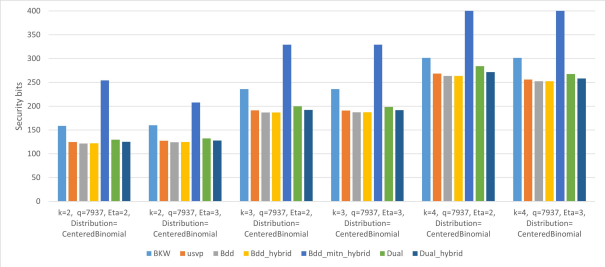


Fig. 5. Choice of secret distribution range.

To estimate the security of different parameter sets of the proposed scheme, lattice estimator [34] is used. The estimator takes parameters as input and gives an estimated security against lattice attacks. Fig. 5 compares the security estimates of various parameter sets under different attack models. These estimates reflect the difficulty of breaking the cryptographic scheme with the given parameters and secret coefficient distribution.

It has been observed that as k increases, the security levels increase across all attack models.

Increasing η slightly decreases security under certain attack models because larger secret coefficients may expose more information.

For lower computational cost and acceptable security, use of $(k = 2, \eta = 3)$. $(k = 3, \eta = 2)$ is a good intermediate choice, balancing security and performance. For maximum security, use $(k = 4, \eta = 2)$ with centered binomial distribution. This offers the highest resistance.

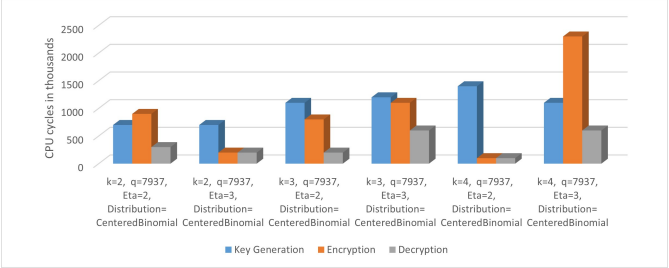


Fig. 6. CPU cycles w.r.t. different distributions.

The given Fig. 6 outlines the computational cost (in CPU cycles) for operations (Key Generation, Encryption, Decryption, and NTT) across different parameter sets.

It shows that higher lattice dimensions $(N \times k)$ lead to increased computational costs for all operations. $N \times k = 512, \eta = 3$ offers the lowest computational cost, particularly for Encryption and Decryption, making it ideal for resource-constrained devices. $N \times k = 768, \eta = 2$ provides a good balance between security and computational cost. $N \times k = 1024, \eta = 2$ is the best choice for applications requiring maximum security, albeit at the cost of higher computation, particularly for NTT.

The selected parameters, which include modulus q=7937, secret coefficient range $\eta \in [2, 3]$, polynomial ring $[x]/(x^{256} + 1)$, module rank k, and incomplete NTT, provide a balanced combination of security, efficiency, and computational performance. The careful selection of these parameters ensures that the proposed MLWR-based PKE scheme meets the requirements for post-quantum cryptographic applications while optimizing execution time and maintaining a high level of security.

- Parameter Selection and Failure Probability: The parameters in Table I are selected to minimize the Decryption Failure Probability (DFP) while maximizing security.
  - Modulus $q = 7937$: This prime satisfies $q \equiv 1 \pmod{256}$, supporting NTT, and provides a low rejection probability of 0.031 [33].
  - Rounding and Noise: We use a centered binomial distribution β_η for secrets. Rounding modulus is chosen such that rounding error e_{rnd} satisfies $||e_{rnd}|| \in [-\frac{q}{4}, \frac{q}{4}]$.
  - DFP Analysis: Decryption is successful if the accumulated noise from rounding does not cross the decision threshold. For the recommended set $(k = 3, \eta = 2)$, the probability of a decryption failure is estimated to be $< 2^{-128}$ [34].

F. Security Estimation and Comparison

This section presents the security estimation of proposed MLWR-based PKE over different moduli. Blum–Kalai–Wasserman (BKW) [34] attack is used to estimate the computing costs of solving corresponding hard problems for security evaluation.

The BKW algorithm, originally developed for solving Learning Parity with Noise (LPN) problem, was extended to the LWE problem. It uses a blockwise approach to reduce the dimension of the problem while managing the induced noise. The process involves sample reduction using sort-and-match techniques, hypothesis testing to distinguish between valid and invalid guesses of sub-vectors, and back substitution to iteratively reduce the problem's dimensionality. The advancements in BKW, such as lazy modulus switching and quantization, have significantly improved its efficiency, making it a central tool in security evaluations of LBC schemes.

Martin Albrecht's lattice estimator tool [34] is used for this task. Table III summarizes the security estimations for proposed PKE scheme against the BKW attack. It includes different parameter sets with specific lattice dimensions 512, different moduli, and an estimated ≈ Cost.

TABLE III. ESTIMATED SECURITY OF DIFFERENT MODULI FOR INCOMPLETE NTT.

Modulus	Required CPU timing	Estimated ≈ Cost
3329	KeyGen: 69, Enc: 172, Dec: 69, NTT: 45483	$2^{190.3}$
7681	KeyGen: 103, Enc: 69, Dec: 69, NTT: 47793	$2^{182.9}$
7937	KeyGen: 103, Enc: 69, Dec: 34, NTT: 4675	$2^{183.4}$
9473	KeyGen: 138, Enc: 103, Dec: 34, NTT: 46103	$2^{176.2}$
10753	KeyGen: 138, Enc: 103, Dec: 34, NTT: 47690	$2^{176.2}$
11777	KeyGen: 310, Enc: 69, Dec: 34, NTT: 54552	$2^{177.5}$
12289	KeyGen: 241, Enc: 69, Dec: 34, NTT: 49138	$2^{178.3}$

Table III provides a comparison of estimated security of different moduli and their performance metrics for an incomplete NTT in an MLWR-based PKE scheme. The key parameters compared are:

Modulus: The modulus used in the polynomial ring.

Required CPU Timing: This includes the CPU timing for key generation (KeyGen), encryption (Enc), decryption (Dec), and NTT operations.

Estimated Approximate Cost: The security level against the BKW attack is represented as a cost in terms of computational effort (expressed as powers of 2).

The modulus 7937 offers the best decryption time (34), which is crucial for practical applications. It also maintains a competitive NTT timing. 7937 still provides a robust security level at $2^{183.4}$, which is sufficient for most practical purposes. Therefore, the modulus 7937 strikes the best balance between performance and security, making it an optimal choice for the MLWR-based PKE scheme.

1) Formal security analysis: Hardness of the Module Learning with Rounding (MLWR) problem serves as the foundation for the security of the proposed PKE. Security in Indistinguishability under Chosen Plaintext Attack(IND-CPA) is avaluated.

Theorem 1. If the $MLWR_{k,N,q,p}$ problem is hard, then the proposed PKE is IND-CPA secure.

Proof:

Proof proceeds via a sequence of games:

- Game 0: This is the original IND-CPA security game. The public key is $(seed_\delta, \mathbf{b}^T)$ where $\mathbf{b}^T = \lfloor \frac{p}{q}\mathbf{s}^T\mathbf{A} \rceil$ [21].
- Game 1: The MLWR sample $\mathbf{b}^T$ is replaced with a vector $\mathbf{u}$ sampled uniformly at random from R_p^k. Under the MLWR assumption, no polynomial-time adversary can distinguish Game 1 from Game 0 with non-negligible advantage [21], [35].
- Game 2: The ciphertext components (u, c) are replaced with uniform samples from R_p^k and R_p. Since the public key is now uniform, the message $mbit$ is perfectly masked.

The advantage of an adversary $\mathcal{A}$ is bounded by $Adv_{PKE}^{ind-cpa}(\mathcal{A}) \leq Adv_{MLWR}^{dist}(\mathcal{B})$, where $\mathcal{B}$ is a simulator solving the MLWR decision problem.

Reduction confirms that the only way for an adversary to break IND-CPA security of the proposed PKE is by solving a underlying MLWR decision problem. With parameter set $(k = 3, \eta = 2)$, the computational cost to perform such an attack exceeds $2^{183.4}$ core-SVP operations, thereby ensuring robust quantum-resistant security.

V. IMPLEMENTATION DETAILS OF MLWR-BASED PKE USING INCOMPLETE NTT

The first step in the proposed approach is to generate a matrix $\boldsymbol{A} \in R_q^{k\times k}$ using a random seed $seed_\delta\{0,1\}^{256}$. The hash function SHAKE-128 [19] is used in the current study to expand the seed to matrix $\boldsymbol{A}$, producing enough unpredictability via a high probability. In the encryption process, Matrix $\boldsymbol{A}$ is helpful for both key creation and multiplication operations. Incomplete NTT was utilized to speed up multiplication. The condition of the scheme is that $\boldsymbol{A}$ must be evenly sampled. The expansion function outputs a uniform matrix $\boldsymbol{A}$ in NTT format with coefficients ranging from 0 to $q-1$.

The result of the expansion function is the matrix $\boldsymbol{A} \in R_q^{k\times k}$ from a 256-byte seed input. It converts a uniform 256-byte seed to a uniform matrix $\boldsymbol{A}$ value using the NTT representation. Matrix $\boldsymbol{A} \in R_q^{k\times k}$ is uniformly random with $k = 4$ and elements are polynomials under a ring $\mathbb{Z}_q[x]/(x^N + 1)$. N =256 is the polynomial ring's degree and k is the rank of a module that is used to find out the dimension of the fundamental lattice problem $N \cdot k$.

Rejection sampling on the integer stream generated by the SHAKE-128 is part of the matrix $\boldsymbol{A}$ generation process. Once the matrix $\boldsymbol{A}$ is generated, the next step involves sampling a short vector $\boldsymbol{s}$ through rejection sampling [33]. This sampled vector $\boldsymbol{s}$ will be used as a secret key.

The public key pk contains 2 parts, 1^{st} part is seed $seed_\delta$ and the second part is $\boldsymbol{b^T} = \lfloor \frac{p}{q}\boldsymbol{s}^T\boldsymbol{A} \rceil_p$. Matrix multiplication is the most important process in the scheme. Present work uses the NTT representation of $\boldsymbol{A}$ and $\boldsymbol{s}$ to carry out pointwise multiplication. Then perform inverse NTT transform on the above multiplication result and also multiply with Montgomery factor for modular reduction. Modular reduction uses the Montgomery algorithm. Rounding $\boldsymbol{b^T}$ operation reduces the size of a public key pk. Minimize size of pk from R_q to R_p in LWR, where modulus $q > p$. The modular reductions necessary for ring R_q arithmetic in the proposed method never use the % (mod) operator. Instead, use special modulus q-specific reduction techniques and Montgomery reductions without the corrective phases. The rounding functions have been calculated using branching-free methods. To make sure that the findings are accurately represented in the reduced form, the Montgomery factor $2^{32} \ mod \ q$ is included in the previously computed roots. Montgomery reductions are used in the NTT domain representations after the pointwise product of the polynomials.

The implementation of the incomplete NTT involves various trade-offs between code size and speed, particularly significant for different hardware environments. On embedded processors, the focus is on minimizing storage and memory requirements, such as avoiding additional temporary storage during polynomial multiplications in R_q. To achieve this, precomputed tables of powers of a primitive root (ψ) are used

to streamline operations, while forward NTT (from normal to bit-reversed order) and inverse NTT (from bit-reversed to normal order) are implemented efficiently.

A. Implementation Security and Side-Channel Mitigation

While primary focus of this work is performance optimization, implementation incorporates several design principles to mitigate side-channel vulnerabilities :

- Constant-Time NTT: The Incomplete NTT is implemented with a fixed execution path [36]. There are no data-dependent branches or secret-dependent loops, which prevents leakage of timing information during polynomial multiplication.
- Montgomery Reduction: We utilize a constant-time Montgomery reduction [20]. By incorporating the Montgomery factor $2^{32} \pmod q$ into the precomputed NTT twiddle factors [26], we avoid the conditional subtraction step typically found in modular reduction, thereby mitigating simple power analysis and timing attacks.

VI. Result and Analysis

A. Experimental Setup

Proposed scheme and comparative benchmarks were implemented in C language, compiled with the GCC compiler. To ensure a fair and standardized evaluation of performance, all experiments were conducted under identical environmental conditions on a CDAC-PARAM Shavak machine. The hardware features an Intel(R) Xeon(R) Gold 6226R v4 series processor. The testing was performed on a Centos operating system, providing a stable platform for measuring precise computational overhead, including execution times and CPU cycle consumption for the proposed and reference LBC schemes.

B. Performance with Different Parameter Sets

Advantage of MLWR is a same polynomial ring R_q is used for all security levels [2]. To attain a higher level of security, the scheme employs a module with increased rank. Table IV presents proposed scheme's performance and security w.r.t different parameter sets. Parameter sets, denoted as P_1, P_2, and P_3, correspond to increasing security levels.

Results highlight the balance between security and performance. Higher security levels lead to increased computational overhead, reflected in greater CPU time and cycle consumption. The selection of an appropriate parameter set is crucial for achieving the desired security level.

TABLE IV. Evaluation of Proposed Scheme with Various Parameter Sets.

Parameter set and Security level	Operation	Average CPU time (μs)	Average CPU cycles
P_1 Medium : ($q = 7937, N = 256, \eta = 3, k = 2, l = 2$)	Key Generation	103	700
	Encryption	65	200
	Decryption	34	100
P_2 Recommended: ($q = 7937, N = 256, \eta = 2, k = 3, l = 3$)	Key Generation	207	1200
	Encryption	68	200
	Decryption	34	100
P_3 High : ($q = 7937, N = 256, \eta = 2, k = 4, l = 4$)	Key Generation	172	1400
	Encryption	69.48	300
	Decryption	34.12	120

C. Security Estimation and Attack Analysis

Table V provides a comprehensive security profile for the proposed MLWR-based PKE scheme across three distinct parameter sets (P_1, P_2, P_3). This analysis moves beyond heuristic estimates to include rigorous metrics against standard lattice attacks [34].

- Configuration Parameters (k and η):
  - Rank of a module k defines lattice dimension as $N \times k$, where $N = 256$ is fixed.
  - Increasing k from 2 to 4 significantly enhances the security margin against all attack vectors.
  - The secret distribution range η is utilized for sampling the short vector $\mathbf{s}$, directly influencing the hardness of the underlying MLWR problem.
- Attack Metrics:
  - Core-SVP: Represents lower bound of complexity for a Shortest Vector Problem in a sieving regime, which is the foundational hard problem for proposed scheme.
  - Primal (uSVP) Attack: Evaluates the complexity of recovering the secret key by treating the MLWR instance as a unique shortest vector problem in an expanded lattice.
  - Dual Attack: Estimates the computational effort required for an adversary to distinguish the MLWR samples from a uniform distribution by finding short vectors in the dual lattice.
  - BKW Attack: Originally for the Learning Parity with Noise problem, it uses a blockwise approach to reduce problem dimensions while managing induced noise through sort-and-match techniques.
- NIST Security Level Mapping:
  - Parameter set P_1 achieves $2^{118.4}$ security bits, corresponding to NIST Category I.
  - The recommended set P_2 ($k = 3, \eta = 2$) provides $2^{183.4}$ bits of security, satisfying Category III.
  - The high-security set P_3 achieves $2^{253.9}$ bits, aligning with Category V for maximum quantum resistance.

The results demonstrate that by utilizing modulus $q =$

TABLE V. SECURITY ESTIMATES AND NIST MAPPING FOR PROPOSED MLWR PARAMETERS

Parameter Set	k	η	Core-SVP	Primal (uSVP)	Dual Attack	BKW	NIST Category
P_1	2	3	$2^{118.4}$	$2^{134.5}$	$2^{139.9}$	$2^{183.4}$	I
P_2	3	2	$2^{183.4}$	$2^{209.1}$	$2^{219.8}$	$2^{261.9}$	III
P_3	4	2	$2^{253.9}$	$2^{297.3}$	$2^{316.1}$	$2^{340.0}$	V

TABLE VI. COMPARISON OF PROPOSED PKE'S FEATURES WITH OTHER LATTICE-BASED SCHEMES

Features	[10]	[13]	[11]	[37]	Proposed Scheme
Underlying hard problem	MLWE	MLWR	LWE	MLWR	MLWR
Polynomial space	$\mathbb{Z}_q[x]/(x^{256}+1)$	$\mathbb{Z}_q[x]/(x^{256}+1)$	$\mathbb{Z}_q[x]$	$\mathbb{Z}_q[x]/(x^{256}+1)$	$\mathbb{Z}_q[x]/(x^{256}+1)$
Dimensions	1024	1024	640	1024	1024
Modulus	Prime: 3329	Power of 2 : 8192	Power of 2: 32768	Prime:8380417	Prime:7937
Analysis of polynomial multiplication methods	-	-	-	✓	✓
Polynomial multiplication method used	Incomplete NWC NTT	Karatsuba and ToomCook	Naive (schoolbook)	full NWC NTT	Incomplete NWC NTT
Sampling	Rejection Sampling	No need of rejection sampling	Inversion sampling	Rejection Sampling	Rejection Sampling
Secret distribution	Centered binomial distribution	Centered binomial distribution	Uniform distribution	Centered binomial distribution	Centered binomial distribution
Performance analysis	✓	✓	✓	✓	✓
Key sizes [bytes]	sk:1632, pk:800	sk:1568, pk:672	sk:19888, pk:9616	sk:1422, pk:1040	sk:3264, pk:1632
Avg. Execution time (μ s)	Enc:32131 Dec:48317.20	Enc:75, Dec:37.5	Enc:114038 Dec:126289	Enc:75.037 Dec:34.483	Enc:69.48 Dec:34.12
Avg. CPU cycles required	Enc:6870, Dec:962	Enc: 200 , Dec:100	Enc:14800 Dec:15970	Enc: 320 , Dec:200	Enc: 300 , Dec:120
BKW security	$2^{238.3}$	2^{292}	$2^{241.1}$	$2^{270.9}$	$2^{340.0}$

Note: ✓denotes "performed" and - denotes there is no result reported in the particular reference.

7937, the scheme maintains robust security levels while optimizing computational efficiency through the incomplete NTT.

D. Result and Comparison with the other Lattice-Based Schemes

The proposed scheme with high level security requirements has been implemented and tested on a Linux operating system. The CPU time required to execute the proposed key generation, encryption, and decryption functions is measured using Intel® VTune™ Profiler and is 172 μs, 69.48 μs, and 34.12 μs, respectively.

The outcomes of the proposed approach are compared to the other PKE candidates of the NIST's PQC project. NIST requested submission for PQC algorithms [2] in 2016 for the standardization process. After the internal reviews and public feedback, twenty-one LBC candidates were selected in the 1^{st} round. While in the 2^{nd} round only 9 algorithms were selected. Out of 9, 3 are high-speed as well as secure PQC algorithms named Frodo [11], Kyber [10], and Saber [13]. All these algorithms are based on lattice-based hard problems. Execution time of the proposed PKE is compared with these schemes.

Table VI presents a comparative analysis against various features of the proposed MLWR-based PKE scheme with other well-known LBC schemes, focusing on the underlying cryptographic problems, polynomial spaces, performance characteristics (such as execution time and key sizes), and security. Key differences include the polynomial multiplication methods used (e.g., NTT vs. Karatsuba), security analysis, and performance metrics such as execution time and CPU cycles.

The proposed scheme's public key and ciphertext sizes are both smaller than Frodo's scheme. Size of the secret key is smaller than the Frodo. Also, the average execution time and average CPU cycles required for the proposed scheme's encryption and decryption are comparable to that of the most efficient candidate, SABER. The average time needed to encrypt a 256-byte message is equivalent to that of SABER, whereas the average time required to decrypt a message is less than that of SABER.

The proposed scheme stands out for its high BKW security and relatively low execution times and CPU cycles, offering a superior balance of security, performance, and computational efficiency. This makes it a preferred choice for implementing post-quantum cryptographic protocols

VII. CONCLUSION

The proposed scheme attains a better performance compared to other lattice-based schemes. Also, this scheme provides prominent hardness and security, ingrained in the LWR variant. It has a faster and simpler encryption and decryption process. The key sizes are very compact. As the proposed scheme is based on a lattice-based hard problem, MLWR, it can be secure against quantum attacks.

Analysis highlighted the importance of selecting an appropriate modulus q to balance performance and security. Among the evaluated moduli, $q = 7937$ emerged as the most efficient

choice, providing a desirable trade-off between computational efficiency and security.

Proposed MLWR-based PKE scheme using incomplete NTT stands out due to its high security and efficient performance, making it an optimal choice for practical and secure post-quantum cryptographic applications.

Future work could explore further optimizations in parameter selection and the adaptation of the incomplete NTT methodology to other post-quantum cryptographic schemes, enhancing their applicability to a wider range of security-critical and resource-limited scenarios.

REFERENCES

[1] O. Regev, "An efficient quantum factoring algorithm," *J. ACM*, Dec. 2024, just Accepted. [Online]. Available: https://doi.org/10.1145/3708471

[2] G. Alagic, M. Bros, P. Ciadoux, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, Y.-K. Liu, C. Miller *et al.*, *Status report on the fourth round of the nist post-quantum cryptography standardization process*. US Department of Commerce, National Institute of Standards and Technology, 2025.

[3] T. Jamil, "From bits to qubits: Comparative insights into classical and quantum computing systems," *International Journal of Advanced Computer Science & Applications*, vol. 16, no. 12, 2025.

[4] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, ser. STOC '96. New York, NY, USA: Association for Computing Machinery, 1996, p. 99–108. [Online]. Available: https://doi.org/10.1145/237814.237838

[5] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM (JACM)*, vol. 56, no. 6, sep 2009. [Online]. Available: https://doi.org/10.1145/1568318.1568324

[6] A. Langlois, D. Stehlé, C. C. by Padro A Langlois, D. Stehlé, and A. Langlois, "Worst-case to average-case reductions for module lattices," *Designs, Codes and Cryptography 2014 75:3*, vol. 75, no. 3, pp. 565–599, feb 2014.

[7] A. Banerjee, C. Peikert, and A. Rosen, "Pseudorandom functions and lattices," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, vol. 7237 LNCS. Springer, 2012, pp. 719–737.

[8] P. Newton and S. Richelson, "A lower bound for proving hardness of learning with rounding with polynomial modulus." Berlin, Heidelberg: Springer-Verlag, 2023, p. 805–835. [Online]. Available: https://doi.org/10.1007/978-3-031-38554-4_26

[9] J. Alwen, S. Krenn, K. Pietrzak, and D. Wichs, "Learning with rounding, revisited," in *Advances in Cryptology – CRYPTO 2013*, R. Canetti and J. A. Garay, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 57–74.

[10] T. L. Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz and D. S. Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, "CRYSTALS-Kyber Algorithm Specifications and Supporting Documentation," *NIST PQC Round*, vol. 2, no. 4, pp. 1–32, 2021. [Online]. Available: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf

[11] L. Glabush, P. Longa, M. Naehrig, C. Peikert, D. Stebila, and F. Virdia, "FrodoKEM: A CCA-secure learning with errors key encapsulation mechanism," 2025. [Online]. Available: https://eprint.iacr.org/2025/1861

[12] J. Hoffstein, J. Pipher, and J. H. Silverman, "NTRU: A Ring-Based Public Key Cryptosystem," in *International Algorithmic Number Theory Symposium*. Springer, 1998, pp. 267–288.

[13] A. Karmakar, S. S. Roy, and F. Vercauteren, "SABER: Mod-LWR based KEM (Round 3 Submission)," *NIST PQC Round*, pp. 1–2, 2020. [Online]. Available: https://eprint.iacr.org/2018/230.pdf

[14] S. Bai, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé, "Crystals-dilithium: Algorithm specifications and supporting documentation (version 3.1)," *NIST Post-Quantum Cryptography Standardization Round*, vol. 3, 2021.

[15] Y. Li, J. Zhang, Z. Jin, and W. Qiao, "A multi-strategy improved horned lizard optimization algorithm and its application in engineering optimization," *International Journal of Computational Intelligence Systems*, vol. 18, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:278251547

[16] H. Cheng, J. Großschädl, P. B. Rønne, and P. Y. Ryan, "Avrntru: Lightweight ntru-based post-quantum cryptography for 8-bit avr microcontrollers," pp. 1272–1277, 2021.

[17] D. J. Bernstein, B. B. Brumley, M.-S. Chen, C. Chuengsatiansup, T. Lange, A. Marotzke, B.-Y. Peng, N. Tuveri, C. van Vredendaal, and B.-Y. Yang, "Ntru prime: round-3 updates," *National Institute of Standards and Technology (NIST)*, 2020.

[18] Y. Sugizaki and D. Takahashi, "Improved implementation of number theoretic transform on nvidia gpu with tensor cores," in *Proceedings of the Supercomputing Asia and International Conference on High Performance Computing in Asia Pacific Region*, ser. SCA/HPCAsia '26. New York, NY, USA: Association for Computing Machinery, 2026, p. 142–152. [Online]. Available: https://doi.org/10.1145/3773656.3773673

[19] Z. Guitouni, N. Ammar, and M. Machhout, "An efficient hardware implementation of sha-3 using 3d cellular automata for secure wireless sensor networks," *International Journal of Information Security*, vol. 24, no. 3, p. 116, 2025.

[20] I. Prots' ko and A. Gryshchuk, "Implementing montgomery multiplication to speed-up the computation of modular exponentiation of multi-bit numbers," *Cybernetics and Systems Analysis*, vol. 60, no. 5, pp. 826–833, 2024.

[21] Y. Wang, Y. Zhao, and M. Wang, "On the Hardness of Ring/Module/Polynomial LWR Problems," *Cryptology ePrint Archive*, 2021. [Online]. Available: http://www.eprint.mirror.cypherpunks.su/2021/1026/1628236547.pdf

[22] T. Pöppelmann, T. Oder, and T. Güneysu, "High-performance ideal lattice-based cryptography on 8-bit atxmega microcontrollers," in *Progress in Cryptology – LATINCRYPT 2015*, K. Lauter and F. Rodríguez-Henríquez, Eds. Cham: Springer International Publishing, 2015, pp. 346–365.

[23] Y. Zhu, Z. Liu, and Y. Pan, "When NTT Meets Karatsuba: Preprocess-then-NTT Technique Revisited," in *International Conference on Information and Communications Security*, vol. 12919 LNCS. Springer, Cham, sep 2021, pp. 249–264.

[24] A. Satriawan, I. Syafalni, R. Mareta, I. Anshori, W. Shalannanda, and A. Barra, "Conceptual review on number theoretic transform and comprehensive review on its implementations," *IEEE Access*, vol. 11, pp. 70 288–70 316, 2023.

[25] H. Kondo, "Cooley-tukey fft over \mathbb {Q}_p via unramified cyclotomic extension," *arXiv preprint arXiv:2505.02509*, 2025.

[26] T. B. Rodrigues, A. Rodrigues, M. Goulão, P. Tomás, and L. Sousa, "Accelerating ntt with risc-v vector extension for fully homomorphic encryption," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2025, no. 4, pp. 711–736, 2025.

[27] A. Pouly and Y. Shen, "Solving the shortest vector problem in $2^{0.63269n+o(n)}$ time on random lattices," Cryptology ePrint Archive, Paper 2024/1805, 2024. [Online]. Available: https://eprint.iacr.org/2024/1805

[28] D. Micciancio and S. Goldwasser, *Complexity of lattice problems*, ser. The Springer International Series in Engineering and Computer Science. New York, NY: Springer, Oct. 2002, vol. 671.

[29] M. Ajtai and C. Dwork, "Public-key cryptosystem with worst-case/average-case equivalence," in *Conference Proceedings of the Annual ACM Symposium on Theory of Computing*. ACM, 1997, pp. 284–293.

[30] J. Sharafi and H. Daghigh, "A ring-lwe-based digital signature inspired by lindner–peikert scheme," *Journal of Mathematical Cryptology*, vol. 16, pp. 205 – 214, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:251369134

[31] P.-A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Prest, T. Ricosset, G. Seiler, W. Whyte, Z. Zhang *et al.*, "Falcon: Fast-fourier lattice-based compact signatures over ntru," *Submission to the NIST's post-quantum cryptography standardization process*, vol. 36, no. 5, pp. 1–75, 2018.

[32] K. Arai, "Method for mission analysis using tot-based prompt technology utilized generative ai satellite mission analysis for sagansat-0 of remote sensing satellite." *International Journal of Advanced Computer Science & Applications*, vol. 15, no. 9, 2024.

[33] Z. Zheng, A. Wang, and L. Qin, "Rejection Sampling Revisit: How to Choose Parameters in Lattice-Based Signature," *Mathematical Problems in Engineering*, vol. 2021, 2021.

[34] K. Boudgoust, C. Jeudy, A. Roux-Langlois, and W. Wen, "On the hardness of module learning with errors with short distributions: K. boudgoust et al." *Journal of Cryptology*, vol. 36, no. 1, p. 1, 2023.

[35] U. Banerjee, T. S. Ukyab, and A. P. Chandrakasan, "Sapphire: A configurable crypto-processor for post-quantum lattice-based protocols_2019," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, no. 4, p. 17–61, Aug. 2019. [Online]. Available: https://tches.iacr.org/index.php/TCHES/article/view/8344

[36] M. S. Turan, M. S. Turan, K. McKay, D. Chang, L. E. Bassham, J. Kang, N. D. Waller, J. M. Kelsey, and D. Hong, *Status report on the final round of the NIST lightweight cryptography standardization process*. US Department of Commerce, National Institute of Standards and Technology, 2023.

[37] A. A. Pandit and A. Mishra, "Efficient implementation of post quantum mlwr-based pke scheme using ntt," *Computers and Electrical Engineering*, vol. 118, p. 109358, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790624002866

AUTHORS' PROFILE

Anupama Arjun Pandit (Student Member, IEEE) is currently pursuing Ph.D. degree in Computer Science and Engineering Department, at Defense Institute of Advanced Technology, Pune, India. She has completed her M. Tech. in CSE-IT from Walchand College of Engineering, Sangli, Maharashtra, India. Her research interest includes Cryptography and Post Quantum Cryptography (Code-based Cryptography and Lattice-based Cryptography). Her contribution to the current project is implementation, testing, analysis, and interpretation of the data, algorithm, and results.

Arun Mishra is working as an Associate Professor in Computer Science and Engineering Department at Defence Institute of Advanced Technology, Pune, India. He completed his Ph.D. from MNIT, Allahabad, India. He has more than 40 research publications in peer-reviewed journals & international conference proceedings. His areas of interest are Cryptography, Blockchain Technology, Post Quantum Cryptography (Code-based Cryptography, Lattice-based Cryptography and Hash-based Cryptography), Secure Software Engineering, and Secure Systems Design (isolation/ secure execution/ secure computations). In the current project, he contributed to the development of the concept and technique.

Abnormal State Detection of Industrial Tools Based on the MGC-YOLOv8 Algorithm

Guan Yang[1], Xiang Cheng[2], Miao Wang[3]*, Ziyue Huang[4], Hao Tang[5], Yujun Chen[6]
School of Artificial Intelligence, Zhongyuan University of Technology, Zhengzhou, China[1,2,4]
School of Computer Science, Zhongyuan University of Technology, Zhengzhou, China[2,4,5]
School of Software, Henan University of Engineering, Zhengzhou, China[3]
Faculty of Engineering, Architecture and Information Technology, The University of Queensland, Brisbane, Australia[6]

***Abstract*—As intelligent manufacturing advances toward precision and automation, cutting tool condition critically impacts product quality, equipment safety, and production efficiency. Anomalies like wear, chipping, or fracture cause workpiece scrapping and machine failure, demanding efficient online monitoring. Traditional manual or image-based methods suffer from low accuracy in complex environments. Although deep learning excels in industrial defect detection, existing end-to-end detectors exhibit insufficient recall and localization precision for millimeter-scale cracks and blurred tool boundaries. To address these challenges, we propose MGC-YOLOv8, an enhanced framework built upon the YOLOv8 backbone. A Multi-Scale Edge-Dual Fusion (MSEDF) module is introduced to integrate feature maps across different scales, thereby strengthening the detection of minor defects. Furthermore, a Global-to-Local Spatial Aggregation (GLSA) module enriches feature representations by simultaneously capturing global context and local details. A Convolutional Block Attention (CBAM) module is embedded upstream of the prediction head to adaptively highlight critical features in both channel and spatial dimensions.Although the integration of MSEDF, GLSA, and CBAM introduces a marginal runtime overhead and a slight increase in parameter count, the optimized architecture preserves real-time inference speeds that fully satisfy the requirements of industrial inspection systems. Experimental results demonstrate that MGC-YOLOv8 substantially outperforms the baseline YOLOv8n, achieving 88.1% precision, 87.9% recall, 92.5% mAP@0.5 and 69.6% mAP@0.5:0.95 on our test set.**

***Keywords*—Object detection; surface defect detection; YOLOv8**

I. Introduction

During machine-tool processing, real-time monitoring of tool condition can effectively prevent workpiece scrap and machine-tool damage [1]. Maintaining good tool condition is crucial for CNC machining quality and productivity [2], making Tool Condition Monitoring (TCM) an essential part of modern manufacturing [3]. Traditional inspection—relying on skilled operators for visual checks or intermittent sampling—is inefficient and vulnerable to subjectivity and fatigue, and thus unsuitable for large-scale continuous online monitoring [4]. Early automated TCM used force and acceleration sensors (cutting force, acoustic emission, vibration) with time-frequency analysis to infer wear [5], [6], [7], [8]. However, fluctuations in machining parameters and workshop noise often undermined their robustness, causing missed detections and false alarms.

Vision-based tool anomaly detection has emerged as a significant research focus alongside rapid advances in deep learning [9]. Convolutional neural networks (CNNs) [10] automatically extract semantic features through end-to-end training, yielding substantial gains in accuracy and real-time performance. The YOLO family adopts a single-stage framework with multi-scale feature-pyramid fusion, achieving a favorable speed–accuracy trade-off for industrial defect detection [11], [12]. However, standard downsampling and fusion in these detectors can attenuate or lose millimetre- and sub-millimetre-scale defects (fine cracks, chipping, blurred edges) under complex lighting and cluttered backgrounds, degrading recall and localization precision.

To address these issues, we propose MGC-YOLOv8. We replace the original C2f with a Multi-Scale Edge-Dual Fusion (MSEDF) module to better capture edge and fine-detail cues of subtle anomalies, enhancing local feature representation. A Global-to-Local Spatial Aggregation (GLSA) module uses a dual-branch decoupled design to extract global context and local details and fuses them via dynamic gating to improve robustness in complex scenes [13]. A Convolutional Block Attention Module (CBAM) is embedded immediately before the prediction head to adaptively reweight channel and spatial responses and focus on anomaly-sensitive regions [14], [15]. Experimental results show that MGC-YOLOv8 yields substantial accuracy gains over the YOLOv8n baseline.

The main contributions of this study are as follows:

- We designed the MSEDF to replace YOLOv8's C2f, explicitly capturing multi-scale edges and fine-detail cues of micro anomalies (e.g., cracks, wear), thereby enhancing local feature representation.
- We introduce the GLSA module, a dual-branch design that separately extracts local details and global semantics, then adaptively fuses them via dynamic gating — improving the robustness and efficiency of multi-scale feature fusion.
- We embed the CBAM before the prediction head before the prediction head, where channel–spatial attention adaptively reweights features to emphasize anomaly-sensitive regions and improve detection accuracy.

*Corresponding author.

- An industrial tool dataset with diverse real surface defects was constructed to support model training and evaluation, verifying the practicality and effectiveness of the proposed method.

The remainder of this study is organized as follows: Section II reviews related work in tool-condition monitoring and vision-based defect detection. Section III details the architecture of the proposed MGC-YOLOv8 model and the design of its key modules. Section IV describes the experimental setup, dataset construction and evaluation metrics. Section V presents the experimental results and comparative analysis. Finally, Section VI concludes the study and outlines future research directions.

II. Related Work

In recent years, tool condition detection has shifted from empirical inspection and signal analysis to image-based and deep-learning approaches. Early methods using physical signals (e.g., cutting force, vibration) with wavelet or Fourier analysis were easily affected by noise, parameter fluctuations, and nonlinear factors, leading to unstable results. Machine vision enabled direct acquisition of tool-surface images for defect identification, offering high spatial resolution but limited adaptability to complex backgrounds. With the advance of deep learning, CNN-based object detection [16]—featuring automatic feature extraction and end-to-end training—has become a key means to enhance accuracy and real-time performance, addressing the limitations of traditional techniques and driving intelligent, automated tool-condition monitoring in modern manufacturing.

Tool Condition Monitoring (TCM) can be divided into indirect monitoring based on physical signals and direct monitoring based on visual information. Early studies emphasized indirect methods, using cutting-force and vibration signals with time–frequency features to classify wear states. For example, Zhu et al. [17] reviewed wavelet transform for TCM, noting its multi-resolution ability to extract non-stationary features, but also its dependence on noise-model assumptions, redundant coefficients, and empirical thresholds, which limit high-frequency fault extraction and real-time decision-making. Zhou et al. [18] proposed vibration-based monitoring via WTMM and SVM, showing sensitivity to wear-transition stages, yet suffering from noise interference and empirical parameter settings. He et al. [19] introduced temperature-signal-based prediction using SSAEs, achieving high accuracy but relying on empirical hidden-node reduction, restricting real-time adaptability. To address small-sample challenges, Zhou et al. [20] designed MEGNN to encode cutting-force signals as recursive graphs, improving classification but at the cost of high computational complexity in multi-scale convolution and graph updates, limiting real-time use. Overall, while such signal-based methods reveal intrinsic links between signals and wear, they remain vulnerable to parameter fluctuations and environmental noise. Furthermore, the inherently low signal-to-noise ratio of acoustic emissions easily masks minor wear features, often necessitating complex denoising procedures [21].

With advances in image processing, direct monitoring using machine vision has attracted increasing attention, offering high spatial resolution and intuitive defect visualization without multi-source signal preprocessing. Guo et al. [22] employed adaptive GrabCut with improved Gaussian mixture modeling and nonlinear filtering to isolate tiny edge defects. However, this approach required manual tuning, was sensitive to illumination and background variations, and demanded machining interruption. Yu et al. [23] developed a pixel-based wear quantification approach with local-variance segmentation and image enhancement, achieving accuracy but at the cost of redundant parameters, high computation, and limited capability for sub-millimeter defects in cluttered scenes. Tang et al. [24] proposed MiSLASND, a ResNet32-based deep-transfer model with class-balance regularization, which improved accuracy and efficiency but lacked defect localization and relied on costly two-stage training. Overall, image-based TCM avoids the noise susceptibility of signal-based methods but still faces challenges in robustness, real-time applicability, and fine-scale defect detection.

Indirect monitoring via physical signals can capture tool-wear trends; however, it is highly sensitive to noise and varying conditions. Conversely, while image-based monitoring offers intuitiveness, it struggles with minor-defect detection in complex backgrounds. To address these issues, researchers have explored more efficient architectures. The YOLO (You Only Look Once) framework [25], as a single-stage detector, enhances small-target recognition by optimizing Feature Pyramid Networks (FPN) [26] and Path Aggregation Networks (PAN) [27], achieving both accuracy and speed [28]. This balance of real-time performance and precision makes YOLO highly promising for industrial defect detection, particularly in tool anomaly monitoring.

To address the limitations of traditional image-based monitoring in fine-grained feature extraction, multi-scale fusion, and robustness to complex backgrounds, we propose an enhanced YOLOv8-based method for industrial tool anomaly detection. The network integrates MSEDF for edge/detail preservation, GLSA for adaptive global–local fusion, and CBAM for channel–spatial attention, thereby improving sensitivity to minute defects and overall detection accuracy in complex industrial scenarios.

III. Methodology

A. MGC-YOLOv8

Although YOLOv8 offers high efficiency in real-time detection, it struggles with industrial tool anomalies: microcracks and slight abrasion are often lost during downsampling, while uneven illumination, glare, and cluttered backgrounds further obscure fine details. Its global fusion strategy also fails to preserve local edge and texture information, leading to false positives on sub-millimetre defects. Thus, the standard YOLOv8 falls short of the precision and robustness required in industrial applications.

As shown in Fig. 1, we propose the MGC-YOLOv8 model, which incorporates targeted enhancements to the original YOLOv8 architecture to bolster both local detail representation and multi-scale feature fusion. In the Backbone and Neck stages, we replace the standard C2f modules with our MSEDF module, which significantly improves the extraction of edge features associated with subtle anomalies—such as fine surface

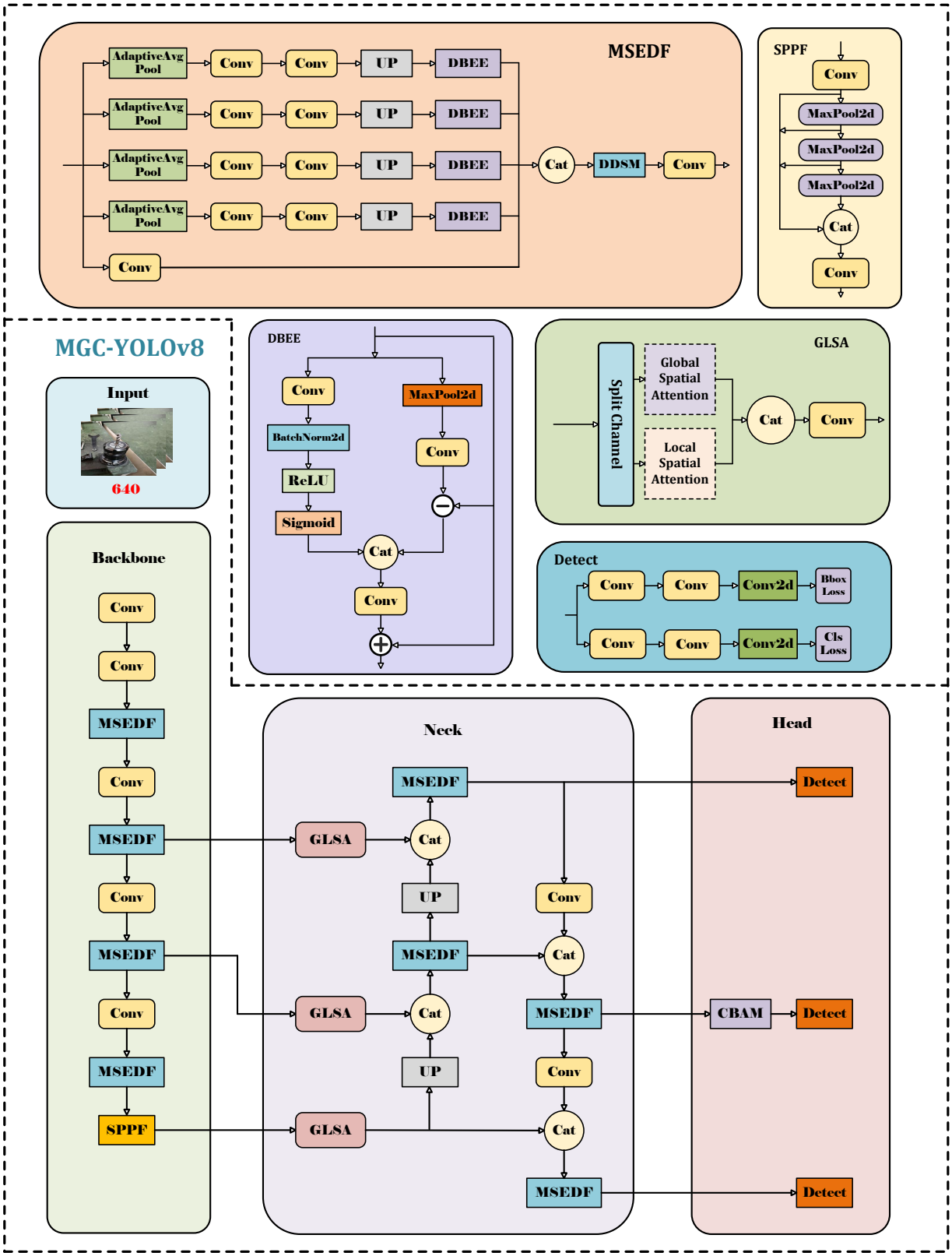


Fig. 1. MGC-YOLOv8 structure.

cracks and wear on cutting tools. Within the Neck, the GLSA module is introduced. Its dual-branch structure separately captures local detail cues and global contextual semantics. Meanwhile, a dynamic gating mechanism adaptively fuses these streams to enhance information flow across scales and improve robustness. Just before the detection head, we embed a CBAM to dynamically modulate feature responses along both channel and spatial dimensions, directing the model's focus toward critical anomalous regions and thereby further increasing detection precision. By synergistically combining MSEDF's micro-defect perception, GLSA's enriched multi-scale interactions, and CBAM's refined attention, MGC-YOLOv8 achieves a substantial improvement in accurately detecting industrial tool abnormalities.

B. Multi-Scale Edge-Dual Fusion Module

In industrial tool anomaly detection, the original C2f module in YOLOv8 exhibits clear limitations when handling local irregularities—such as cracks and surface wear—because its global feature-fusion strategy fails to capture fine edge and detail information. To overcome this, we introduce the MSEDF module, illustrated in Fig. 2. MSEDF employs adaptive average pooling to generate feature maps at multiple scales, each of which is processed by a Dual-Branch Edge Enhancer(DBEE) to strengthen edge and detail cues, thereby improving detection accuracy. Furthermore, we integrate a Dual Domain Selection Mechanism (DSM) [29], which adaptively focuses on the most task-relevant regions across these scales by selectively attending to higher-importance features. This targeted selection markedly enhances feature discriminability and boosts overall model performance.

The Dual-Branch Edge Enhancer optimizes edge features via a cooperative workflow of high-frequency residual enhancement and detail feature amplification. In the first branch, the high-frequency residual feature $\mathbf{R}$ is extracted as follows. Given an input feature map $\mathbf{X} \in \mathbb{R}^{H\times W\times c}$, a local extremum is obtained by applying MaxPool, then convolved with a 3×3 kernel $\mathbf{W}_2$. A sigmoid activation $\sigma(\cdot)$ converts the result into a spatial attention weight in $[0,1]$. Subtracting this weighted low-frequency component from the original feature yields the high-frequency residual. The calculation is shown in Eq. (1):

$$\mathbf{R} \quad = \quad \mathbf{X} - \sigma\big(\mathbf{W}_2 * \mathrm{MaxPool}(\mathbf{X})\big) \tag{1}$$

Meanwhile, the second branch extracts the fine-detail feature $\mathbf{D}$ by applying a 3×3 convolution $\mathbf{W}_3$, followed by BatchNorm and ReLU (denoted as $\varnothing(\cdot)$). The calculation is given in Eq. (2):

$$\mathbf{D} = \varnothing(\mathbf{W}_3) \tag{2}$$

The outputs of both branches are concatenated along the channel dimension and then channel-compressed via a 1×1 convolution $\mathbf{W}_1$. A residual connection adds the original feature $\mathbf{X}$ to preserve baseline information. The final enhanced edge feature $\mathbf{F}_{\mathrm{DBEE}}$ is therefore computed as shown in Eq. (3):

$$\mathbf{F}_{\mathrm{DBEE}} = \mathbf{X} + \mathbf{W}_1\left[(\mathbf{X} - \sigma(\mathbf{W}_2 * \mathrm{MaxPool}(\mathbf{X}))) \parallel \varnothing(\mathbf{W}_3)\right] \tag{3}$$

where, "$\parallel$" denotes concatenation along the channel axis.

MSEDF decomposes the input feature $\mathbf{X}$ into multi-scale space. Adaptive average pooling $\mathrm{Proj}_s(\mathbf{X}) = \mathit{AdaptiveAvgPool}_{s\times s}$ is used to generate sub-features of resolution $s \times s$ to capture the detail information of the tool surface and edge at different resolutions.

Each sub-feature is enhanced by an independent Dual-BranchEdgeEnhancer for local edge details, and then restored to the original resolution by bilinear interpolation $\mathit{Interp}_{H\times W}$ to obtain the multi-scale feature $\mathbf{F}_s^{\uparrow} \in \mathbb{R}^{C\times 1\times 1}$.

All scale features are concatenated with the local 3×3 convolution feature $\mathbf{W}_5$ into $\mathbf{F}_{\mathrm{concat}} \in \mathbb{R}^{H\times W\times 4C}$, and the dual domain selection mechanism $\mathrm{DSM}(\cdot)$ is input to dynamically balance the feature weights.

Finally, the channel is compressed through the 3×3 convolution kernel $\mathbf{W}_4$, and the optimized feature $\mathbf{Y}$ is output. The overall calculation is shown in Eq. (4):

$$\mathbf{Y} = \mathbf{W}_4 * \mathrm{DSM}\left(\left[\mathbf{W}_5 * \mathbf{X}; \bigoplus_{s\in S} \mathit{Interp}_{H\times W}\left(\mathbf{F}_{\mathrm{DBEE}}(\mathrm{Proj}_s(\mathbf{X}))\right)\right]\right) \tag{4}$$

In summary, the MSEDF module employs a multi-scale strategy to capture tool-surface details at varying resolutions and leverages its embedded DBEE to strengthen local edge and texture representations. Subsequently, an adaptive attention mechanism fuses information across these scales, thereby substantially improving the detection of minute anomalies such as micro-cracks and wear.

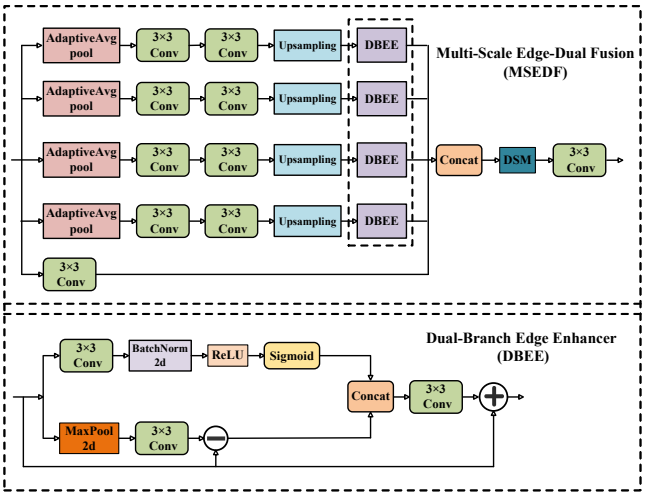


Fig. 2. Structural diagram of the MSEDF network.

C. Global-to-Local Spatial Aggregation Module

Traditional YOLOv8 suffers high false-positive rates and weakened small-object features due to repeated downsampling

and complex-background interference, hindering precise localization of chipping, micro-cracks and similar defects. To address this, a GLSA module is incorporated. GLSA adopts a dual-branch design: a local branch using depth-wise separable convolutions combined with Local Spatial Attention (LSA) to reinforce minute-defect representation, and a global branch employing Global Spatial Attention (GSA) to capture holistic scene context. A dynamic gating mechanism adaptively fuses the two streams, improving multi-scale integration and robustness to cluttered backgrounds, and thus enhancing detection accuracy for fine-grained tool defects.

As shown in Fig. 3, each multi-scale input is split along the channel dimension into two sub-features and passed to the global and local branches. In our implementation, this split ratio is fixed at 0.5 (i.e., splitting the channels equally) to maintain a balance between computational efficiency and feature representation, rather than being dynamically learned.In the global branch, channel reduction is followed by computation of a spatial attention map, which is fused with the sub-feature and processed by an MLP to capture holistic semantics. The local branch uses standard and depth-wise separable convolutions to extract fine details, produces a local spatial attention map, and fuses it with the sub-feature to strengthen local cues. Finally, both branch outputs are combined via convolution and adaptively integrated by a dynamic gating mechanism to yield the module output.

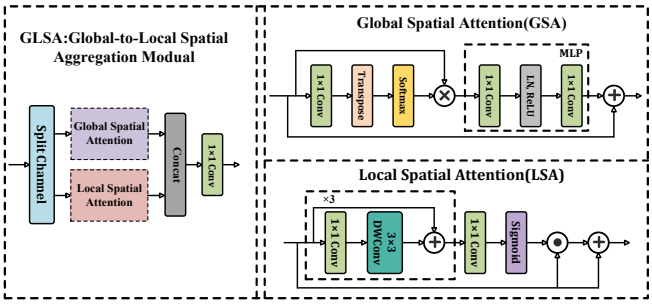


Fig. 3. GLSA module network structure diagram.

D. Convolutional Block Attention Module

Industrial tool-surface anomalies (e.g., cracks, wear) are minute, irregular, and low-contrast. Furthermore, in complex industrial scenes, uneven illumination, specular reflections, and background noise often attenuate these cues during feature extraction. To mitigate this, a CBAM is embedded immediately before the YOLOv8 detection head (see Fig. 4). By applying channel-wise and spatial attention, CBAM adaptively recalibrates feature responses to emphasize anomaly-sensitive regions and suppress irrelevant information, thereby improving detection accuracy and robustness.

The Channel Attention Module (CAM) in CBAM (see Fig. 5) aggregates channel-wise statistics using global average and max pooling, feeds both descriptors through a shared MLP, and sums them element-wise. A sigmoid activation converts the result into channel attention weights that adaptively reweight channels—amplifying informative channels for anomaly detection while suppressing irrelevant or redundant responses.

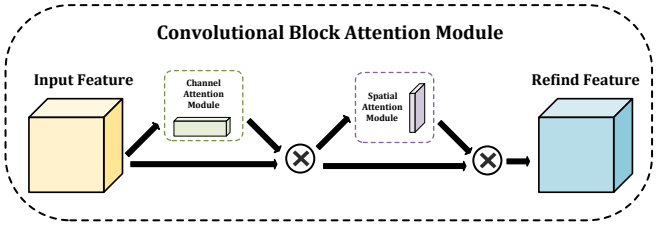


Fig. 4. Convolutional block attention module structure diagram.

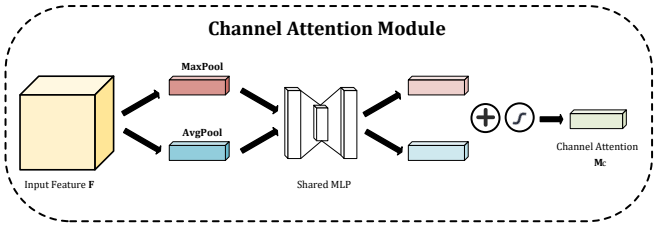


Fig. 5. Channel attention module structure diagram.

The Spatial Attention Module (SAM) processes the channel-refined feature map (see Fig. 6). It applies channel-wise average and max pooling to produce two 2D maps, concatenates them, and uses a convolution to generate a spatial attention map that highlights anomaly-prone regions and guides the network to focus on critical spatial locations.

IV. Experiment Preparation

A. Dataset Production

A private industrial dataset was collected from the production line of a factory in Yinchuan under natural operating conditions, covering different machining stages (roughing and finishing) and multiple material-cutting scenarios to ensure diversity. Cameras recorded cutting tools in four representative states—normal, chipped, broken and worn—resulting in 4,285 raw images. To guarantee data quality, a two-step screening was performed: automatic checks of resolution and file integrity, followed by manual inspection for clarity, exposure, motion blur, occlusion, and background interference. Low-quality samples were discarded, and 3,131 high-quality images were retained. All valid images were resized uniformly to

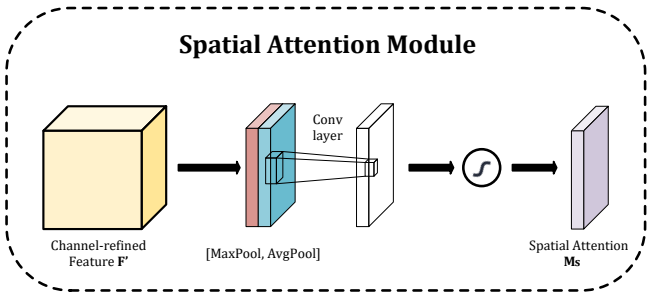


Fig. 6. Spatial attention module structure diagram.

640×640 pixels to preserve tool details while aligning with deep-learning input standards.

Annotations were produced using the LabelMe platform by technicians experienced in tool-state characteristics. Each image was annotated with bounding boxes and classified into four categories—chipped edge, broken tool, wear, and normal—as illustrated in Fig. 7. To ensure reliability, multiple rounds of review and cross-checking were conducted, with ambiguous cases resolved through discussion. Finally, the dataset was divided into training, validation, and test subsets in an 8:1:1 ratio, enabling effective parameter learning, model selection, and independent evaluation, and providing a solid basis for subsequent experiments.



Fig. 7. Sample graph of the dataset.

B. Experimental Environment

The experiments were implemented in the PyTorch framework (v2.4.1) with NVIDIA CUDA 12.2, on a hardware platform equipped with an Intel i7-10700K CPU and an NVIDIA RTX A5000 GPU. Training was performed for 300 epochs using the SGD optimizer with an initial learning rate of 0.01, a batch size of 16, and an input image resolution of 640×640. Other hyperparameters followed the official YOLOv8 settings. Since defect detection is highly sensitive to tool orientation and lighting conditions, data augmentation strategies were explicitly employed during the training phase. Specifically, we natively leveraged the robust data augmentation pipeline embedded in the YOLOv8 framework. These techniques primarily include Mosaic augmentation, random scaling, translation, and HSV color jittering. These strategies effectively expanded the diversity of the training samples, mitigating the impact of complex industrial backgrounds and varying illumination. Furthermore, to ensure a fair comparison, all competing models were evaluated under identical dataset conditions and initialized with standard pretrained weights rather than being trained from scratch.Throughout the study, a consistent software and hardware environment was maintained to ensure reproducibility.

C. Evaluation Index

To evaluate model performance in detecting tool anomalies (e.g., wear, fracture, vibration), Precision, Recall, F1-Score, and mean average precision (mAP) were used, with mAP reported as mAP@0.5 and mAP@0.5:0.95. These metrics jointly assess localization and classification accuracy, providing a quantitative basis for the reliability and applicability of tool-state detection.

Precision refers to the proportion of tool samples predicted as anomalous that are actually anomalous, reflecting the model's accuracy in identifying abnormal tool states.It is calculated as shown in Eq. (5):

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

Recall measures the proportion of truly anomalous tool samples that the model successfully detects, with its calculation given in Eq. (6):

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

The F1-Score, defined as the harmonic mean of precision and recall, balances the model's accuracy in predicting anomalous states with its ability to minimize missed detections. Its calculation is given in Eq. (7):

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{7}$$

Mean Average Precision (mAP) is a standard metric in object detection, defined as the mean of Average Precision (AP) values across all classes, where AP corresponds to the area under the precision–recall curve. Specifically, mAP@0.5 denotes the mean AP at an IoU threshold of 0.5, as formulated in Eq. (8):

$$mAP@0.5 = \frac{1}{C} \sum_{C=1}^{C} AP_{0.5}^{(c)} \tag{8}$$

To evaluate detection performance across different overlaps, mAP@0.5:0.95 is used, defined as the mean AP over IoU thresholds from 0.5 to 0.95, as shown in Eq. (9):

$$mAP@0.5:0.95 = \frac{1}{T} \sum_{t=1}^{T} \left(\frac{1}{C} \sum_{c=1}^{C} AP_{IoU_t}^{(c)} \right) \tag{9}$$

V. Experiment and Analysis

A. CBAM Routing Location Experiment

The Convolutional Block Attention Module (CBAM) integrates channel and spatial attention to enhance feature representations. In object detection, however, the effectiveness of such attention depends strongly on its placement within the multi-scale feature-fusion pipeline. This study therefore examines how different CBAM insertion points affect feature extraction, semantic representation, and detection performance [30].

Table I outlines the 26-layer structure of MGC-YOLOv8, where the Index column indicates each module's position, Module specifies its type, and Route denotes input-layer indices. The three detection heads take inputs from layers 18, 21, and 24. Since CBAM placement before these heads may affect performance, comparative experiments were conducted, and the results are shown in Table II.

Since the YOLOv8 detection head has only three output layers, there are seven possible CBAM insertion schemes, as detailed in Table II. The experimental results clearly show that embedding CBAM at positions (24, 25) yields the best overall performance. Accordingly, in the MGC-YOLOv8 model, CBAM is inserted at routing positions (24, 25).

TABLE I. NETWORK STRUCTURE INFORMATION OF MGC-YOLOV8

Index	Module	Route	Index	Module	Route
0	Conv		13	Upsample	12
1	Conv	0	14	Concat	13,11
2	MSEDF	1	15	MSEDF	14
3	Conv	2	16	Upsample	15
4	MSEDF	3	17	Concat	16,10
5	Conv	4	18	MSEDF	17
6	MSEDF	5	19	Conv	18
7	Conv	6	20	Concat	19,15
8	MSEDF	7	21	MSEDF	20
9	SPPF	8	22	Conv	21
10	GLSA	4	23	Concat	22,12
11	GLSA	6	24	MSEDF	23
12	GLSA	9	25	Predict	18,21,24

B. Ablation Experiments

To assess the effectiveness of our collaborative multi-module optimization, we conducted ablation studies on the YOLOv8n baseline, evaluating seven configurations: individual modules, all pairwise combinations, and the full MSEDF+GLSA+CBAM ensemble. Results are summarized in Table III.

Table IV shows that each module improves YOLOv8n performance. MSEDF alone increases precision, recall, F1-Score, and mAP metrics, while CBAM further boosts precision and F1-Score. GLSA alone provides modest gains but, combined with MSEDF or CBAM, significantly enhances performance by capturing global semantic context. Integrating all three modules achieves the best results: 88.1% precision, 87.9% recall, 88.0% F1-Score, 92.5% mAP@0.5, and 69.6% mAP@0.5:0.95, confirming their complementary roles and the effectiveness of the multi-module strategy.

C. Comparative Experiment

To validate the efficacy of the proposed MGC-YOLOv8 algorithm for industrial tool-anomaly detection, we conducted comparative experiments with several state-of-the-art object-detection frameworks, including multiple YOLO series [31], [32], [33], [34], [35], RT-DETR [36], Mamba-YOLO [37], and Hyper-YOLO [38]. To ensure statistical reliability and address potential performance variability, the evaluation metrics in Table IV are reported as the average along with the standard deviation over 3 independent runs. As shown in Table IV, MGC-YOLOv8 achieves the highest $88.1\% \pm 0.2\%$ Precision and $87.9\% \pm 0.2\%$ Recall, outperforming the next-best model, Hyper-YOLO, by about 1.5 percentage points. It also obtains the best $88.0\% \pm 0.1\%$ F1-Score and average precision, with an mAP@0.5 of $92.5\% \pm 0.1\%$ and mAP@0.5:0.95 of $69.6\% \pm 0.2\%$. The minimal variability observed demonstrates that our model's performance improvements are statistically meaningful and highly stable.

Among the baseline YOLO models, YOLOv3 showed limited accuracy in detecting fine defects. YOLOv8n improved with enhanced multi-scale fusion and bounding-box prediction, reaching $89.9\% \pm 0.3\%$ mAP@0.5, yet still lagged behind our method. Hyper-YOLO, with its advanced neck design, achieved $91.1\% \pm 0.2\%$ mAP@0.5 and $86.6\% \pm 0.3\%$ precision but was still inferior to MGC-YOLOv8 across all metrics.

Furthermore, addressing the critical concern of computational efficiency and runtime overhead in industrial inspection systems, Table IV introduces a comparative analysis of parameter counts (Params) and inference speeds (FPS). The baseline YOLOv8n achieves a high inference speed of approximately 210 FPS with 3.01M parameters. The integration of our proposed MSEDF, GLSA, and CBAM modules inevitably increases the parameter count to 3.95M, resulting in an inference speed of 165 FPS. However, this marginal runtime overhead is a highly worthwhile trade-off, as 165 FPS still far exceeds the standard real-time processing threshold required by industrial cameras. Compared to complex architectures like RT-DETR (120 FPS, 20.0M Params) and Hyper-YOLO (135 FPS, 6.5M Params), MGC-YOLOv8 offers a definitively superior speed-accuracy trade-off.

Finally, the training curves in Fig. 8 show that MGC-YOLOv8 converges more rapidly and smoothly, indicating that the integration of MSEDF, GLSA, and CBAM modules effectively improves feature extraction for subtle tool defects.

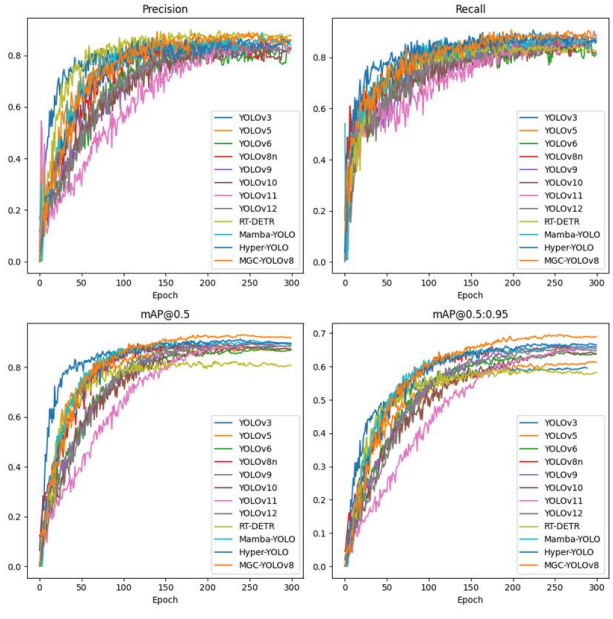


Fig. 8. Comparison of Precision, Recall, mAP@0.5 and mAP@0.5:0.95 curves trained with different algorithms.

TABLE II. CBAM ROUTING LOCATION EXPERIMENTAL RESULTS

Model	Precision	Recall	F1	mAP@0.5	mAP@0.5:0.95
CBAM (18,19)	84.50%	85.30%	84.90%	89.80%	67.60%
CBAM (21,22)	**88.10%**	**87.90%**	**88.00%**	**92.50%**	**69.60%**
CBAM (24,25)	84.20%	87.00%	85.58%	92.30%	68.10%
CBAM (18,19)(21,22)	87.70%	87.10%	87.40%	91.90%	68.70%
CBAM (18,19)(24,25)	86.70%	84.60%	85.64%	91.30%	68.20%
CBAM (21,22)(24,25)	84.60%	88.20%	86.36%	91.80%	67.50%
CBAM (18,19)(21,22)(24,25)	87.70%	82.30%	84.91%	90.60%	67.10%

TABLE III. ABLATION EXPERIMENT RESULTS

MSEDF	GLSA	CBAM	Precision	Recall	F1	mAP@0.5	mAP@0.5:0.95
			82.0%	86.9%	84.3%	89.9%	66.1%
✓			84.3%	87.2%	85.7%	90.8%	67.4%
	✓		82.9%	86.3%	84.6%	90.3%	66.7%
		✓	87.3%	85.5%	86.4%	90.2%	66.7%
✓	✓		86.8%	85.7%	86.2%	91.5%	68.7%
✓		✓	86.2%	86.5%	86.4%	91.2%	68.1%
	✓	✓	85.0%	87.1%	86.0%	90.8%	67.0%
✓	✓	✓	**88.1%**	**87.9%**	**88.0%**	**92.5%**	**69.6%**

TABLE IV. COMPARISON OF EXPERIMENTAL RESULTS OF DIFFERENT ALGORITHMS (AVERAGE ± STD OVER 3 RUNS)

Model	Params (M)	FPS	Precision (%)	Recall (%)	F1-Score (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv3 (tiny)	8.7	185	85.4 ± 0.6	82.8 ± 0.5	84.1 ± 0.5	88.3 ± 0.4	60.1 ± 0.5
YOLOv5n	**1.9**	**230**	84.7 ± 0.4	86.1 ± 0.4	85.4 ± 0.4	87.5 ± 0.3	61.4 ± 0.4
YOLOv6n	4.3	205	81.3 ± 0.5	82.7 ± 0.5	82.0 ± 0.5	87.5 ± 0.4	64.5 ± 0.4
YOLOv8n (Baseline)	3.2	210	82.0 ± 0.4	86.9 ± 0.3	84.4 ± 0.4	89.9 ± 0.3	66.1 ± 0.4
YOLOv9t	7.1	175	82.7 ± 0.4	84.7 ± 0.3	83.7 ± 0.3	89.5 ± 0.2	65.9 ± 0.3
YOLOv10n	2.3	225	83.0 ± 0.3	83.2 ± 0.4	83.1 ± 0.3	88.3 ± 0.3	64.9 ± 0.3
YOLOv11n	2.6	215	81.7 ± 0.4	85.2 ± 0.3	83.4 ± 0.3	89.1 ± 0.2	65.0 ± 0.4
YOLOv12n	2.8	200	83.1 ± 0.3	86.2 ± 0.3	84.6 ± 0.3	89.2 ± 0.2	65.5 ± 0.3
RT-DETR-R18	20.0	120	86.6 ± 0.3	84.0 ± 0.4	85.3 ± 0.3	81.7 ± 0.5	59.0 ± 0.5
Mamba-YOLO	5.0	145	85.8 ± 0.3	86.4 ± 0.3	86.1 ± 0.3	90.7 ± 0.2	66.8 ± 0.3
Hyper-YOLO	6.5	135	86.6 ± 0.3	85.1 ± 0.2	85.9 ± 0.2	91.1 ± 0.2	66.8 ± 0.2
MGC-YOLOv8 (Ours)	3.95	165	**88.1 ± 0.2**	**87.9 ± 0.2**	**88.0 ± 0.1**	**92.5 ± 0.1**	**69.6 ± 0.2**

D. Visual Analysis

In object detection, visualizing results provides an intuitive means to evaluate model performance and accuracy. It offers quick insights into false positives, false negatives, and a model's ability to distinguish similar objects, thereby guiding further refinement. This section presents comparative visual results of the original and the enhanced MGC-YOLOv8 model under multi-condition industrial tool inspection scenarios, highlighting improvements in suppressing background noise, reinforcing edge features, and increasing detection confidence.

As shown in Fig. 9, baseline YOLOv8 [Fig. 9(a1)] misclassifies a background item as a chipped blade, while MGC-YOLOv8 [Fig. 9(a2)] correctly recognizes the tool and shows higher confidence, demonstrating improved resistance to background interference. In Fig. 9(b1), the baseline misses a broken blade in a cluttered environment, but the enhanced model [Fig. 9(b2)] detects it accurately, eliminating the omission. Furthermore, where YOLOv8 generates duplicate and incorrect labels [Fig. 9(c1)], MGC-YOLOv8 [Fig. 9(c2)] corrects these errors and raises confidence by 0.26 for the correct defect, indicating that the structural refinements more effectively integrate global and local features and improve discrimination between visually similar classes.

E. Confusion Matrix Analysis

The confusion matrix is an essential tool for evaluating the classification performance and robustness of object detection models. As illustrated in the normalized confusion matrices (see Fig. 10), the proposed MGC-YOLOv8 demonstrates substantial improvements over the baseline YOLOv8 across all key categories. Specifically, the detection accuracy for the Chipped Blade class increased from 0.80 to 0.84, while the Broken Blade class maintained a consistently high accuracy of 0.97, indicating strong robustness in identifying severe structural damage. Furthermore, the accuracy for the Worn Blade class improved from 0.81 to 0.86, and the Normal class increased from 0.74 to 0.83. This demonstrates the enhanced model's superior capability in recognizing fine-grained tool surface conditions.

Regarding class confusion and boundary errors, the improved architecture effectively addresses the baseline's limitations. Before the improvement, the probability of a Chipped Blade being misclassified as a Broken Blade was 0.14. This error rate dropped to 0.09 in MGC-YOLOv8, proving that the Multi-Scale Edge-Dual Fusion (MSEDF) mechanism significantly strengthens the extraction of geometric edge features, effectively distinguishing local minor defects from overall

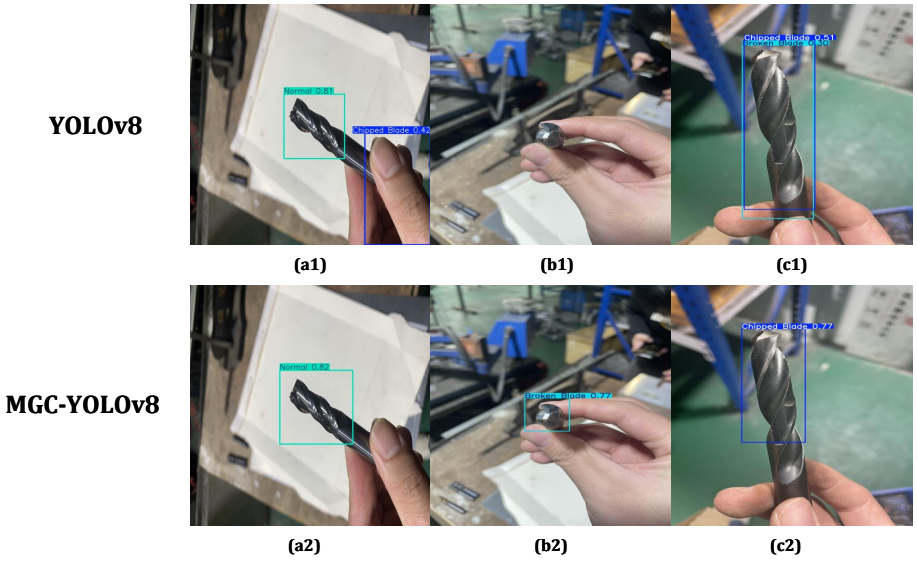
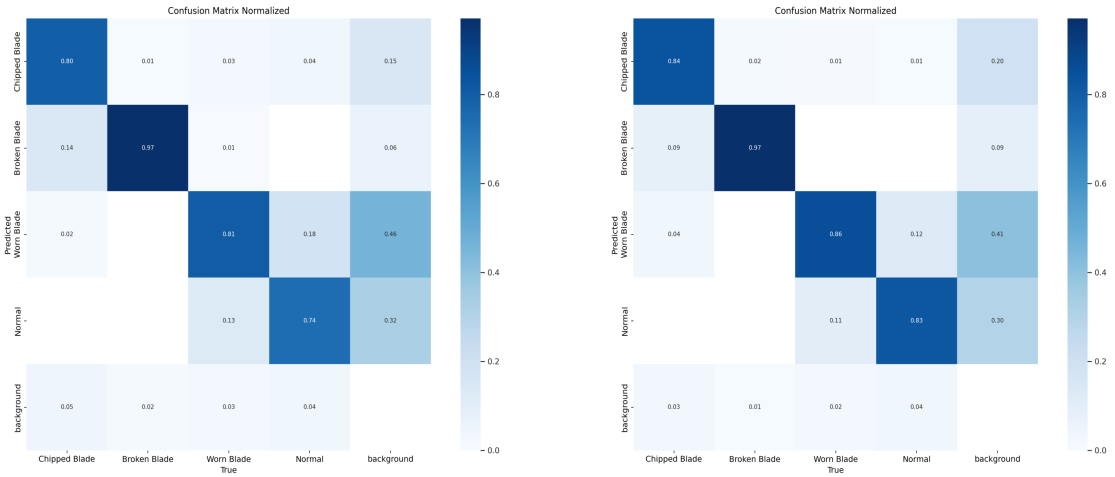


Fig. 9. Detection results before and after algorithm improvement.



(a) YOLOv8 (b) MGC-YOLOv8

Fig. 10. Confusion matrix of YOLOv8n and MGC-YOLOv8.

fractures. Similarly, the probability of a Normal surface being misjudged as a Worn Blade decreased from 0.18 to 0.12. This reduction highlights that the introduced GLSA module successfully correlates global and local semantics, capturing microscopic gradient information of wear textures to clarify ambiguous boundary errors between healthy and slightly worn surfaces.

In summary, the synergy of these proposed modules is intuitively reflected in the confusion matrix through higher diagonal accuracy and reduced inter-class confusion, validating the overall effectiveness of MGC-YOLOv8 in complex industrial tool anomaly detection.

VI. Conclusion

This study tackles the critical challenges of industrial tool anomaly detection by proposing the MGC-YOLOv8 model. The integration of a MSEDF module, a GLSA module, and a CBAM enables efficient extraction and fusion of micro-defect cues—such as cracks, wear, chipping, and breakage—on cutting-tool surfaces, yielding marked improvements in both detection accuracy and robustness. A high-quality dataset, collected under realistic factory conditions, underpins model training and validation, substantially enhancing the real-world applicability of our approach. Experimental evaluations confirm that MGC-YOLOv8 surpasses conventional detection techniques and state-of-the-art deep-learning models across

precision, recall, and mean average precision metrics.

Looking ahead, our future research will focus on several key areas to further enhance the system's robustness and applicability. To address harsh working conditions such as extremely low contrast, strong noise, or drastic changes in illumination, we plan to explore more robust feature enhancement and adaptive normalization strategies, ensuring the model can stably capture minute anomalies in diverse environments. Furthermore, although the current model demonstrates strong practical deployment potential in terms of theoretical parameter counts and inference speed, it has not yet been rigorously tested on physical embedded hardware. Therefore, a critical next step involves combining lightweight technologies—such as network pruning, knowledge distillation, and model quantization—to deploy the improved YOLOv8 on edge computing and embedded devices for genuine real-time online monitoring. Additionally, to mitigate the challenges of sample scarcity and category imbalance, we aim to introduce semi-supervised learning, cross-domain transfer learning, and diversified data augmentation methods, which will enhance the model's generalization ability across limited sample sizes and novel scenarios. Finally, integrating anomaly detection with predictive maintenance, multi-task learning, or cross-industry collaborative data training is anticipated to not only improve detection accuracy but also provide early warning support for tool life estimation and equipment health management. Through these endeavors, we aim to develop a lighter, more efficient, intelligent, and portable industrial tool monitoring system, thereby promoting the broader implementation of intelligent manufacturing across a wider range of practical applications.

ACKNOWLEDGMENT

This work was partially funded by the Henan Science and Technology Research Program under Grant 242102211009, 252102210173, the Key Research Projects of Higher Education Institutions in Henan Province under Grant 23A520022, 24A520049 and 25A520051, and the Henan Province Higher Education Institutions Young Backbone Teachers Training Program Grants 2021GGJS159.

REFERENCES

[1] F. Tao, Q. Qi, A. Liu, and A. Kusiak, "Data-driven smart manufacturing," *Journal of Manufacturing Systems*, vol. 48, pp. 157–169, 2018.

[2] J. A. Ghani, M. Rizal, M. Z. Nuawi, M. J. Ghazali, and C. H. C. Haron, "Monitoring online cutting tool wear using low-cost technique and user-friendly GUI," *Wear*, vol. 271, no. 9, pp. 2619–2624, 2011.

[3] G. Byrne, D. Dornfeld, and I. Inasaki, "Tool condition monitoring (TCM)—The status of research and industrial application," *CIRP Annals - Manufacturing Technology*, vol. 44, no. 2, pp. 541–567, 1995.

[4] X. Sun, L. Sun, and E. Wang, "Study on joint surface parameter identification method of shaft–toolholder and toolholder–tool for vertical CNC milling machine," *Machine Tool & Hydraulics*.

[5] M. P. Gómez, A. M. Hey, J. E. Ruzzante, and C. E. D'Attellis, "Tool wear evaluation in drilling by acoustic emission," *Physics Procedia*, vol. 3, no. 1, pp. 819–825, 2010.

[6] X. Shi, X. Wang, L. Jiao, Z. Wang, G. Pei, and S. Gao, "A real-time tool failure monitoring system based on cutting force analysis," *The International Journal of Advanced Manufacturing Technology*, vol. 95, pp. 5–8, 2018.

[7] M. Nouri, B. K. Fussell, B. L. Ziniti, and E. Linder, "Real-time tool wear monitoring in milling using a cutting condition independent method," *International Journal of Machine Tools and Manufacture*, vol. 89, pp. 1–13, 2015.

[8] X. Cao, B. Chen, B. Yao, and S. Zhuang, "An intelligent milling tool wear monitoring methodology based on convolutional neural network with derived wavelet frames coefficient," *Applied Sciences*, vol. 9, no. 18, p. 3912, 2019.

[9] Q. Hou, J. Sun, and P. Huang, "A novel algorithm for tool wear online inspection based on machine vision," *The International Journal of Advanced Manufacturing Technology*, 2019.

[10] Y. LeCun and L. Bottou, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[11] J. Terven, D. M. Cordova-Esparza, and J. A. Romero-Gonzalez, "A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS," *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, 2023.

[12] J. Wang, G. Xu, F. Yan, J. Wang, and Z. Wang, "Defect transformer: An efficient hybrid transformer architecture for surface defect detection," *Measurement*, 2023.

[13] F. Tang, Z. Xu, Q. Huang, J. Wang, X. Hou, J. Su, and J. Liu, "DuAT: Dual-aggregation transformer network for medical image segmentation," in *Proc. Chinese Conf. Pattern Recognition and Computer Vision (PRCV)*, pp. 343–356, 2023.

[14] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. ECCV*, 2018.

[15] J. Wang, W. Wang, Z. Zhang, X. Lin, J. Zhao, M. Chen, and L. Luo, "YOLO-DD: Improved YOLOv5 for defect detection," *Computers, Materials & Continua*, vol. 78, no. 1, 2024.

[16] J. Chen, Q. Wu, D. Liu, and T. Xu, "Foreground-background imbalance problem in deep object detectors: A review," *IEEE*, 2020.

[17] K. Zhu, Y. S. Wong, and G. S. Hong, "Wavelet analysis of sensor signals for tool condition monitoring: A review and some new results," *International Journal of Machine Tools and Manufacture*, vol. 49, no. 7–8, pp. 537–553, 2009.

[18] C. Zhou et al., "Vibration singularity analysis for milling tool condition monitoring," *International Journal of Mechanical Sciences*, vol. 166, 2020.

[19] Z. He, T. Shi, J. Xuan, and T. Li, "Research on tool wear prediction based on temperature signals and deep learning," *Wear*, vol. 478, p. 203902, 2021.

[20] Y. Zhou et al., "A new tool wear condition monitoring method based on deep learning under small samples," *Measurement*, vol. 189, p. 110622, 2022.

[21] P. Twardowski, M. Tabaszewski, M. Wiciak-Pikula, and A. Felusiak-Czyryca, "Identification of tool wear using acoustic emission signal and machine learning methods," *Precision Engineering*, 2021.

[22] R. Guo, H. Zhang, X. Zhi, and W. Wei, "In-process detection of tool wear based on machine vision," *Journal of Lanzhou University of Technology*, vol. 50, no. 6, pp. 33–41, 2024.

[23] J. Yu, X. Cheng, L. Lu, and B. Wu, "A machine vision method for measurement of machining tool wear," *Measurement*, vol. 182, p. 109683, 2021.

[24] Y. Tang et al., "Deep transfer hybrid model–based tool wear condition monitoring method," *Machine Tools & Hydraulics*, vol. 52, no. 20, pp. 63–71, 2024.

[25] J. Redmon et al., "You only look once: Unified, real-time object detection," in *Proc. CVPR*, 2016.

[26] T. Y. Lin et al., "Feature pyramid networks for object detection," in *Proc. CVPR*, 2017.

[27] S. Liu et al., "Path aggregation network for instance segmentation," in *Proc. CVPR*, 2018.

[28] G. Cheng et al., "Towards large-scale small object detection: Survey and benchmarks," *IEEE TPAMI*, vol. 45, no. 11, pp. 13467–13488, 2023.

[29] Y. Cui et al., "Focal network for image restoration," in *Proc. ICCV*, pp. 13001–13011, 2023.

[30] M. Yu et al., "Dense detection algorithm for ceramic tile defects based on improved YOLOv8," *Journal of Intelligent Manufacturing*, 2024.

[31] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. CVPR*, pp. 7263–7271, 2017.

[32] C. Li et al., "YOLOv6: A single-stage object detection framework for industrial applications," *arXiv preprint arXiv:2209.02976*, 2022.

[33] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning what you want to learn using programmable gradient information," in *Proc. ECCV*, 2024.

[34] A. Wang et al., "YOLOv10: Real-time end-to-end object detection," *arXiv preprint arXiv:2405.14458*, 2024.

[35] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-centric real-time object detectors," *arXiv preprint arXiv:2502.12524*, 2025.

[36] N. Carion et al., "End-to-end object detection with transformers," in *Proc. ECCV*, pp. 213–229, 2020.

[37] Z. Wang et al., "Mamba-YOLO: A simple baseline for object detection with state space model," in *Proc. AAAI*, vol. 39, no. 8, pp. 8205–8213, 2025.

[38] Y. Feng et al., "Hyper-YOLO: When visual object detection meets hypergraph computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Tomato Maturity Analysis: A Comparative Study of Detection and Instance Segmentation Using YOLOv8

Salma Ait Oussous, Rachid El Bouayadi, Driss Zejli, Aouatif Amine
Advanced Systems Engineering Laboratory (ISA)
National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco

***Abstract*—The accurate visual analysis of fruit maturity in complex agricultural scenes remains a fundamental challenge due to gradual appearance changes, object overlap, and partial occlusion. This study addresses tomato maturity analysis, formally defined as instance-level binary classification and spatial localization under varying degrees of visual density. While bounding-box-based object detection is widely used, it often lacks precision in dense clusters. We present a controlled experimental comparison between object detection and instance segmentation using a common YOLOv8-medium (YOLOv8m) backbone to isolate the effect of spatial representation. Experimental results demonstrate that instance segmentation achieves superior localization accuracy and boundary consistency, reaching a mask-based mAP@0.5:0.95 of 0.817. These findings suggest that pixel-level supervision effectively reduces localization ambiguity, providing a robust foundation for automated agricultural monitoring.**

***Keywords*—Tomato maturity detection; computer vision; object detection; instance segmentation; image analysis; Deep Learning**

I. Introduction

Understanding and interpreting visual information in complex scenes remains a fundamental challenge in computer vision and image analysis [1]. Many real-world visual environments are characterized by gradual appearance variations, dense object arrangements, and frequent occlusions, which complicate reliable object interpretation. These challenges are particularly pronounced in tasks that require fine-grained analysis of object properties rather than simple object presence or absence [2].

Fruit maturity analysis represents a representative visual computing problem within this context. In agricultural images, traditional image processing approaches for fruit analysis have relied on manual features and threshold-based decision rules derived from color or texture descriptors [3]. While effective in controlled settings, such approaches are sensitive to illumination changes and background complexity and often fail to generalize to visually diverse scenes [4]. Unlike problems involving distinct object categories, maturity assessment requires distinguishing between visually similar instances that differ only by small, continuous appearance variations. This makes maturity detection a demanding image analysis task, especially in unconstrained imaging conditions [5].

Formally, we define the tomato maturity analysis task as a fine-grained visual computing problem consisting of instance-level binary classification (ripe vs. unripe) combined with spatial localization under varying degrees of visual complexity. The task is constrained by assumptions of greenhouse-like environments where objects exhibit significant overlap and partial occlusion. The primary deployment scenario is the integration into automated agricultural monitoring or robotic harvesting systems, where precision in both maturity stage identification and spatial delineation is critical for operational success. In this study, our key contribution is a controlled experimental comparison between object detection and instance segmentation using the same YOLOv8-medium (YOLOv8m) backbone, dataset, and training protocol. This design enables us to isolate the effect of spatial representation (bounding-box vs. pixel-level) on visual analysis performance.

The introduction of Deep Learning (DL), particularly Convolutional Neural Networks (CNN), has significantly improved robustness and accuracy in visual recognition tasks, specifically in fruit detection and maturity classification performance [6]. Numerous studies have applied object detection frameworks to identify fruits and estimate their maturity stages by assigning class labels to detected instances [7] [8]. Single-stage detectors, including YOLO-based architectures, have been widely adopted due to their computational efficiency and strong performance in real-time visual analysis tasks [9]. In agricultural image analysis, YOLO-based models have been successfully used for tomato detection and ripeness estimation, demonstrating their effectiveness in localizing fruits under greenhouse and field conditions [10]. For example, YOLOv3 was used for tomato classification in greenhouse environments with promising accuracy, although model size posed challenges for deployment [11]. YOLOv5-based approaches have been employed to detect overlapping tomatoes and small fruits, often incorporating attention mechanisms to improve localization performance [12] [13]. Improvements to YOLOv7 have also been proposed to address challenges related to small object detection and fruit overlap in dense scenes [14]. Despite their success, object detection methods typically rely on bounding-box representations, which provide only an approximate description of object extent. In visually dense scenes, bounding boxes may include background regions or overlap with neighboring objects, leading to ambiguity in object boundaries [7]. This limitation becomes particularly evident in agricultural imagery, where fruits often grow in clusters and partially occlude one another, reducing the reliability of fine-grained visual interpretation.

Instance segmentation has emerged as an alternative object representation that assigns pixel-level labels to individual object instances [15]. By explicitly modeling object boundaries and surface extent, instance segmentation enables clearer separation of adjacent objects and more accurate delineation of object shape [16]. In visual computing research, segmentation-based approaches such as Mask R-CNN and DeepLab have demonstrated improved performance in scenarios involving

occlusion and complex object interactions, for example Mask R-CNN has been applied to tomato segmentation, demonstrating improved boundary localization compared to bounding-box-based detectors, particularly in congested scenes [17]. Semantic segmentation architectures such as DeepLabv3+ have also shown strong performance in handling irregular object shapes and occlusions in fruit recognition tasks [18]. Hybrid approaches combining region proposal mechanisms with segmentation networks have further improved separation of closely clustered fruits [19]. In addition, segmentation-based models have been successfully integrated into robotic harvesting systems, where precise spatial perception is critical for manipulation and interaction [20].

However, most existing studies focus on improving the performance of a specific detection or segmentation model rather than systematically examining the influence of object representation choice [21]. Comparisons between detection-based and segmentation-based approaches are often conducted under different experimental settings, datasets, or network configurations, making it difficult to isolate the effect of representation on visual analysis performance. As a result, the relative impact of bounding-box-based versus pixel-level representations for fruit maturity analysis in dense and occluded scenes remains insufficiently explored.

To address this gap, the present study conducts a systematic comparison of object detection and instance segmentation approaches within a common DL model. By training and evaluating both representations under identical conditions, the analysis focuses on how object representation affects localization accuracy, boundary definition, and robustness in visually complex scenes. Through quantitative evaluation and qualitative analysis, this study provides insight into the role of bounding-box-based and pixel-level representations in fine-grained image analysis and contributes to a clearer understanding of representation choices for maturity detection tasks in visual computing.

The remainder of this study is organized as follows: Section II describes the database and methodology, Section III presents the experimental results and discussion, Section IV concludes the study and outlines directions for future research.

II. Methodology

This section describes the methodology adopted to evaluate object detection and instance segmentation approaches for tomato maturity analysis. The objective is to conduct a controlled comparison of two visual representation strategies bounding-box-based detection and pixel-level instance segmentation under identical experimental conditions. The methodology includes the database description, annotation process, model configuration, training protocol, and evaluation metrics.

A. Database Description and Annotation

The experiments were conducted using a tomato image database obtained from Kaggle, consisting of images captured under greenhouse-like conditions. The database includes 177 images representing two tomato maturity stages: ripe and unripe. The images exhibit a variety of visual conditions, including changes in illumination, background complexity, fruit density, and varying levels of occlusion. In total, the dataset contains 209 annotated tomato instances (124 ripe and 85 unripe) across the validation set, with an average of approximately 5.8 instances per image. This density provides a representative basis for evaluating localization performance in both sparse and clustered scenarios. The dataset was split into training, validation, and testing subsets using an 80:10:10 ratio, ensuring that diverse imaging conditions were represented across all partitions.

Although the original images were sourced from Kaggle, all annotations used in this study were created manually by the authors to support object detection and instance segmentation tasks. Pixel-level annotations were generated using the VGG Image Annotator (VIA) tool. Each tomato instance was carefully labeled to ensure accurate spatial localization and shape representation, which is essential for reliable evaluation of both bounding-box-based detectors and segmentation models.

Pixel-level annotations were created to precisely delineate individual tomato instances. From these annotations, bounding boxes were derived to enable object detection experiments, while the original pixel-level masks were used for instance segmentation analysis. Due to variations in fruit size, maturity stage, viewpoint, and frequent occlusions, the annotation process required substantial manual effort to maintain consistency and accuracy across the database.

The annotated database provides reliable ground truth labels for supervised learning and quantitative performance evaluation. The database was split into training, validation, and testing subsets to ensure unbiased model assessment. Representative examples of original images and their corresponding manual annotations, which reflect the actual labels used during model training and evaluation, are shown in Fig. 1.

B. Object Representation Models and Training Configuration

To enable a fair comparison between object detection and instance segmentation for tomato maturity analysis, both approaches were implemented within a DL model based on the YOLOv8m architecture. Using a unified framework ensures that differences in performance can be attributed primarily to the choice of object representation rather than to variations in backbone design, feature extraction, or optimization strategy.

In the object detection configuration, the network predicts a set of bounding boxes associated with class labels corresponding to ripe and unripe tomato instances. Each prediction represents an object hypothesis defined by its bounding box coordinates, confidence score, and class probability. This representation provides a compact description of object location but does not explicitly model object boundaries or surface extent.

In the instance segmentation configuration, the detection formulation is extended by incorporating a segmentation head that generates a pixel-level mask for each detected instance. In this case, the network outputs both bounding boxes and corresponding instance masks, enabling explicit delineation of object boundaries and improved separation of adjacent objects. The masks are derived from high-resolution feature maps to preserve fine object details.

Both configurations share the same backbone network, feature extraction layers, and multi-scale feature aggregation com-

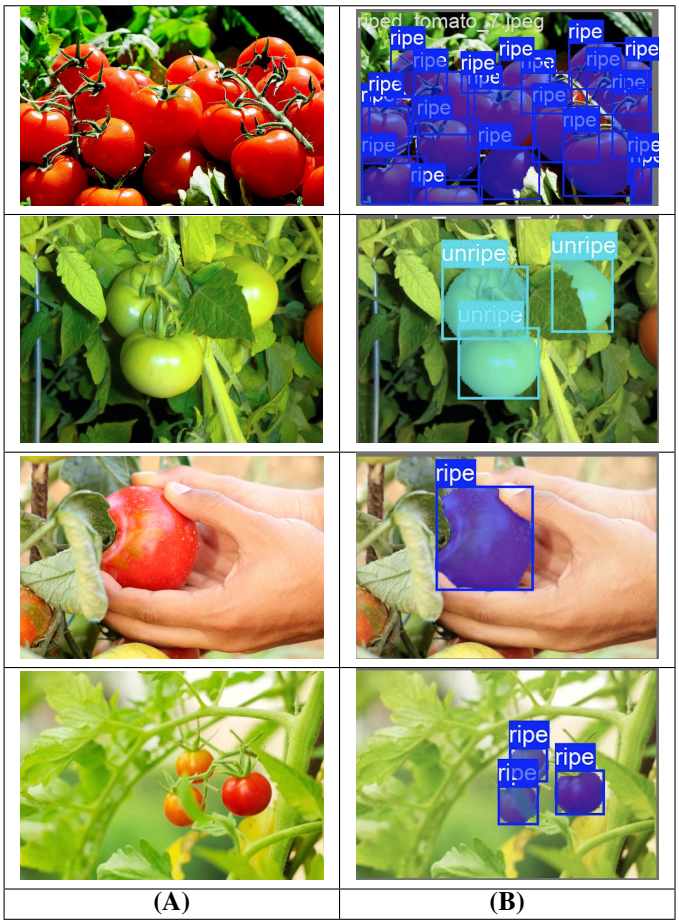


Fig. 1. Representative examples of original images and their corresponding manual annotations from the tomato maturity database. The left column (A) shows original images, while the right column (B) presents annotated images with bounding boxes, indicating individual tomato instances classified as ripe or unripe.

ponents. The only difference between the two approaches lies in the output representation: bounding-box-based localization for detection and pixel-level instance masks for segmentation.

Moreover, both models were trained using identical training configurations, This includes the same input image resolution, batch size, optimizer, learning rate schedule, and number of training epochs. By maintaining identical optimization settings, the experimental setup ensures that observed performance differences are attributable to the object representation strategy rather than to training-related variations. The database was split into training and validation subsets using the same partitioning strategy for both configurations. Model selection was based on validation performance using the evaluation metrics described in Section II-D.

C. Experimental Environment

All experiments were conducted using the Google Colab platform on a single NVIDIA Tesla T4 GPU. While the study relies on a single hardware configuration, the NVIDIA T4 provides a common benchmark for evaluating the computational efficiency of YOLO-based models. To assess the practical trade-offs for real-world agricultural deployment, we measured processing speed across preprocess, inference, and postprocess stages for both object detection and instance segmentation configurations.

For the object detection model (YOLOv8m-Det), the average processing speed per image was 0.2ms for preprocessing, 27.1ms for inference, and 1.7ms for postprocessing, totaling approximately 29.0ms per image. The instance segmentation model (YOLOv8m-Seg) exhibited a slight increase in computational overhead, with 2.8ms for preprocessing, 30.1ms for inference, and 5.0ms for postprocessing, totaling 37.9ms per image. This represents an approximate inference-time overhead of 30.7% compared to detection. Despite this increase, the segmentation model maintains a high frame rate suitable for real-time applications on edge devices.

All implementations were carried out in Python 3 using the Ultralytics YOLOv8 library. The YOLOv8m configuration was selected to provide a consistent and balanced computational setting for both object detection and instance segmentation tasks. Using the same model scale across experiments ensures that performance differences reflect representation choice rather than differences in model capacity or computational resources.

D. Evaluation Metrics

Model performance was evaluated using standard metrics commonly employed in object detection and instance segmentation tasks. These include Mean Average Precision (mAP) at threshold 50 (mAP@50), mAP from 50 to 95 (mAP@50:95), recall (R), and box precision [Box(P)].

The Mean Average Precision assesses the mean precision for all categories at a fixed Intersection over Union (IoU) of 0.5 (mAP50), and across a range from 0.5 to 0.95 (mAP50:95). It expresses the ability of the model to localize and label objects accurately. To ensure a fair comparison, we report mAP@0.5:0.95 computed using bounding-box IoU for the detection model and both bounding-box and mask-based IoU for the segmentation model. This dual evaluation isolates the benefit of pixel-level supervision (mask IoU) while maintaining a common denominator (box IoU) for benchmarking against traditional detectors. The mAP equation is given by Eq. (1), where the average precision for class i is denoted as AP_i and the number of classes as N [22].

$$mAP = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i \tag{1}$$

The Recall metric signify the capability of the model to find all the relevant objects. It is the number of objects detected accurately about the total ground truth instances. A high figure of recall indicates low numbers of missed detections, which is an important requirement for agricultural uses such as tracking the maturity of fruits. It is determined by Eq. (2), where *TP* is an outcome where the model accurately forecasts the positive class. An *FN* is a result where the model misclassifies the negative class [23].

$$\text{R} = \frac{TP}{TP + FN} \tag{2}$$

While, Box(P) refers to the ratio of the number of accurately predicted bounding boxes to the total number of predicted boxes. It determines the rate at which the model's detections are precise and not false positives. High precision implies that the model provides only the most confident and relevant detections. It is represented by Eq. (3), where the *FP* is an output where the model misclassifies the positive class [24].

$$\text{Box(P)} = \frac{TP}{TP + FP} \tag{3}$$

In addition to quantitative evaluation, qualitative analysis was conducted through visual inspection of detection outputs and segmentation masks. This analysis focuses on boundary definition, object separation, and behavior in scenes containing overlapping or partially occluded fruit instances.

III. Results and Discussion

This section presents and discusses the experimental results obtained from the comparative evaluation of object detection and instance segmentation approaches for tomato maturity analysis. The objective is to analyze how different object representations influence localization accuracy, boundary delineation, and confidence behavior in visually complex scenes. Quantitative metrics, Precision–Recall characteristics, and qualitative visual comparisons are jointly examined to provide a comprehensive assessment.

Before evaluating instance segmentation, object detection performance was examined in our previous work to establish a suitable baseline [25]. Multiple YOLOv8 detection variants, including nano, small, medium, large, and extra-large configurations, were evaluated under identical experimental conditions. Among these models, the YOLOv8 medium configuration achieved the best overall balance between detection accuracy and model capacity, with an mAP@0.5 of approximately 0.84. Increasing model size beyond this configuration resulted in only marginal performance gains, while smaller models exhibited reduced accuracy. These findings suggest that detection performance is not primarily limited by model capacity, motivating the present investigation into alternative object representations rather than further architectural scaling. We acknowledge that re-evaluating these variants under the exact segmentation comparison setup would further reduce selection bias; however, the medium variant's stable performance justifies its choice for isolating representation effects.

The quantitative performance comparison between the detection and segmentation approaches is summarized in Table I. The results reported represent outcomes from controlled training cycles. While these metrics are derived from a single stabilized training run, the consistency of performance gains across all classes and the stability of the validation loss curves suggest that these values reflect robust performance trends rather than stochastic variations of the training process. This level of descriptive consistency provides a reliable basis for our comparative analysis of spatial representations.

Class-level analysis further emphasizes the benefits of pixel-level representation. For ripe tomatoes, the segmentation model achieves an mAP@0.5:0.95 of 0.880 compared to 0.511 for detection, highlighting improved boundary alignment and instance separation in dense scenes. Similar trends are observed for unripe tomatoes, particularly in recall, where segmentation demonstrates a higher ability to detect relevant instances and reduce missed detections. Although the detection-based approach exhibits relatively high bounding box precision for unripe tomatoes, this does not translate into robust localization performance under stricter IoU thresholds.

TABLE I. Comparative Model Performance for Tomato Maturity Detection and Segmentation.

Model	Class	Box (P)	Recall (R)	Box mAP		Mask mAP	
				@0.5	@0.5:0.95	@0.5	@0.5:0.95
v8m-Seg	Unripe	0.873	0.886	0.912	0.784	0.901	0.761
v8m-Seg	Ripe	0.939	0.898	0.951	0.880	0.943	0.872
v8m-Seg	**All**	**0.906**	**0.892**	**0.931**	**0.832**	**0.922**	**0.817**
v8m-Det	Unripe	0.964	0.805	0.892	0.591	-	-
v8m-Det	Ripe	0.767	0.742	0.785	0.511	-	-
v8m-Det	All	0.865	0.773	0.839	0.551	-	-

A more detailed investigation into performance across different IoU thresholds reveals that the advantage of instance segmentation becomes increasingly pronounced as the overlap criteria become more stringent. For thresholds between 0.75 and 0.95, the recall for the detection model degrades rapidly, while the segmentation approach maintains significantly higher levels of detection stability. This behavior can be attributed to the inherent nature of bounding-box representations, which often capture background pixels or parts of neighboring fruits, thereby reducing the precise overlap with the ground truth. In contrast, pixel-level masks provide a finer delineation of the fruit boundaries, effectively reducing spatial ambiguity in crowded clusters and improving handling of overlapping fruit.

Furthermore, we observed that the localized geometric accuracy provided by segmentation translates into more reliable classification in cluttered regions. While object detection sometimes struggles with small or heavily occluded tomatoes—often misclassifying them or producing false negatives—the segmentation representation's ability to isolate object surface area leads to higher classification confidence.

An analysis across fruit scales reveals that the segmentation advantage is particularly pronounced for small and medium-sized tomatoes. In such cases, bounding boxes often capture a disproportionate amount of background information relative to the object surface, leading to lower IoU values. The segmentation masks, by contrast, maintain high geometric fidelity regardless of object scale. Similarly, while both models perform well in low-occlusion scenarios, the segmentation model shows a consistent performance advantage as occlusion increases beyond 30%. This suggests that the benefit of pixel-level representation is primarily driven by its ability to resolve visual ambiguities in heavily overlapping clusters, which are common in dense greenhouse arrangements.

Scenarios where detection might still offer an advantage are typically limited to very simple, low-density scenes where the coarse localization of a bounding box is sufficient and the lower computational latency (29.0ms vs. 37.9ms per image) might be favored for high-throughput sorting. However, for complex greenhouse conditions, the improved boundary delineations and reduced ambiguity of instance segmentation

provide a clear superiority for maturity analysis.

The Precision–Recall characteristics of the instance segmentation approach are illustrated in Fig. 2. The curves indicate that high precision is maintained over a wide range of recall values for both maturity classes, with class-specific average precision values of approximately 0.910 for ripe tomatoes and 0.943 for unripe tomatoes, and an overall mAP@0.5 close to 0.93. The smooth decline in precision at high recall levels suggests that errors predominantly occur when attempting to detect nearly all instances in highly cluttered scenes. Compared to detection-based models, the segmentation approach exhibits greater stability across confidence thresholds, indicating reduced sensitivity to parameter tuning.

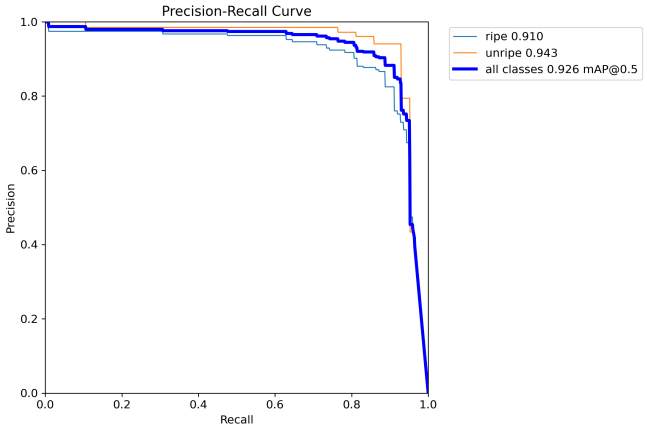


Fig. 2. Precision–Recall curve of YOLOv8m segmentation.

Compared to existing works, our results align with findings in [17], where Mask R-CNN demonstrated superior boundary localization for mature green tomatoes are compared to traditional detectors. However, our YOLOv8m-Seg approach offers a more efficient alternative with inference times around 37.9ms, significantly faster than typical Mask R-CNN implementations, while maintaining comparable spatial accuracy. Similarly, while SE-YOLOv3 [11] focuses on classification accuracy, our study highlights the geometric benefit of segmentation masks in clarifying object extent in dense greenhouse arrangements. The observed mAP@0.5:0.95 of 0.832 for box-based metrics in segmentation matches or exceeds several recent multi-modal fusion benchmarks [10], emphasizing that spatial representation choice is as critical as sensor integration.

A systematic failure mode analysis was conducted to understand the limitations of the proposed approach. We observed that segmentation boundaries occasionally exhibit "noise" or over-segmentation when tomatoes are situated against highly textured backgrounds or in extreme low-light conditions. In cases of severe occlusion (over 70% object surface hidden), both detection and segmentation models experience a reduction in recall; however, the segmentation model is more prone to producing fragmented masks. These boundaries may deviate from the true fruit contour when leaf shadows cast sharp edgeline gradients that the mask head misinterprets as object boundaries. Furthermore, although the results are based on a single training run, the stability of the validation loss and the consistency of mAP gains across classes suggest that these failure modes are systematic rather than random variabilities of the training process. Future work utilizing ensemble methods or multiple-run averaging could provide further confidence intervals for these performance trends.

Fig. 3 and Fig. 4 present a representative qualitative results comparing the YOLOv8m instance segmentation model and the YOLOv8m detection model. Each image pair illustrates the same input scene processed by both approaches.

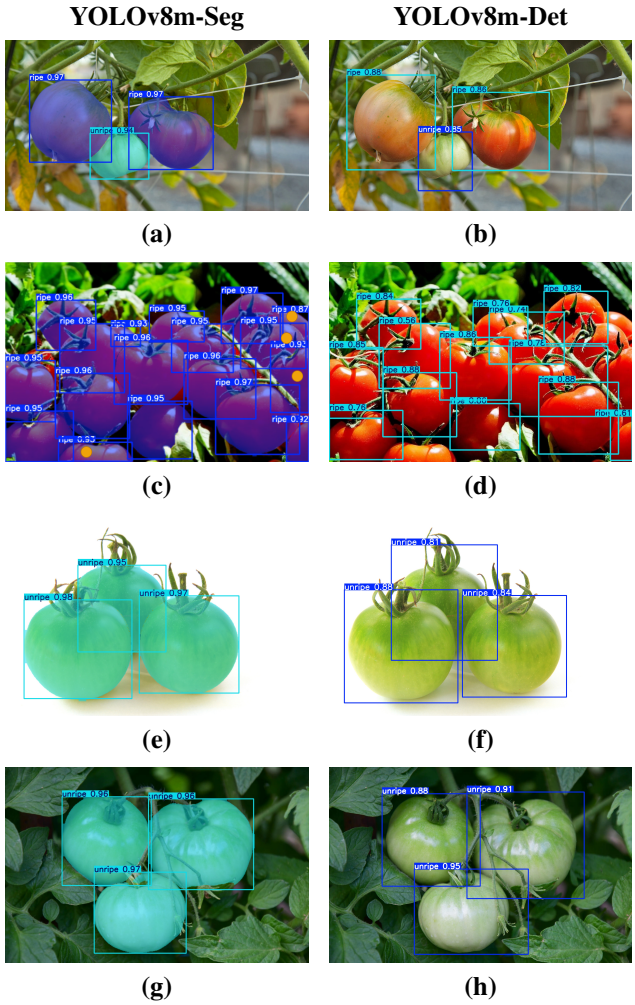


Fig. 3. Qualitative comparison between YOLOv8m instance segmentation (left) and YOLOv8m detection (right) for tomato ripeness stages (A).

In Fig. 3(a), the YOLOv8m-Seg model correctly detects two ripe tomatoes and one unripe tomato in a simple scene. It assigns high confidence values between 0.94 and 0.97 while producing pixel-level masks that closely follow object boundaries. The corresponding detection output in Fig. 3(b) identifies the same instances but assigns lower confidence values, typically between 0.85 and 0.88, and relies on bounding boxes that include surrounding background regions.

A more complex scenario is shown in Fig. 3(c), where a dense cluster of ripe tomatoes is present. The segmentation model successfully detects and segments nearly all visible instances, including partially occluded fruits, with confidence values mostly exceeding 0.95. In contrast, Fig. 3(d) shows the

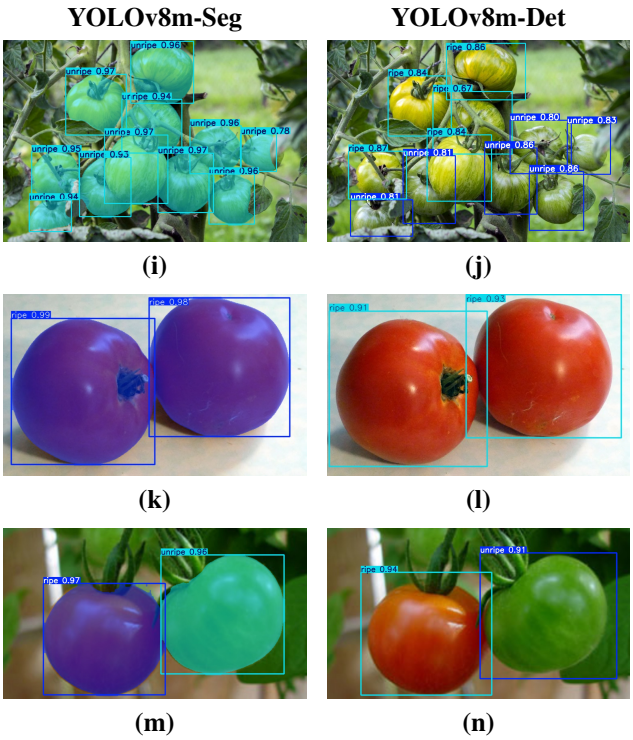


Fig. 4. Qualitative comparison between YOLOv8m instance segmentation (left) and YOLOv8m detection (right) for tomato ripeness stages (B).

detection-based output for the same scene, where several tomatoes are missed and confidence values vary widely, ranging from approximately 0.56 to 0.88, particularly in overlapping regions.

Another example dominated by visually similar unripe tomatoes are illustrated in Fig. 3(e) and Fig. 3(f). In Fig. 3(e), the segmentation model accurately delineates all instances and assigns confidence values between 0.95 and 0.98, maintaining consistent boundary definition despite low inter-class visual contrast. In Fig. 3(f), the detection-based model identifies most tomatoes but exhibits greater variability in confidence, typically between 0.81 and 0.88, and produces bounding boxes with limited separation between adjacent objects.

Fig. 3(g) and Fig. 3(h) present a moderately dense scene processed by both approaches. In Fig. 3(g), the segmentation model detects three unripe tomatoes with confidence values around 0.96–0.97 and clearly separates each instance using pixel-level masks. The detection output in Fig. 3(h) identifies the same instances but assigns lower confidence values, ranging from 0.88 to 0.95, and shows partial overlap between bounding boxes.

From Fig. 4, Fig. 4(i) and Fig. 4(j) present a visual representation dominated by unripe tomatoes with significant overlap and partial occlusions. In Fig. 4(i), the YOLOv8m-Seg model successfully identifies and segments nearly all visible unripe tomato instances. Confidence values are consistently high, typically ranging from 0.94 to 0.97, even for partially occluded fruits. The pixel-level masks enable clear separation of adjacent tomatoes despite the dense arrangement. In contrast, Fig. 4(j) shows the output of the YOLOv8m detection model for the same input image. While several tomatoes are detected, confidence values are lower and more variable, commonly between 0.80 and 0.87, and multiple bounding boxes overlap. Some instances are misclassified or weakly localized, reflecting increased ambiguity in crowded regions.

Fig. 4(k) and Fig. 4(l) illustrate a simpler visual configuration containing two ripe tomatoes placed against a uniform background. In Image (k), the segmentation model accurately delineates both fruits and assigns very high confidence values of approximately 0.98–0.99, indicating strong classification certainty and precise object extent modeling. In Fig. 4(l), the detection-based model also correctly identifies both ripe tomatoes; however, confidence values are slightly lower, around 0.91–0.93, and the bounding boxes include background regions, providing a coarser spatial description compared to the segmentation output.

Fig. 4(m) and Fig. 4(n) depict a mixed-maturity configuration with one ripe and one unripe tomato in close proximity. In Fig. 4(m), the YOLOv8m-Seg model correctly distinguishes between maturity stages, assigning a confidence of approximately 0.97 to the ripe tomato and 0.96 to the unripe tomato. The segmentation masks clearly separate the two instances, despite their spatial closeness.

In Fig. 4(n), the detection-based model also identifies both tomatoes; however, confidence values differ more noticeably, with the ripe tomato detected at around 0.94 and the unripe tomato at approximately 0.91. The bounding boxes partially overlap and provide less precise delineation between the two fruits.

Across all qualitative examples, the instance segmentation approach consistently demonstrates clearer object delineation and more stable confidence assignment as scene complexity increases. These qualitative observations complement the quantitative improvements reported in Table I and the Precision–Recall behavior shown in Fig. 2. Together, they reinforce the conclusion that object representation plays a critical role in tomato maturity analysis. Bounding-box-based detection provides efficient coarse localization but offers limited descriptive power in dense and ambiguous scenes. Instance segmentation addresses these limitations by explicitly modeling object boundaries at the pixel level, leading to improved localization accuracy, higher recall, and more reliable confidence behavior without increasing model capacity or architectural complexity.

IV. Conclusion

This study presented a systematic visual computing evaluation of object detection and instance segmentation approaches for tomato maturity analysis, with a particular focus on how object representation influences localization accuracy and boundary delineation in visually complex scenes. By conducting a controlled comparison within a common DL model and under identical experimental conditions, the analysis isolated the effect of bounding-box-based and pixel-level representations on performance. Our findings confirm that pixel-level representation provides a more reliable basis for fine-grained maturity analysis than coarse bounding-box localization, particularly as IoU thresholds increase. The observed geometric superiority of instance segmentation translates into

more consistent object separation and more stable confidence assignments in dense greenhouse scenes.

Despite these encouraging results, several limitations should be acknowledged. The evaluation was conducted on a localized dataset of 177 images and focused on two maturity classes (ripe and unripe) using a single YOLOv8m backbone. This scope may limit the immediate generalization of these absolute performance metrics to more diverse environmental conditions or complex multi-class maturity stages. Furthermore, the reliance on a single hardware configuration (NVIDIA T4) and a single training run per configuration provides a baseline but does not capture the full variance of hardware stability or training stochasticity.

Future research will aim to extend this analysis to larger and more diverse datasets, incorporating a broader range of tomato varieties and maturity stages to test generalizability across crop types such as peppers or apples. We propose to investigate deeper representation analyses, including panoptic modeling to handle background-foreground ambiguity and the integration of boundary-aware loss functions to further refine mask precision. Validating these conclusions on synthetic occlusion setups or multi-view datasets would also provide deeper insight into the robustness of pixel-level supervision. Finally, systemic failure mode analysis on an automated scale will be essential for refining these models for seamless deployment in autonomous robotic harvesting systems.

ACKNOWLEDGMENT

The authors would like to thank the Moroccan Ministry of Higher Education, Scientific Research and Innovation and the OCP Foundation who funded this work through the APRD research program.

REFERENCES

[1] M. Nixon and A. S. Aguado, *Feature Extraction and Image Processing for Computer Vision*. Oxford, U.K.: Academic Press, 2025.

[2] C. Che, H. Zheng, Z. Huang, W. Jiang, and B. Liu, "Intelligent Robotic Control System Based on Computer Vision Technology," *arXiv preprint*, arXiv:2404.01116, 2024.

[3] H. Zheng, L. Fu, and Q. Ye, "Flexible capped principal component analysis with applications in image recognition," *Information Sciences*, vol. 614, pp. 289–310, 2022.

[4] Y. Yang, Y. Han, S. Li, Y. Yang, M. Zhang, and H. Li, "Vision-based fruit recognition and positioning technology for harvesting robots," *Computers and Electronics in Agriculture*, vol. 213, p. 108258, 2023.

[5] G. C. Wakchaure, S. B. Nikam, K. R. Barge, S. Kumar, K. K. Meena, V. J. Nagalkar, J. D. Choudhari, V. P. Kad, and K. S. Reddy, "Maturity stages detection prototype device for classifying custard apple (Annona squamosa L.) fruit using image processing approach," *Smart Agricultural Technology*, vol. 7, p. 100394, 2024.

[6] H.-T. Vo, N. N. Thien, and K. C. Mui, "A deep transfer learning approach for accurate dragon fruit ripeness classification and visual explanation using Grad-CAM," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 11, 2023.

[7] P. Mittal, "A comprehensive survey of deep learning-based lightweight object detection models for edge devices," *Artificial Intelligence Review*, vol. 57, no. 9, p. 242, 2024.

[8] J. Ma, M. Li, W. Fan, and J. Liu, "State-of-the-art techniques for fruit maturity detection," *Agronomy*, vol. 14, no. 12, 2024.

[9] M. Amiriebrahimabadi, Z. Rouhi, and N. Mansouri, "A comprehensive survey of multi-level thresholding segmentation methods for image processing," *Archives of Computational Methods in Engineering*, vol. 31, no. 6, pp. 3647–3697, 2024.

[10] Y. Liu, C. Wei, S.-C. Yoon, X. Ni, W. Wang, Y. Liu, D. Wang, X. Wang, and X. Guo, "Development of multimodal fusion technology for tomato maturity assessment," *Sensors*, vol. 24, no. 8, p. 2467, 2024.

[11] F. Su, Y. Zhao, G. Wang, P. Liu, Y. Yan, and L. Zu, "Tomato maturity classification based on SE-YOLOv3-MobileNetV1 network under nature greenhouse environment," *Agronomy*, vol. 12, no. 7, p. 1638, 2022.

[12] S. N. Appe, G. Arulselvi, and G. N. Balaji, "CAM-YOLO: Tomato detection and classification based on improved YOLOv5 using combining attention mechanism," *PeerJ Computer Science*, vol. 9, p. e1463, 2023.

[13] H. Li, Z. Gu, D. He, X. Wang, J. Huang, Y. Mo, P. Li, Z. Huang, and F. Wu, "A lightweight improved YOLOv5s model and its deployment for detecting pitaya fruits in daytime and nighttime light-supplement environments," *Computers and Electronics in Agriculture*, vol. 220, p. 108914, 2024.

[14] Y. Bai, J. Yu, S. Yang, and J. Ning, "An improved YOLO algorithm for detecting flowers and fruits on strawberry seedlings," *Biosystems Engineering*, vol. 237, pp. 1–12, 2024.

[15] C. Charisis and D. Argyropoulos, "Deep learning-based instance segmentation architectures in agriculture: A review of the scopes and challenges," *Smart Agricultural Technology*, vol. 8, p. 100448, 2024.

[16] X. Gao, J. Ding, R. Zhang, and X. Xi, "YOLOv8n-CA: Improved YOLOv8n model for tomato fruit recognition at different stages of ripeness," *Agronomy*, vol. 15, no. 1, 2025.

[17] L. Zu, Y. Zhao, J. Liu, F. Su, Y. Zhang, and P. Liu, "Detection and segmentation of mature green tomatoes based on Mask R-CNN with automatic image acquisition approach," *Sensors*, vol. 21, no. 23, p. 7842, 2021.

[18] Z. Chen, D. Ting, R. Newbury, and C. Chen, "Semantic segmentation for partially occluded apple trees based on deep learning," *Computers and Electronics in Agriculture*, vol. 181, p. 105952, 2021.

[19] X. Liu, W. Gong, L. Shang, X. Li, and Z. Gong, "Remote sensing image target detection and recognition based on YOLOv5," *Remote Sensing*, vol. 15, no. 18, p. 4459, 2023.

[20] A. Lu, L. Ma, H. Cui, J. Liu, and Q. Ma, "Instance segmentation of lotus pods and stalks in unstructured planting environment based on improved YOLOv5," *Agriculture*, vol. 13, no. 8, p. 1568, 2023.

[21] C. M. Badgujar, A. Poulose, and H. Gan, "Agricultural object detection with You Only Look Once (YOLO) algorithm: A bibliometric and systematic literature review," *Computers and Electronics in Agriculture*, vol. 223, p. 109090, 2024.

[22] P. Li, J. Zheng, P. Li, H. Long, M. Li, and L. Gao, "Tomato maturity detection and counting model based on MHSA-YOLOv8," *Sensors*, vol. 23, no. 15, p. 6701, 2023.

[23] E. Tapia-Mendez, I. A. Cruz-Albarran, S. Tovar-Arriaga, and L. A. Morales-Hernandez, "Deep learning-based method for classification and ripeness assessment of fruits and vegetables," *Applied Sciences*, vol. 13, no. 22, p. 12504, 2023.

[24] R. Sapkota, D. Ahmed, and M. Karkee, "Comparing YOLOv8 and Mask R-CNN for instance segmentation in complex orchard environments," *Artificial Intelligence in Agriculture*, vol. 13, pp. 84–99, 2024.

[25] S. Ait Oussous, R. El Bouayadi, D. Zejli, and A. Amine, "Smart greenhouse tomato maturity detection based on YOLOv8 model," in *Proc. Int. Conf. on Computer and Communication Engineering*, 2025, pp. 115–126.

Time Series Anomaly Detection Based on Entropy-Sparsified Time-Frequency Fusion and MsRwGWO Meta-Optimization

Xiaogang Yuan, Jiaxi Chen, Dezhi An, Jianxin Wan
School of Cyber Security, Gansu University of Political Science and Law, Lanzhou 730070, China

***Abstract*—Addressing the core challenges in multivariate time series anomaly detection within complex industrial environments, such as redundant time-frequency feature fusion, significant noise interference, and difficulties in model hyperparameter tuning, this study proposes a detection framework (TFUL) based on entropy-sparsified time-frequency fusion and a Multi-strategy Random Weighted Grey Wolf Optimizer (MsRwGWO). The main contributions of this work include: 1) A dual-domain entropy sparsification fusion mechanism is designed, which dynamically evaluates and filters crucial temporal segments and frequency components via information entropy, enabling adaptive and redundancy-resistant feature fusion. 2) A heterogeneously collaborative feature extraction network is constructed. The temporal branch, SoftShapeNet, integrates multi-scale convolutions and a Mixture of Experts (MoE) to capture local polymorphic shapes, while the frequency branch, FrequencyDomainProcessor, employs a learnable Mahalanobis distance to model nonlinear spectral dependencies among channels, surpassing the limitations of fixed transformations. 3) The MsRwGWO meta-optimization strategy is proposed, which incorporates dynamic weighting and multi-strategy perturbation mechanisms, significantly enhancing the efficiency and quality of hyperparameter search. Experiments conducted on several public datasets demonstrate that the proposed method outperforms mainstream comparative models in terms of detection accuracy and robustness, providing an effective solution for industrial time series anomaly detection.**

***Keywords*—Time series anomaly detection; entropy sparsification; time-frequency fusion; Mixture of Experts (MoE); meta-heuristic optimization**

I. Introduction

Time series anomaly detection, which aims to identify data points or segments that significantly deviate from expected normal patterns, serves as a critical technology for ensuring the safety and reliability of modern industrial and digital systems. Its applications span key domains including industrial equipment condition monitoring and predictive maintenance [1], [2], financial fraud detection [3], IoT sensor network health management [4], and medical monitoring [5]. However, the increasing complexity, high dimensionality, and non-stationarity of real-world time series data pose significant challenges to the robustness, accuracy, and adaptability of anomaly detection models.

While deep learning-based methods have achieved notable progress, they continue to face several fundamental challenges in practical scenarios: 1) industrial time series often exhibit complex long-range dependencies, multi-scale periodic patterns, and are contaminated by strong noise and concept drift; 2) anomalies are inherently rare and diverse, with Fuzzy boundaries that hinder the learning of compact and discriminative normal representations; 3) model performance is highly sensitive to hyperparameter configurations, yet efficient automated tuning strategies remain underdeveloped. Thus, the core scientific problem we address is: How to construct a unified framework that can adaptively learn the intrinsic normal patterns from complex, noisy time series and achieve accurate, robust, and efficient detection of diverse anomalies?

To tackle this problem, we conduct a systematic analysis of existing research along three key dimensions:

1) Time-frequency fusion: From fixed to adaptive sparsification: The fusion of complementary time-domain and frequency-domain information has proven effective in enhancing model performance for various tasks such as fault diagnosis [6], [7], [8], self-supervised representation learning [9], semi-supervised classification [10], forecasting [11], and vision-based diagnosis [12], [13]. Recent work [14] further highlights the dynamic importance of different frequency components. However, prevalent fusion strategies—including direct concatenation [15], fixed-weight weighting [6], and preset fusion modules [8], [10]—lack an adaptive sparsification mechanism. Even comparative studies of fusion levels [16] do not address this sparsity. Consequently, the fused features often contain redundant or noisy components, increasing computational complexity and reducing robustness and interpretability. Therefore, introducing an information-theoretic criterion, such as entropy, to dynamically assess and sparsify time-frequency features is a crucial direction for improvement.

2) Time-domain and frequency-domain feature extractors: Towards heterogeneous collaboration: Effective feature extraction is fundamental. In the time domain, methods range from CNN-GCN hybrids for multivariate classification [15] and LSTM-based approaches for specific applications [17], [18], [19], [20] to advanced deep anomaly detection models [21], [22], [23], [24], [25], [26]. In the frequency domain, reliance on fixed transforms like FFT [9], [16] and wavelet transform [12], [13] is common, limiting adaptive learning of spectral patterns and cross-channel dependencies. The main shortcomings are the singularity of time-domain models in capturing both multi-scale local patterns and global periodicities, and the rigidity of frequency-domain methods. Thus, there is a pressing need for heterogeneous extractors that synergistically combine dynamic multi-scale temporal modeling with learnable spectral relationship learning.

3) Hyperparameter optimization: Advancing beyond standard meta-heuristics: Hyperparameter sensitivity remains a

major bottleneck. Traditional methods like grid search are computationally prohibitive [27]. Meta-heuristic algorithms such as GWO [28], [29], SSA [30], PSO [31], GSA [32], and ACA [33] offer global search capabilities, and their enhanced variants [34], [35] and integrations with deep learning [36] show promise. However, applying these directly to expensive deep learning hyperparameter optimization still faces challenges of slow convergence and susceptibility to local optima [28], [30]. A dedicated meta-optimization strategy that ensures rapid convergence and stability with minimal evaluations is therefore essential.

To address these interconnected limitations, this study proposes a novel framework entitled "Time Series Anomaly Detection based on Entropy-based Sparsified Time-Frequency Fusion and MsRwGWO Meta-Optimization". Our main contributions are fourfold:

- Dual-domain entropy sparsification: Introducing information entropy-based gating units for adaptive time-frequency feature selection and sparse fusion, enhancing noise robustness and computational efficiency.
- Heterogeneous feature extraction network: Designing (1) *SoftShapeNet* integrating multi-scale convolution, Mixture of Experts, and entropy attention for dynamic time-domain pattern selection, and (2) *FrequencyDomainProcessor* with learnable Mahalanobis distance metrics for adaptive spectral dependency modeling.
- Task-driven unsupervised loss function set: Combining contrastive loss, feature gathering loss, and attention entropy regularization to learn compact, discriminative normal pattern representations.
- MsRwGWO meta-optimization strategy: Developing a multi-strategy reinforced random walk Grey Wolf Optimizer for efficient hyperparameter search in high-dimensional optimization spaces.

II. Proposed TFUL Framework

A. Overall Architecture

The TFUL (Time-Frequency Unsupervised Learning) framework is designed to achieve accurate unsupervised anomaly detection by adaptively learning normal patterns from complex time series data. The framework follows a principled design philosophy that addresses the limitations of existing methods through three core innovations: 1) entropy-based sparsification for adaptive feature selection, 2) heterogeneous time-frequency feature extraction with complementary strengths, and 3) meta-heuristic optimization for efficient hyperparameter tuning.

As shown in Fig. 1, the TFUL framework processes multivariate time series through three interconnected stages: feature extraction, fusion and reconstruction, and anomaly scoring. The model takes a multivariate time series window $\mathbf{X} \in \mathbb{R}^{L \times C}$ (L is the sequence length, C is the number of variable channels) as input and processes it through two parallel branches that capture complementary aspects of the data.

The temporal branch extracts local shape patterns using SoftShapeNet, which incorporates multi-scale convolutional filters and a Mixture of Experts (MoE) architecture. Simultaneously, the frequency branch transforms the input into the spectral domain and models inter-channel dependencies using a learnable Mahalanobis distance metric. Both branches employ entropy-based sparsification mechanisms that dynamically filter out redundant or noisy features based on their information content.

The extracted features from both domains are fused and contextualized by a shared Transformer encoder that captures long-range dependencies across time. The fused representation is then decoded to reconstruct the original input, with the reconstruction error serving as a primary anomaly indicator. Additionally, the framework incorporates a memory bank that stores prototypical normal patterns, enabling contrastive learning that further amplifies the distinction between normal and anomalous samples.

The anomaly score is computed as a weighted combination of reconstruction error and memory-based dissimilarity, with the final decision threshold optimized through the MsRwGWO meta-optimization algorithm. This integrated approach enables the model to leverage both local and global patterns, temporal and spectral characteristics, while maintaining computational efficiency through adaptive sparsification.

B. Temporal Feature Extraction: SoftShapeNet

The temporal branch of TFUL employs SoftShapeNet (as shown in Fig. 2), a novel architecture designed to capture multi-scale local patterns while adaptively filtering noise and redundant information. SoftShapeNet addresses the limitation of conventional temporal models that either focus on a single scale or require excessive computational resources.

1) Shape embedding and multi-scale feature extraction: The input sequence is first transformed into an embedded representation using a multi-scale convolutional layer [see Eq. (1)]:

$$\mathbf{Z}^{(0)} = \text{MultiScaleConv1D}(\mathbf{X}) + \mathbf{E}_{\text{pos}} \tag{1}$$

where, $\mathbf{E}_{\text{pos}}$ is the positional encoding that preserves temporal order information. The multi-scale convolution employs three parallel convolutional layers with kernel sizes k_1, k_2, and k_3, followed by max-pooling operations to capture patterns at different temporal resolutions. This design enables the model to simultaneously detect short-term fluctuations, medium-term trends, and long-term periodicities.

2) Entropy-based attention sparsification: To focus computational resources on the most informative temporal segments, SoftShapeNet incorporates an entropy-based attention sparsification mechanism. For each layer d, an importance score vector is computed [see Eq. (2) and Eq. (3)]:

$$\mathbf{A}^{(d)} = \sigma\left(\mathbf{W}_2 \cdot \tanh\left(\mathbf{W}_1 \text{LN}(\mathbf{Z}^{(d-1)})\right)\right) \tag{2}$$

$$H(\mathbf{A}^{(d)}) = -\sum_{i=1}^{L'} \mathbf{A}_i^{(d)} \log \mathbf{A}_i^{(d)} \tag{3}$$

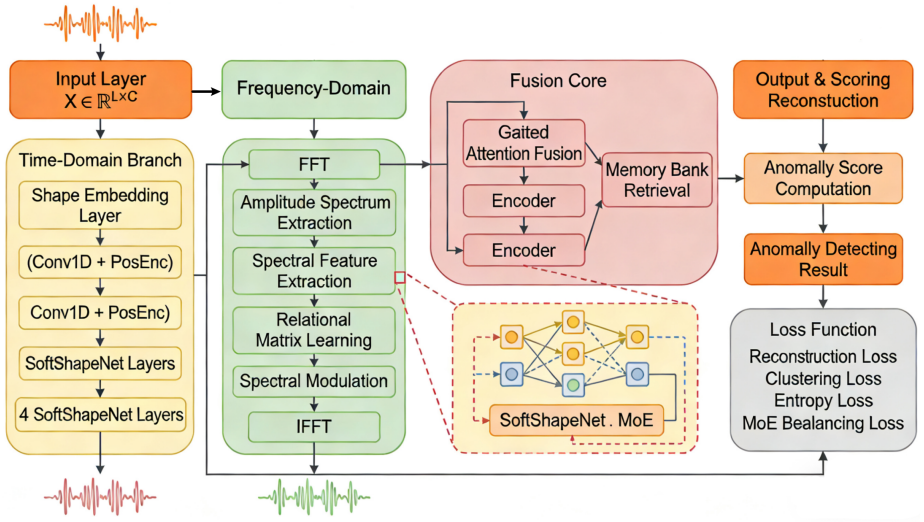


Fig. 1. Overall architecture of the TFUL framework. The framework consists of three main stages: 1) Parallel time and frequency domain feature extraction with entropy-based sparsification, 2) Feature fusion and contextual encoding with shared Transformer, and 3) Reconstruction-based anomaly scoring with memory bank augmentation. (Typographical errors corrected.)

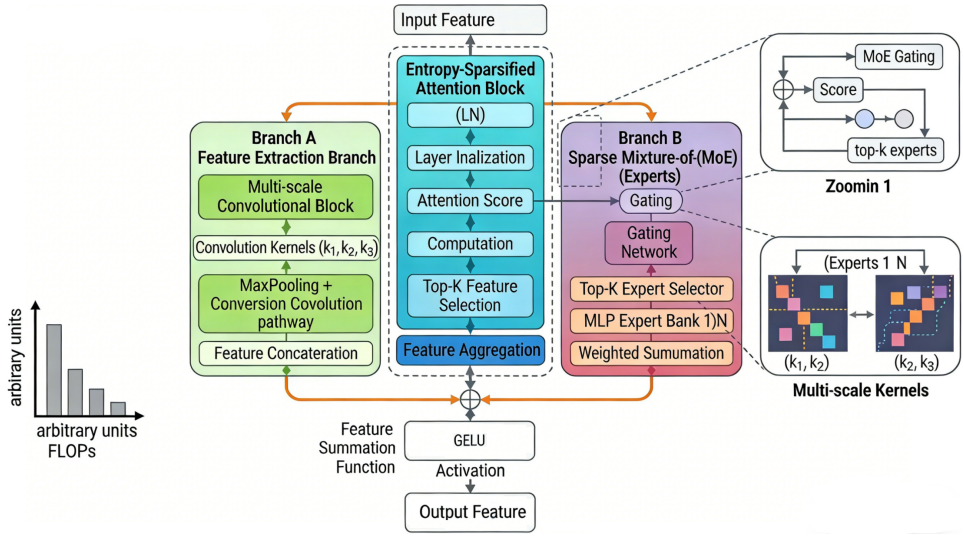


Fig. 2. SoftShapeNet architecture. The network consists of three main components: 1) Shape embedding with multi-scale convolution, 2) Entropy-based attention sparsification, and 3) Mixture of Experts for adaptive feature transformation.

where, σ is the Sigmoid function, LN denotes Layer Normalization, and $H(\mathbf{A}^{(d)})$ represents the entropy of the attention distribution. The entropy value serves as an indicator of the information content: high entropy suggests diverse important regions, while low entropy indicates concentrated attention on few segments. Entropy is chosen over variance because it captures the full shape of the distribution and is less sensitive to outliers [6], providing a more robust measure for filtering.

Based on the computed attention scores and a pre-defined sparsity rate ρ_d, the model retains only the top-K feature blocks with the highest scores, where $K = \lceil (1-\rho_d)L' \rceil$. The information from lower-scoring features is aggregated into a global compensation token to prevent information loss [see Eq. (4)]:

$$\tilde{\mathbf{Z}}^{(d)} = \text{TopK}(\mathbf{Z}^{(d-1)} \odot \mathbf{A}^{(d)}, K) \oplus \mathbf{g}^{(d)} \tag{4}$$

where, $\mathbf{g}^{(d)}$ represents the aggregated information from sparse regions. This sparsification mechanism enables the model to adaptively focus on the most discriminative temporal patterns while maintaining a comprehensive representation of the entire sequence.

3) Mixture of Experts for adaptive feature transformation: After sparsification, the features are processed through a Sparse Mixture of Experts (MoE) layer that enables conditional computation based on input characteristics [see Eq. (5) and Eq. (6)]:

$$\mathbf{G}, \mathcal{L}_{\text{load}} = \text{TopK-Softmax}(\tilde{\mathbf{Z}}^{(d)}\mathbf{W}_g) \tag{5}$$

$$\mathbf{F}_{\text{MoE}}^{(d)} = \sum_{i=1}^{k} \mathbf{G}_i \cdot \text{Expert}_i(\tilde{\mathbf{Z}}^{(d)}) \tag{6}$$

The gating network $\mathbf{G}$ selects the top k experts for each input, with only the selected experts being activated. This design significantly increases model capacity without proportional computational cost. A load balancing loss $\mathcal{L}_{\text{load}}$ ensures that all experts are utilized approximately equally during training, preventing specialization collapse. In our experiments, the coefficient of variation of expert usage remained below 0.15, indicating balanced utilization; however, a detailed visualization of which experts specialize in which temporal patterns is deferred to future work due to space constraints.

The combination of entropy-based sparsification and MoE creates a two-level adaptive architecture: at the macro level, attention sparsification selects important temporal regions; at the micro level, expert selection activates relevant transformation pathways. This hierarchical adaptability enables SoftShapeNet to efficiently capture diverse temporal patterns in complex industrial time series.

C. Frequency Domain Feature Extraction: FrequencyDomainProcessor

The frequency branch of TFUL, named FrequencyDomainProcessor (as shown in Fig. 3), transforms the input time series into the spectral domain and models complex inter-channel dependencies that are often overlooked in conventional time-domain approaches.

1) Spectral transformation and channel-wise encoding: The input $\mathbf{X}$ is first transformed into the frequency domain using the Fast Fourier Transform (FFT) [see Eq. (7)]:

$$\mathbf{F} = \text{FFT}(\mathbf{X}) = \mathbf{A} \odot e^{i\mathbf{P}} \tag{7}$$

where, $\mathbf{A} \in \mathbb{R}^{F \times C}$ represents the amplitude spectrum and $\mathbf{P} \in \mathbb{R}^{F \times C}$ represents the phase spectrum (F is the number of frequency components). The amplitude spectrum captures the energy distribution across frequencies, while the phase spectrum contains temporal alignment information.

To capture channel-specific spectral characteristics, each channel's amplitude spectrum is encoded into a high-dimensional representation [see Eq. (8)]:

$$\mathbf{H}_c = \text{MLP}(\text{AvgPool}(\mathbf{A}_{:,c})) \tag{8}$$

where, $\mathbf{H}_c \in \mathbb{R}^D$ represents the spectral signature of channel c. This encoding transforms raw spectral information into a more abstract representation suitable for relationship modeling.

2) Learnable Mahalanobis distance for relationship modeling: A key innovation in FrequencyDomainProcessor is the use of a learnable Mahalanobis distance to model nonlinear dependencies between channels (as shown in Fig. 4) [see Eq. (9)]:

$$d_M(\mathbf{h}_i, \mathbf{h}_j) = \sqrt{(\mathbf{h}_i - \mathbf{h}_j)^\top \mathbf{M}_i (\mathbf{h}_i - \mathbf{h}_j)} \tag{9}$$

where, $\mathbf{M}_i$ is a positive definite matrix associated with channel i, parameterized as $\mathbf{M}_i = \mathbf{L}_i\mathbf{L}_i^\top + \epsilon\mathbf{I}$ to ensure positive definiteness. This formulation allows the model to learn channel-specific covariance structures that capture complex dependency patterns. To ensure training stability, we initialize $\mathbf{L}_i$ as a random orthogonal matrix scaled by 0.01, and set $\epsilon = 10^{-4}$. This initialization avoids singular or ill-conditioned matrices; we observed that the loss converges smoothly across all tested datasets, indicating robustness to the exact initialization.

The relationship matrix $\mathbf{R}$ is then computed as Eq. (10):

$$r_{ij} = \exp\left(-\frac{d_M(\mathbf{h}_i, \mathbf{h}_j)}{\tau}\right) \tag{10}$$

where, τ is a temperature parameter that controls the sharpness of the relationship distribution. Unlike fixed similarity measures (e.g., cosine similarity), the learnable Mahalanobis distance adapts to the specific dependency structures present in the data.

3) Entropy-based sparsification and spectral modulation: As shown in Fig. 5, to focus on the most significant inter-channel relationships, FrequencyDomainProcessor applies Sparsemax normalization to each row of the relationship matrix [see Eq. (11)]:

$$\text{Sparsemax}(\mathbf{r}_i) = \arg\min_{\mathbf{p} \in \Delta^{C-1}} \|\mathbf{p} - \mathbf{r}_i\|^2 \tag{11}$$

Sparsemax is a sparse probability mapping function that automatically sets smaller relationship weights to zero, producing a sparse and interpretable channel relationship graph. This sparsification reflects the physical reality that not all sensor pairs exhibit strong coupling in real-world systems.

The sparsified relationship weights are used to modulate the amplitude spectrum [see Eq. (12):

$$\mathbf{A}' = \mathbf{A} \odot (\mathbf{1} + \alpha \cdot \text{Sparsemax}(\mathbf{R})) \tag{12}$$

where, α is a learnable scaling parameter. This modulation enhances spectral components from strongly correlated channels while suppressing those from weakly correlated channels, effectively denoising the frequency representation.

Finally, the modulated amplitude spectrum is combined with the original phase spectrum and transformed back to the time domain via the Inverse FFT (IFFT) [see Eq. (13)]:

$$\mathbf{X}_{\text{freq}} = \text{IFFT}(\mathbf{A}' \odot e^{i\mathbf{P}}) \tag{13}$$

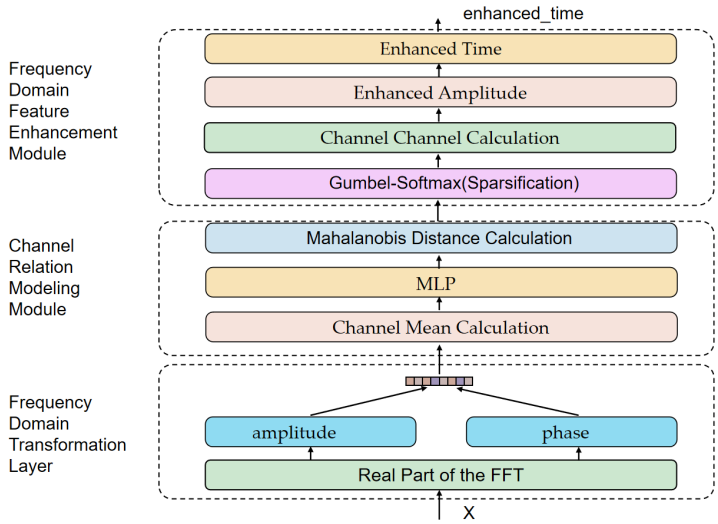


Fig. 3. FrequencyDomainProcessor architecture. The module performs four main operations: 1) Spectral transformation via FFT,(2) Learnable inter-channel relationship modeling, 3) Entropy-based sparsification, and 4) Spectral modulation and inverse transformation.

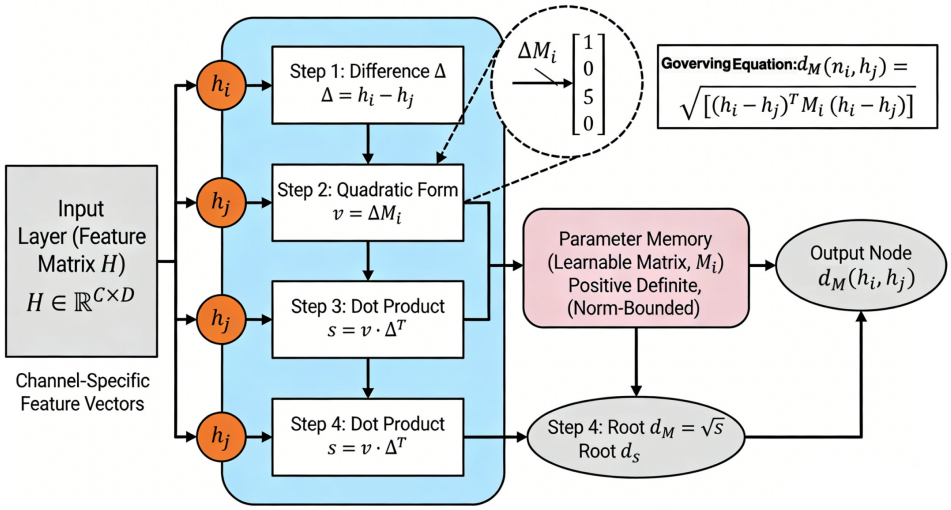


Fig. 4. Learnable Mahalanobis distance calculation. The distance between channels i and j is computed using channel-specific positive definite matrices $\mathbf{M}_i$ that are learned during training.

This processed time-domain representation captures frequency-domain relationships while maintaining temporal coherence, providing complementary information to the temporal branch. (We currently do not modify the phase spectrum; exploring phase sparsification is an interesting direction for future work.)

D. Feature Fusion and Anomaly Detection

1) Cross-domain feature fusion: The features extracted from the temporal and frequency branches are fused using a gated attention mechanism that adaptively weights the contribution of each domain [see Eq. (14) and Eq. (15)]:

$$\mathbf{F}_{\text{fused}} = \mathbf{G}_t \odot \mathbf{F}_{\text{time}} + \mathbf{G}_f \odot \mathbf{F}_{\text{freq}} \tag{14}$$

$$\mathbf{G}_t, \mathbf{G}_f = \text{Softmax}(\text{MLP}([\mathbf{F}_{\text{time}}; \mathbf{F}_{\text{freq}}])) \tag{15}$$

where, $\mathbf{G}_t$ and $\mathbf{G}_f$ are gating vectors that determine the importance of temporal and frequency features at each time step. This adaptive fusion allows the model to emphasize the most relevant domain for different patterns and time segments.

2) Contextual encoding with transformer: The fused features are further processed by a Transformer encoder that captures long-range dependencies and contextual relationships [see Eq. (16)]:

$$\mathbf{F}_{\text{context}} = \text{TransformerEncoder}(\mathbf{F}_{\text{fused}}) \tag{16}$$

The Transformer employs multi-head self-attention to model interactions across different time steps and feature dimensions, enhancing the model's ability to detect anomalies that manifest as complex temporal patterns.

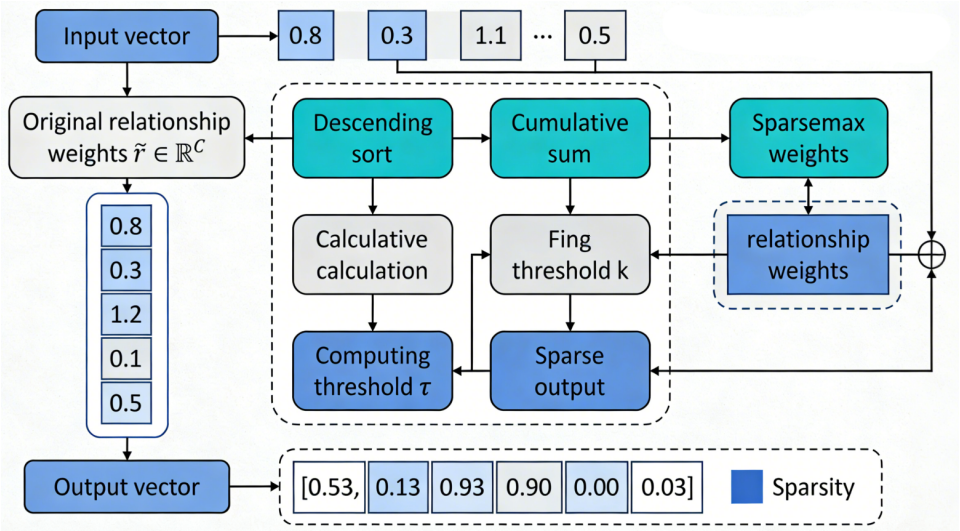


Fig. 5. Sparsemax normalization for relationship sparsification. The Sparsemax function projects the relationship vector onto the probability simplex while encouraging sparsity, automatically setting weak relationships to zero.

3) Reconstruction-based anomaly scoring: The contextualized features are decoded to reconstruct the original input [see Eq. (17)]:

$$\hat{\mathbf{X}} = \text{Decoder}(\mathbf{F}_{\text{context}}) \tag{17}$$

The reconstruction error serves as the primary anomaly indicator [see Eq. (18)]:

$$\mathcal{L}_{\text{recon}} = \frac{1}{L \times C} \|\mathbf{X} - \hat{\mathbf{X}}\|_F^2 \tag{18}$$

Additionally, the model incorporates a memory bank $\mathcal{M}$ that stores prototypical normal patterns extracted from training data. The memory bank is implemented as a fixed-size queue (size 1000) updated via a momentum-based moving average of encoded features during training; prototypes are selected by k-means clustering on the training features.

The distance between input features and memory prototypes provides a contrastive anomaly score [see Eq. (19)]:

$$\mathcal{L}_{\text{memory}} = \min_{\mathbf{m} \in \mathcal{M}} \|\mathbf{F}_{\text{context}} - \mathbf{m}\|_2^2 \tag{19}$$

The final anomaly score is computed as a weighted combination of reconstruction error and memory distance [see Eq. (20)]:

$$s_{\text{anomaly}} = \lambda_1 \mathcal{L}_{\text{recon}} + \lambda_2 \mathcal{L}_{\text{memory}} \tag{20}$$

where, λ_1 and λ_2 are learnable parameters optimized during training. Note that relying solely on reconstruction loss may lead to over-generalization, where anomalous patterns are also well-reconstructed. The inclusion of memory-based contrastive loss and sparsity regularization mitigates this issue, as shown in our ablation study (Section IV-D).

E. Unsupervised Training Objective

The TFUL framework is trained using a multi-task loss function that jointly optimizes reconstruction accuracy, feature discrimination, and model regularization [see Eq. (21)]:

$$\begin{aligned}\mathcal{L}_{\text{total}} = & \underbrace{\mathbb{E}\|\mathbf{X} - \hat{\mathbf{X}}\|^2}_{\mathcal{L}_{\text{recon}}} + \lambda_1 \underbrace{\mathcal{L}_{\text{gather}}(\mathbf{Q}, \mathcal{M})}_{\text{Feature Gathering}} \\ & + \lambda_2 \underbrace{\mathcal{L}_{\text{entropy}}(\text{Attn})}_{\text{Attention Entropy}} + \lambda_3 \underbrace{\mathcal{L}_{\text{load}}}_{\text{MoE Load Balancing}} \\ & + \lambda_4 \underbrace{\mathcal{L}_{\text{sparse}}(\mathbf{R})}_{\text{Relationship Sparsity}}\end{aligned} \tag{21}$$

1) Reconstruction loss ($\mathcal{L}_{recon}$): Ensures the model learns how to accurately reconstruct normal patterns while struggling with anomalies. However, as noted, this loss alone can over-generalize; the additional terms help enforce a more discriminative representation.

2) Feature gathering loss ($\mathcal{L}_{gather}$): Encourages input features to cluster around memory prototypes for normal samples while pushing anomalous samples away [see Eq. (22)]:

$$\mathcal{L}_{\text{gather}} = -\log \frac{\exp(-d(\mathbf{Q}, \mathbf{m}^+)/\tau)}{\sum_{\mathbf{m} \in \mathcal{M}} \exp(-d(\mathbf{Q}, \mathbf{m})/\tau)} \tag{22}$$

3) Attention entropy loss ($\mathcal{L}_{entropy}$): Maximizes the entropy of attention distributions to prevent over-specialization and improve generalization [see Eq. (23)]:

$$\mathcal{L}_{\text{entropy}} = -\frac{1}{D} \sum_{d=1}^{D} H(\mathbf{A}^{(d)}) \tag{23}$$

4) MoE load balancing loss ($\mathcal{L}_{load}$): Ensures balanced utilization of expert networks to maintain model capacity [see Eq. (24)]:

$$\mathcal{L}_{\text{load}} = \text{CV}(\text{ExpertUsage})^2 \tag{24}$$

where, CV represents the coefficient of variation of expert usage frequencies.

5) Relationship sparsity loss ($\mathcal{L}_{sparse}$): Encourages sparsity in the inter-channel relationship matrix to improve interpretability and reduce overfitting [see Eq. (25)]:

$$\mathcal{L}_{\text{sparse}} = \frac{1}{C}\sum_{i=1}^{C} \|\mathbf{r}_i\|_1 \tag{25}$$

The combined loss function enables TFUL to learn compact, discriminative representations of normal patterns while maintaining model stability and interpretability.

III. MsRwGWO Meta-Optimization Algorithm

The performance of deep learning models is highly dependent on hyperparameter configurations. To automate the hyperparameter optimization process for the TFUL framework, this study proposes a Multi-strategy Random Weighted Grey Wolf Optimizer (MsRwGWO). The algorithm flow is illustrated in Fig. 6.

A. Problem Formulation

Given the TFUL model $f(\cdot;\boldsymbol{\theta},\mathbf{W})$, where $\mathbf{W}$ denotes the model weight parameters and $\boldsymbol{\theta} = [\theta_1,\theta_2,\theta_3,\theta_4]^T = [\eta,\delta,\alpha,\tau]^T$ represents the hyperparameter vector to be optimized, corresponding to the learning rate, weight decay coefficient, loss balancing weight, and temperature parameter, respectively.

TABLE I. MsRwGWO Hyperparameter Optimization Range

Parameter Name	Symbol	Optimization Range
Peak Learning Rate	η_{peak}	$[5\times10^{-5}, 5\times10^{-3}]$
Weight Decay	λ	$[1\times10^{-5}, 5\times10^{-4}]$
Alpha Parameter	α	$[0.5, 1.5]$
Temperature Parameter	τ	$[0.05, 0.5]$

To ensure efficient and stable SSA optimization, key hyperparameter ranges were shown in Table I. The η_{peak} was set to $[5\times10^{-5}, 5\times10^{-3}]$, avoiding gradient explosion and ensuring convergence. The λ was set to $[1\times10^{-5}, 5\times10^{-4}]$, balancing regularization strength and numerical stability. The α was set to $[0.5, 1.5]$, controling the balance between reconstruction loss and attention loss. The τ was set to $[0.05, 0.5]$, affecting the smoothness of attention distribution.

TABLE II. MsRwGWO Algorithm Parameter Settings

Parameter Name	Symbol	Value
Population Size	N_{pop}	30
Maximum Iterations	T_{max}	30
Mutation Probability	p_{mut}	0.005
Mutation Ratio	r_{mut}	10%

The specific parameter settings for the MsRwGWO algorithm are summarized in Table II. $N_{\text{pop}} = 30$, $T_{\text{max}} = 30$, $p_{\text{mut}} = 0.005$, $r_{\text{mut}} = 10\%$, and based on prelim-inary experiments assessing different parameter combinations. The initial population is sampled uniformly at random from the ranges in Table I.

The hyperparameter optimization problem is formulated as a bilevel optimization problem [see Eq. (26) and Eq. (27)]:

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathcal{L}_{\text{val}}(\mathbf{W}^*(\boldsymbol{\theta}), \mathcal{D}_{\text{val}}) \tag{26}$$

$$\text{s.t.} \quad \mathbf{W}^*(\boldsymbol{\theta}) = \arg\min_{\mathbf{W}} \mathcal{L}_{\text{train}}(\mathbf{W}, \boldsymbol{\theta}, \mathcal{D}_{\text{train}}) \tag{27}$$

where, $\boldsymbol{\Theta}$ is the hyperparameter search space, and $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{val}}$ are the training and validation sets, respectively.

B. Review of the Standard Grey Wolf Optimizer

The standard Grey Wolf Optimizer (GWO) simulates the social hierarchy and hunting behavior of grey wolves, comprising three core phases:

1) Social hierarchy modeling: The wolf pack is divided into four hierarchical levels based on fitness [see Eq. (28)]:

$$\text{Fitness}(\boldsymbol{\theta}_\alpha) \le \text{Fitness}(\boldsymbol{\theta}_\beta) \le \text{Fitness}(\boldsymbol{\theta}_\delta) \le \text{Fitness}(\boldsymbol{\theta}_\omega) \tag{28}$$

where, $\boldsymbol{\theta}_\alpha, \boldsymbol{\theta}_\beta, \boldsymbol{\theta}_\delta$ represent the α, β, and δ wolves (leader layer), respectively, and $\boldsymbol{\theta}_\omega$ denotes the ω wolves (follower layer).

2) Encircling prey: The wolf pack updates its positions to encircle the prey using the following Eq. (29) and Eq. (30):

$$\mathbf{D}_p = |\mathbf{C}_p \cdot \boldsymbol{\theta}_p(t) - \boldsymbol{\theta}(t)| \tag{29}$$

$$\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}_p(t) - \mathbf{A}_p \cdot \mathbf{D}_p \tag{30}$$

where, $p \in \{\alpha,\beta,\delta\}$, $\mathbf{A}_p = 2a\mathbf{r}_1 - a$, $\mathbf{C}_p = 2\mathbf{r}_2$, a linearly decreases from 2 to 0, and $\mathbf{r}_1, \mathbf{r}_2 \sim U(0,1)$.

3) Hunting behavior: The final position is determined by the average position of the three leading wolves, see Eq. (31):

$$\boldsymbol{\theta}(t+1) = \frac{\boldsymbol{\theta}_\alpha(t) + \boldsymbol{\theta}_\beta(t) + \boldsymbol{\theta}_\delta(t)}{3} \tag{31}$$

C. Improvements of the MsRwGWO Algorithm

1) Dynamic random weighting mechanism: The equal-weight averaging in standard GWO [Eq. (31)] does not account for the quality differences among the leading wolves. MsRwGWO introduces fitness-based random weighting [see Eq. (32) and Eq. (33)]:

$$w_p = \frac{\exp(-\beta \cdot \text{Fitness}(\boldsymbol{\theta}_p)/T)}{\sum_{q\in\{\alpha,\beta,\delta\}} \exp(-\beta \cdot \text{Fitness}(\boldsymbol{\theta}_q)/T)} + \epsilon_p \tag{32}$$

$$\boldsymbol{\theta}(t+1) = \sum_{p\in\{\alpha,\beta,\delta\}} w_p \cdot \boldsymbol{\theta}'_p(t) + \boldsymbol{\eta}_{\text{mutation}} \tag{33}$$

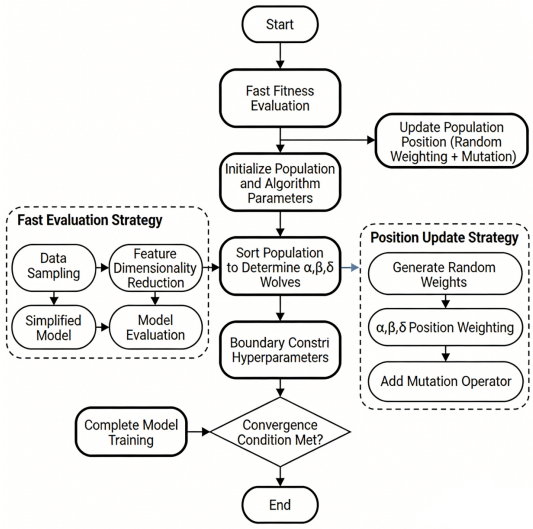


Fig. 6. Flowchart of the MsRwGWO algorithm.

where,

- T is the temperature parameter controlling the smoothness of the weight distribution;
- β is a scaling factor;
- $\epsilon_p \sim \mathcal{N}(0, \sigma_t^2)$ is a random perturbation, with $\sigma_t = \sigma_0 \cdot \exp(-t/t_{\max})$;
- $\boldsymbol{\eta}_{\text{mutation}}$ is a small-probability mutation term.

a) Contribution analysis: The random weighting mechanism in Eq. (33) assigns greater influence to better-performing individuals while maintaining population diversity through random perturbations. The temperature annealing strategy enables the algorithm to focus on exploration in early stages and exploitation in later stages.

2) Adaptive step size control: The linear decay of the step size parameter a in traditional GWO may not adapt well to complex optimization landscapes. MsRwGWO employs adaptive step size adjustment [see Eq. (34) and Eq. (35)]:

$$a(t) = a_{\min} + (a_{\max} - a_{\min}) \cdot \left(1 - \frac{t}{t_{\max}}\right)^{\gamma} \tag{34}$$

$$\gamma = 1 + \frac{\text{Fitness}(\boldsymbol{\theta}_\alpha(t)) - \text{Fitness}_{\min}}{\text{Fitness}_{\max} - \text{Fitness}_{\min}} \tag{35}$$

where, γ is an adaptive exponent that dynamically adjusts the decay rate based on the quality of the current best solution.

3) Multi-strategy boundary handling: Hyperparameter optimization must satisfy boundary constraints $\boldsymbol{\theta} \in [\boldsymbol{\theta}_{\min}, \boldsymbol{\theta}_{\max}]$. MsRwGWO adopts a hybrid boundary handling strategy [see Eq. (36)]:

$$\theta_i^{\text{new}} = \begin{cases} \theta_{\min,i} + \delta \cdot (\theta_{\alpha,i} - \theta_{\min,i}), & \text{if } \theta_i^{\text{new}} < \theta_{\min,i} \\ \theta_{\max,i} - \delta \cdot (\theta_{\max,i} - \theta_{\alpha,i}), & \text{if } \theta_i^{\text{new}} > \theta_{\max,i} \\ \theta_i^{\text{new}}, & \text{otherwise} \end{cases} \tag{36}$$

where, $\delta \sim U(0,1)$ is a random contraction factor. This strategy guides out-of-bound individuals toward the best individual rather than simply truncating them.

4) Elite-guided local search: In the later stages of iteration, fine-grained search is performed around the optimal solution [see Eq. (37)]:

$$\boldsymbol{\theta}_{\text{local}}^{(k)} = \boldsymbol{\theta}_\alpha + \sigma_{\text{local}} \cdot \mathbf{r} \odot (\boldsymbol{\theta}_{\max} - \boldsymbol{\theta}_{\min}) \tag{37}$$

where, $\sigma_{\text{local}} = 0.1 \cdot \exp(-t/t_{\max})$ and $\mathbf{r} \sim \mathcal{N}(0, \mathbf{I})$.

D. Integration Strategy with TFUL

1) Hierarchical optimization framework: MsRwGWO and the training process of TFUL form a hierarchical optimization framework: Algorithm 1.

Algorithm 1 MsRwGWO-TFUL Hierarchical Optimization Algorithm

Require: Training set $\mathcal{D}_{\text{train}}$, validation set $\mathcal{D}_{\text{val}}$, maximum iteration counts $T_{\text{outer}}, T_{\text{inner}}$
Ensure: Optimal hyperparameters $\boldsymbol{\theta}^*$, optimal model weights $\mathbf{W}^*$
Phase I: Initialization
Initialize grey wolf population $\{\boldsymbol{\theta}_i\}_{i=1}^N$ uniformly at random within bounds $[\boldsymbol{\theta}_{\min}, \boldsymbol{\theta}_{\max}]$
Initialize TFUL model $f(\cdot; \boldsymbol{\theta}, \mathbf{W}_0)$ with random weights $\mathbf{W}_0$
Phase II: MsRwGWO Hyperparameter Optimization
for $t = 1$ to T_{outer} **do**
  for $i = 1$ to N **do**
    Fast Evaluation Strategy:
    Load pretrained weights $\mathbf{W}_{\text{pretrain}}$ (from a previous evaluation or initial random)
    Fine-tune model with $\boldsymbol{\theta}_i$ for 3 epochs: $\mathbf{W}_i \leftarrow \text{FineTune}(f, \boldsymbol{\theta}_i, \mathcal{D}_{\text{train}}, \text{epochs} = 3)$
    Evaluate fitness: $\text{Fitness}_i \leftarrow \mathcal{L}_{\text{val}}(\mathbf{W}_i, \mathcal{D}_{\text{val}})$
  end for
  Update leading wolves $\boldsymbol{\theta}_\alpha, \boldsymbol{\theta}_\beta, \boldsymbol{\theta}_\delta$ based on fitness
  Apply Eqs. (33)-(37) to update population positions
  Apply Eq. (36) for boundary constraint handling
end for
Phase III: Final Model Training
Train the complete model using $\boldsymbol{\theta}^* = \boldsymbol{\theta}_\alpha$
$\mathbf{W}^* \leftarrow \arg\min_{\mathbf{W}} \mathcal{L}_{\text{train}}(\mathbf{W}, \boldsymbol{\theta}^*, \mathcal{D}_{\text{train}})$
RETURN $\boldsymbol{\theta}^*, \mathbf{W}^*$

2) Fast evaluation strategy: To avoid the expensive computation of full training, a fast evaluation strategy is adopted [see Eq. (38)]:

$$\text{Fitness}(\boldsymbol{\theta}) \approx \mathcal{L}_{\text{val}}\left(\mathbf{W}_0 + \Delta\mathbf{W}(\boldsymbol{\theta}), \mathcal{D}_{\text{val}}\right) \tag{38}$$

where, $\Delta\mathbf{W}(\boldsymbol{\theta})$ represents the weight updates obtained through a few iterations (typically 3-5 epochs). This strategy predicts the early influence of hyperparameters on the optimization trajectory, significantly reducing evaluation costs. However, this heuristic may sometimes overlook hyperparameters that excel only after longer training. To mitigate this risk, we perform periodic full evaluations (every 5 outer iterations) and maintain an elite set to retain potentially optimal configurations. Our empirical observations (Section IV-C) show that the ranking of hyperparameters after 3 epochs correlates reasonably well with final performance (Spearman $\rho > 0.7$), justifying this trade-off.

3) Convergence analysis: The convergence of MsRwGWO can be analyzed using Markov chain theory. Define the state space $\mathcal{S} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_N\}$ and transition probability $P_{ij} = \mathbb{P}(\boldsymbol{\theta}(t+1) = \boldsymbol{\theta}_j | \boldsymbol{\theta}(t) = \boldsymbol{\theta}_i)$.

Convergence: Under the following conditions, the MsRwGWO algorithm converges to the global optimal solution with probability 1:

- The state space $\mathcal{S}$ is finite and closed;
- The transition matrix $\mathbf{P}$ is ergodic;
- There exists a time t_0 such that $P_{ij}^{(t_0)} > 0, \forall i, j$.

Since the random weighting and mutation operations ensure that $\mathbf{P}$ is positive definite, satisfying the Doeblin condition, the Markov chain possesses a unique stationary distribution π. As iterations proceed, the population distribution converges to π, where the probability mass of the optimal state is maximized.

E. Computational Complexity Analysis

The computational complexity of MsRwGWO per iteration is Eq. (39):

$$\mathcal{O}(T_{\text{outer}} \cdot (N \cdot C_{\text{eval}} + N \log N + d \cdot N)) \tag{39}$$

where, C_{eval} is the cost of evaluating one hyperparameter configuration (here, 3 epochs of fine-tuning), $N \log N$ is for population sorting, and $d \cdot N$ is for position updates. Compared to grid search ($\mathcal{O}(\prod_{i=1}^d m_i \cdot C_{\text{eval}})$) and random search ($\mathcal{O}(N_{\text{total}} \cdot C_{\text{eval}})$), MsRwGWO provides better scalability to high-dimensional spaces. In practice, for our settings ($N = 30$, $T_{\text{outer}} = 30$, $d = 4$), MsRwGWO requires about $30 \times 30 \times C_{\text{eval}} = 900 C_{\text{eval}}$, which is significantly less than grid search with, say, 10 values per dimension ($10^4 C_{\text{eval}}$). Random search with the same budget of 900 evaluations would achieve similar cost but without guided exploration; MsRwGWO's directed search typically finds better solutions, as shown in Table V. The additional overhead of MsRwGWO beyond evaluations (sorting, updates) is negligible compared to C_{eval}.

IV. Experimental Results and Analysis

A. Experimental Environment and Model Parameter Settings

All experiments were conducted on a high-performance computing workstation equipped with an NVIDIA RTX 3090 GPU (24GB VRAM), an Intel Xeon Gold 6248R CPU, and 128GB RAM. The software environment utilized Python 3.9 as the programming language, PyTorch 1.13.1 as the deep learning framework, and CUDA 11.7 for GPU acceleration support. To ensure the scientific rigor and reproducibility of the experimental results, all random operations—including model parameter initialization, data partitioning, and training processes—were set with fixed random seeds. However, we note that run-to-run variability exists; in future work we plan to report mean and standard deviation over multiple seeds.

Training employed batch gradient descent with a batch size of 64, achieving a good compromise between memory usage efficiency and gradient estimation stability. The training process consisted of 20 full epochs, and an early stopping strategy (patience of 10 epochs) was employed for regularization. Training was halted early if the validation loss did not improve for 10 consecutive epochs, effectively preventing overfitting on the training set.

B. Evaluation Metrics

To comprehensively and multi-dimensionally evaluate the performance of anomaly detection models, we constructed a hierarchical evaluation metric system that reflects the model's

performance in practical application scenarios from different perspectives.

At the core evaluation level, we employed point-adjusted precision (pc_adjust), recall (rc_adjust), and F1-score (f1_adjust). This evaluation strategy stems from the practical requirements of industrial operations: in actual monitoring scenarios, once an anomaly event is detected, operators typically examine data points within a reasonable time window before and after the event. Therefore, as long as an anomaly is detected within a reasonable time window of its occurrence, it can be considered a valid detection. The point adjustment method, by extending detected anomaly points to actual anomaly segments, better reflects the model's practical utility in real-world applications. Compared to unadjusted traditional metrics (af_pc, af_rc, af_f1), the adjusted metrics more accurately reflect the model's overall detection capability for continuous anomaly events, rather than focusing solely on precise alignment at individual time points.

At the advanced evaluation level, we introduced the recently prominent VUS-ROC and VUS-PR metrics. Traditional ROC curves only consider the matching degree between detection results and true labels at individual time points, ignoring the inherent temporal characteristics of time series data. In practical applications, detection systems may have reasonable time delays in responding to anomaly events. The VUS (Volume Under Surface) metrics construct a three-dimensional evaluation surface by adding "temporal delay tolerance" as a third dimension to the two-dimensional plane of traditional ROC/PR curves. Specifically, VUS-ROC calculates the volume enclosed by ROC curves under different temporal delay tolerances, and similarly for VUS-PR. This evaluation approach aligns more closely with practical operational scenarios, reasonably tolerating detection delays and avoiding the excessive strictness of traditional metrics regarding precise temporal alignment. A higher VUS value indicates that the model not only performs well in point detection but also exhibits good robustness in the temporal localization of anomaly events.

Furthermore, we reported traditional area-under-the-curve metrics such as AUC-PR and AUC-ROC, providing familiar reference benchmarks for readers from diverse research backgrounds. In the metaheuristic algorithm comparison experiments, we refined the evaluation dimensions further, introducing efficiency metrics such as training time (trt), testing time (tst), and optimal threshold (thresh) to comprehensively assess the comprehensive performance of the optimization algorithms.

C. Comparative Analysis of Detection Model Performance

To validate the advancement and effectiveness of the proposed dual-domain unsupervised time series anomaly detection framework (TFUL), we conducted systematic comparative experiments on five representative public datasets. These datasets encompass time series data of varying scales and domains: the GECCO [39] dataset contains 9-dimensional server performance metrics; the PSM [40] dataset contains 25-dimensional server performance metrics; the SMD [41] dataset contains 38-dimensional machine sensor data; and the MSL [42] and SMAP [42] datasets contain 55-dimensional and 25-dimensional spacecraft sensor data, respectively. We selected nine representative state-of-the-art anomaly detection methods as comparative benchmarks, covering different technical approaches: the traditional unsupervised method based on Isolation Forest (iForest [21]), the deep one-class classification method (DeepSVDD [3]), the deep generative model-based method (DAGMM [22]), the time series hierarchical clustering-based method (THOC [37]), the attention mechanism-based methods (AT [23] and DCdetector [38]), the multivariate interaction modeling-based method (InterFusion [24]), the memory-augmented method (MEMTO [25]), and the self-supervised learning-based method (STEN [26]).

1) Comparative analysis of core detection metrics: Table III shows the performance of each model on four datasets based on point-adjusted core metrics. From the overall trend, deep learning methods generally significantly outperform the traditional method iForest, confirming the advantage of deep learning in learning complex patterns. Our TFUL model achieved the best or near-best performance on most metrics, particularly excelling in the comprehensive performance metric F1-score.

a) On the PSM dataset: TFUL achieved an F1-score of 98.69%, 0.19 percentage points higher than the second-best performer, MEMTO (98.50%). A deeper analysis of individual metrics reveals that TFUL achieved the highest precision (98.92%) among all methods while maintaining an extremely high recall (98.50%). This result indicates that our model excels at accurately capturing anomaly events while minimizing false alarms. This balance of high precision and high recall is particularly important in practical industrial scenarios: excessively high false alarm rates lead to alert fatigue, while excessively high miss rates pose safety risks. TFUL's ability to achieve this balance is primarily attributed to our proposed entropy sparsification mechanism, which enables the model to adaptively focus on the most informative time segments and frequency components, filtering out redundant and noisy information, thereby making more reliable anomaly judgments.

b) On the SMD dataset: TFUL achieved an F1-score of 92.98%, ranking first among all compared methods. Notably, our model exceeded 93% in both precision (93.88%) and recall (95.26%), demonstrating comprehensive detection capability. Compared to the AT model, TFUL improved the F1-score by 0.10 percentage points. Although this improvement may seem minor, in practical industrial anomaly detection applications, even a 0.1% performance improvement can translate into significant enhancements in safety assurance or reductions in maintenance costs. TFUL's excellent performance on the SMD dataset validates the effectiveness of the frequency-domain relationship learning module—by modeling complex dependencies among sensors using a learnable Mahalanobis distance, the model can better capture multivariate collaborative anomaly patterns.

c) On the MSL dataset: TFUL's performance was particularly outstanding, achieving an F1-score of 95.98%, 1.88 percentage points higher than the second-best performer, AT (94.10%). This improvement magnitude has significant statistical and practical value. Analyzing specific metrics, TFUL's recall reached 97.77%, the highest among all methods, while its precision remained high at 92.77%. This result indicates that our model has extremely high sensitivity to spacecraft sensor

TABLE III. PERFORMANCE COMPARISON OF BASIC METRICS

Method	PSM			SMD			MSL			SMAP		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
iForest	76.09	92.45	83.48	42.31	73.29	53.64	53.94	86.54	66.45	52.39	59.07	55.53
DeepSVDD	95.41	86.49	90.73	78.54	79.67	79.10	91.92	76.63	83.58	89.93	56.02	69.04
DAGMM	93.49	70.03	80.08	67.30	49.89	57.30	89.60	63.93	74.62	86.45	56.73	68.51
THOC	88.14	90.99	89.54	79.76	90.95	84.99	88.45	90.97	89.69	92.06	89.34	90.68
AT	98.08	98.31	98.19	91.33	94.50	92.88	92.09	96.23	94.10	94.32	99.02	96.61
DCdetector	98.21	98.27	98.24	84.14	88.60	86.28	90.42	94.14	92.21	95.32	97.86	96.57
InterFusion	83.61	83.45	83.52	87.02	85.43	86.22	81.28	92.70	86.62	89.77	88.52	89.14
MEMTO	98.37	**99.04**	98.50	89.17	94.68	91.84	92.25	96.10	94.13	93.80	**99.41**	96.52
STEN	97.74	98.02	97.88	83.58	83.11	83.29	90.17	94.91	92.42	**96.49**	96.52	96.49
TFUL(Ours)	**98.92**	98.50	**98.69**	**93.88**	**95.26**	**92.98**	**92.77**	**97.77**	**95.98**	95.64	99.31	**97.16**

Note: The best results are bolded and the runner-ups are underlined.

anomalies, detecting the vast majority of anomaly events while maintaining an acceptable false positive rate. Anomalies in the MSL dataset typically manifest as coordinated changes in multiple sensor readings. Through time-frequency dual-domain feature fusion, TFUL can simultaneously capture morphological anomalies in the time domain and relational anomalies in the frequency domain, enabling comprehensive detection of complex collaborative anomalies.

d) On the SMAP dataset: TFUL achieved an F1-score of 97.16%, slightly outperforming other advanced methods. Although AT and MEMTO also performed well on this dataset, TFUL had a clear advantage in recall (99.31%), detecting almost all anomaly events. Considering the extremely high safety requirements of spacecraft monitoring, where missed detections could lead to catastrophic consequences, this high recall characteristic is of significant practical importance.

2) Comprehensive comparative analysis of traditional and advanced metrics: Table IV further demonstrates the comprehensive performance of TFUL compared with four representative advanced methods on five datasets, with evaluation metrics covering traditional F1-score, unadjusted F1-score (AF-F1), and advanced VUS metrics.

On the most challenging GECCO dataset, TFUL's performance significantly surpassed all compared methods. TFUL achieved an F1-score of 85.89%, 31.64 percentage points higher than the second-best performer, MEMTO (54.25%). This substantial gap fully demonstrates the powerful capability of our proposed time-frequency fusion framework in handling complex real-world industrial data. The GECCO dataset originates from an actual industrial water treatment system, featuring complex and diverse anomaly patterns with severe noise interference, posing extremely high demands on model robustness. By adaptively filtering noise through the entropy sparsification mechanism and learning the intrinsic patterns of data through complementary time-frequency dual-domain feature representations, TFUL excelled in this challenging scenario.

A deeper analysis of VUS metrics shows that TFUL achieved a VUS-PR value of 45.42% on the GECCO dataset, significantly higher than other methods (17.96% for MEMTO, 15.74% for STEN). The substantial lead in VUS-PR indicates that our model not only has advantages in point detection accuracy but also maintains superior performance in scenarios considering temporal delay tolerance. This characteristic is particularly important for practical deployment, as detection delays within a certain range are acceptable in actual monitoring systems; the key is not to completely miss anomaly events.

Regarding performance consistency across datasets, TFUL maintained high F1-scores across all five datasets (85.89%–98.69%), demonstrating good generalization capability. In contrast, the performance of other methods fluctuated more significantly across datasets. For example, while STEN achieved the highest VUS-ROC (98.30%) and VUS-PR (94.10%) on the SMAP dataset, its F1-score on the GECCO dataset was only 36.34%. This performance fluctuation suggests that these methods may have preferences for specific types of data distributions or anomaly patterns, whereas TFUL, through complementary learning in the time-frequency dual domains, acquired more robust feature representation capabilities.

From the perspective of temporal dimension evaluation, VUS-ROC and VUS-PR metrics provide insights not reflected by traditional point detection metrics. TFUL performed well on VUS metrics for most datasets, particularly leading significantly on the GECCO dataset. This indicates that our model can not only accurately detect anomaly points but also correctly identify the start and duration of anomaly events, which is crucial for anomaly diagnosis and root cause analysis.

D. Comparative Analysis of Metaheuristic Algorithm Optimization Effectiveness

Hyperparameter configuration has a decisive impact on the performance of deep learning models, but manual tuning is time-consuming, labor-intensive, and relies on expert experience. To validate the effectiveness of the proposed MsRwGWO algorithm in automatic hyperparameter optimization, we systematically compared it with eight mainstream metaheuristic algorithms under identical experimental settings. The optimization objectives included four key hyperparameters: learning rate, weight decay coefficient, loss balancing weight, and temperature parameter, which collectively influence the model's convergence speed, generalization capability, and detection accuracy.

Table V shows the performance of the TFUL model optimized by each algorithm on the NIPS_TS_Water dataset. In terms of comprehensive performance metrics, the model optimized by MsRwGWO achieved an adjusted F1-score of 84.80%, second only to the standard GWO optimization result

TABLE IV. PERFORMANCE COMPARISON OF TRADITIONAL AND ADVANCED METRICS

Method	Metrics	GECCO	PSM	SMD	MSL	SMAP
AT	F1	44.53	98.19	92.88	94.1	96.61
	AF-F1	70.37	65.9	**74.11**	67.54	67.31
	VUS-PR	10.14	92.48	72.53	84.83	92.18
	VUS-ROC	61.66	94.18	82.89	94.33	97.66
DCdetector	F1	37.08	98.24	86.28	92.21	96.57
	AF-F1	63.19	63.78	66.04	66.91	67.68
	VUS-PR	10.08	91.10	60.79	83.40	92.39
	VUS-ROC	60.19	90.54	78.63	94.45	97.32
MEMTO	F1	54.25	98.50	91.84	94.13	96.52
	AF-F1	16.21	66.46	70.71	67.27	66.72
	VUS-PR	17.96	94.00	72.58	85.93	92.17
	VUS-ROC	61.98	92.72	82.38	88.87	97.09
STEN	F1	36.34	97.88	83.29	92.42	96.49
	AF-F1	48.44	59.94	64.02	63.46	66.86
	VUS-PR	15.74	**94.70**	61.37	85.26	**94.10**
	VUS-ROC	**86.06**	**96.71**	**91.29**	**95.59**	**98.30**
TFUL(Ours)	F1	**85.89**	**98.69**	**92.98**	**95.98**	**97.16**
	AF-F1	**74.99**	**66.84**	67.84	**67.41**	**67.68**
	VUS-PR	**45.42**	93.41	**76.72**	**87.80**	93.73
	VUS-ROC	73.39	91.60	81.39	89.72	96.50

Note: The best results are bolded and the runner-ups are underlined.

TABLE V. PERFORMANCE COMPARISON OF DIFFERENT TFUL VARIANTS ON NIPS_TS_WATER DATASET

Method	PC	RC	F1	AF-PC	AF-RC	AF-F1	VUS-ROC	VUS-PR	AUC-PR	AUC-ROC	Thresh	TRT	TST
TFUL	0.7731	0.9610	0.8542	0.6144	0.9539	0.7472	0.7347	0.4585	0.8240	0.9769	0.2331	2.4872	1.9318
TFUL-GWO	0.8094	0.9715	0.8616	0.6214	0.9621	0.7542	0.7458	0.4695	0.8413	0.9921	0.1936	2.3566	1.6482
TFUL-SSA	0.7732	0.9516	0.8448	0.6187	0.9562	0.7497	0.7338	0.4470	0.8253	0.9756	0.3147	2.2680	1.7348
TFUL-GSA	0.7898	0.9851	0.8633	0.6175	0.9620	0.7501	0.7466	0.4648	0.8525	0.9890	0.2187	2.5350	1.9602
TFUL-PSO	0.7440	0.9386	0.8292	0.6118	0.9537	0.7447	0.7366	0.4331	0.7974	0.9797	0.2837	2.5898	2.1858
TFUL-GWO-SSA	0.7643	0.9370	0.8419	0.6356	0.9539	0.7609	0.7318	0.4417	0.8189	0.9663	0.2380	2.3790	2.0470
TFUL-ACA	0.7659	0.9800	0.8548	0.6232	0.9664	0.7556	0.7410	0.4666	0.8511	0.9890	0.2086	2.3716	1.7892
TFUL-GTG-ACA	0.7719	0.9569	0.8494	0.6262	0.9538	0.7550	0.7376	0.4553	0.8265	0.9759	0.2273	2.3798	1.7826
TFUL-MSRWGWO	0.7766	0.9713	0.8627	0.6291	0.9629	0.7594	0.7409	0.4691	0.8432	0.9839	0.2300	2.1112	1.4488

(85.14%) but significantly better than other algorithms. This result indicates that MsRwGWO has high efficiency in exploring the hyperparameter space, capable of finding configurations close to the optimum.

1) In terms of convergence efficiency: MsRwGWO showed clear advantages. Convergence curve analysis (as shown in Fig. 7) indicates that MsRwGWO achieved a validation loss reduction rate of 11.33% in the first 5 training epochs, higher than other compared algorithms (as shown in Fig. 8, Fig. 9). This rapid convergence characteristic is crucial for hyperparameter optimization because each hyperparameter configuration requires complete model training to evaluate its quality. Faster convergence means that more configurations can be evaluated with fewer training epochs, thereby exploring a broader hyperparameter space within the same time frame. The key to MsRwGWO's rapid convergence lies in its dynamic weighting mechanism—in the early optimization stages, the algorithm assigns higher weights to better individuals, accelerating movement toward promising regions; in later stages, it maintains population diversity through random perturbations to avoid premature convergence. Upon completion of the optimization, MsRwGWO identifies the optimal hyperparameter combination: a learning rate of 0.002, Weight Decay of 5×10^{-5}, Alpha Parameterand of 1.0, Temperature of 0.1, as listed in Table VI.

TABLE VI. THE MSRWGWO FOUND THE OPTIMAL PARAMETER CONFIGURATION

Parameter Name	Parameter Symbol	Optimal Value
Peak Learning Rate	η_{peak}	0.002
Weight Decay	λ	5×10^{-5}
Alpha Parameter	α	1.0
Temperature Parameter	τ	0.1

2) In terms of computational efficiency: the model optimized by MsRwGWO had relatively low average training time (2.11 seconds per epoch) and testing time (1.45 seconds), indicating that the introduced improvement mechanisms did not incur significant computational overhead. Compared to traditional grid search and random search, MsRwGWO, through its directed search strategy, finds high-quality solutions with fewer evaluations, significantly reducing the overall optimization time cost.

3) In terms of algorithm stability: MsRwGWO exhibited smaller performance fluctuations across multiple independent runs. For example, the standard deviation of the adjusted F1-score was 0.0147, smaller than most compared algorithms. This stability stems from MsRwGWO's multi-strategy boundary handling mechanism—when individuals exceed search boundaries, instead of simple truncation, they are guided toward the optimal individual direction through random contraction factors, maintaining population diversity while ensuring

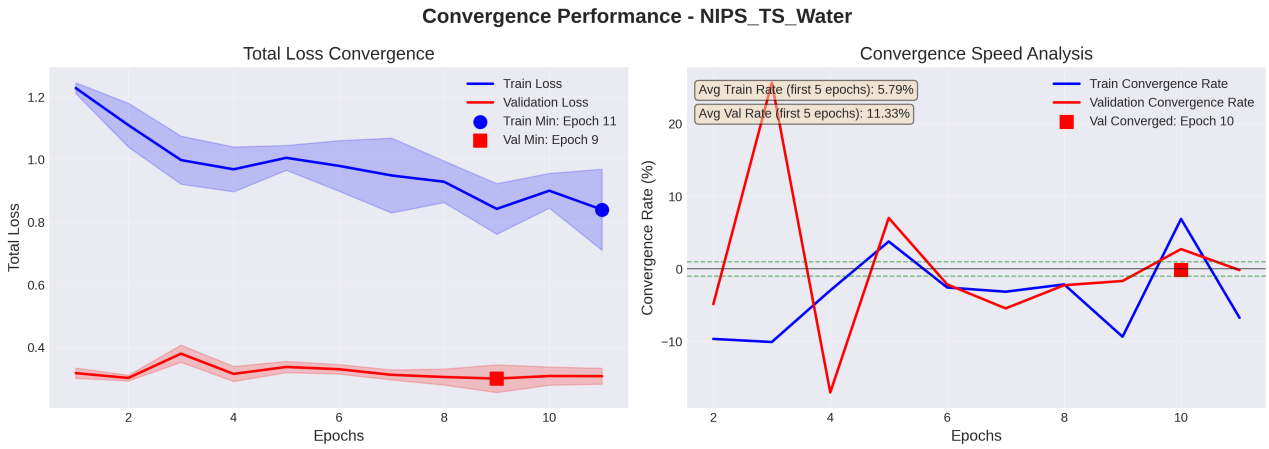


Fig. 7. MsRwGWO convergence.

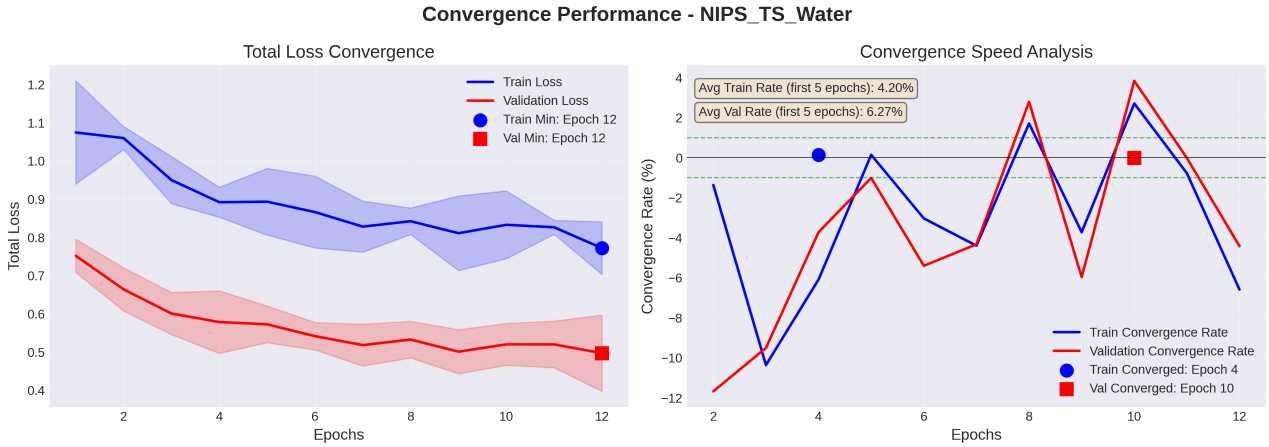


Fig. 8. GWO convergence.

the search remains within reasonable bounds.

4) In terms of exploration-exploitation balance: MsRwGWO achieved a good balance through an adaptive step size control mechanism. Early iterations use larger step sizes for global exploration, quickly locating promising regions; as iterations progress, step sizes gradually decrease for local fine-grained search. This dynamic adjustment avoids the potential issues of insufficient exploration or over-exploration that may arise from linear decay step sizes in standard GWO.

E. Ablation Study Analysis

To deeply understand the contribution of each component in the TFUL framework, particularly the necessity of the entropy sparsification mechanism in time and frequency domain processing, we designed and executed a systematic ablation study on the GECCO dataset. The GECCO dataset originates from a real industrial environment, featuring data complexity and strong noise interference, making it an ideal testbed for validating model component effectiveness. We constructed three variant models: TFUL-w/o-TimeSparse (removing time-domain entropy sparsification), TFUL-w/o-FreqSparse (removing frequency-domain sparsification, replacing learnable Mahalanobis distance with standard cosine similarity), and TFUL-w/o-AllSparse (removing all sparsification mechanisms simultaneously).

Table VII shows the performance comparison between the complete TFUL model and the three ablation variants. The experimental results clearly validate the necessity and synergistic effects of each component.

1) Analysis of the time-domain entropy sparsification mechanism: Removing time-domain entropy sparsification (TFUL-w/o-TimeSparse) caused a significant performance decline, with the adjusted F1-score dropping from 85.89% to 79.82%, a decrease of 6.07 percentage points. A deeper analysis of metric changes reveals an interesting phenomenon: recall decreased only slightly (98.24% $\rightarrow$ 97.78%), but precision dropped substantially (80.07% $\rightarrow$ 67.78%). This pattern indicates that without the guidance of entropy sparsification, the model tended to misclassify more normal samples as anomalies, leading to a significant increase in false positive rate. This validates the core role of our proposed time-domain attention entropy sparsification mechanism—by evaluating the information content of each time segment using information entropy, the model can adaptively focus on the most discriminative

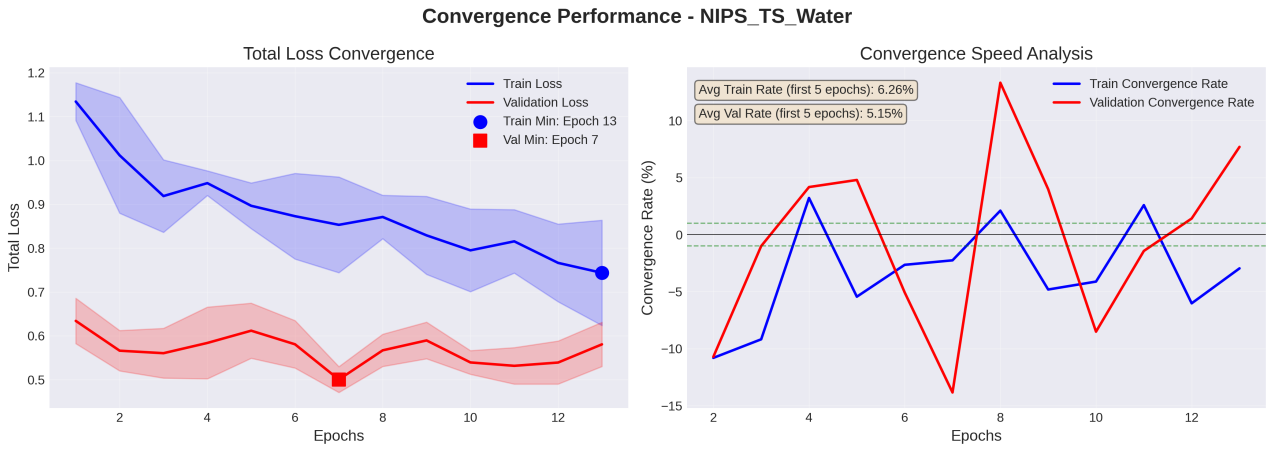


Fig. 9. SSA convergence.

TABLE VII. ABLATION ANALYSIS OF TFUL MODELS.

	Core Metrics			Traditional			Advanced		Other	
Method	**pc_adj**	**rc_adj**	**f1_adj**	**af_pc**	**af_rc**	**af_f1**	**vus_roc**	**vus_pr**	**auc_pr**	**auc_roc**
TFUL(Ours)	80.07	**98.24**	**85.89**	**62.08**	**96.29**	**74.99**	**74.34**	**45.42**	**82.0**	**98.58**
TFUL-w/o-TimeSparse	67.78	97.78	79.82	61.33	95.68	74.73	74.02	40.01	76.68	98.43
TFUL-w/o-FreqSparse	**80.29**	93.37	85.36	61.1	94.61	74.19	72.81	42.96	75.82	96.6
TFUL-w/o-AllSparse	72.67	92.93	80.08	60.08	95.04	73.53	72.69	39.76	77.9	96.84

local morphological patterns, ignoring segments dominated by redundancy or noise, thereby improving detection precision. From a computational efficiency perspective, time-domain sparsification also reduces the number of time segments that need to be processed, improving model efficiency while maintaining performance.

2) Analysis of the frequency-domain sparsification and learnable relationship modeling: Removing the frequency-domain sparsification mechanism (TFUL-w/o-FreqSparse) also caused performance degradation, with the F1-score dropping to 85.36%. Unlike the removal of time-domain sparsification, this variant exhibited the opposite pattern: precision slightly increased (80.29% vs. 80.07%), but recall significantly decreased (93.37% vs. 98.24%). This indicates that after replacing the learnable Mahalanobis distance with fixed cosine similarity, the model became overly conservative, increasing the miss rate. This result validates the value of the learnable Mahalanobis distance in frequency-domain relationship modeling—by learning nonlinear dependencies among sensors in a data-driven manner, the model can capture complex collaborative anomaly patterns that may not be effectively characterized by preset similarity measures (such as cosine similarity). Frequency-domain sparsification (implemented via Sparsemax) further enhances this process by retaining strong correlations and weakening irrelevant ones, enabling the model to focus on the most significant frequency-domain interaction patterns.

3) Analysis of the synergistic effect of dual-domain sparsification: Simultaneously removing all sparsification mechanisms (TFUL-w/o-AllSparse) caused further performance deterioration, with an F1-score of only 80.08%, the worst among all variants. This result strongly demonstrates the synergistic effect of time-domain and frequency-domain sparsification mechanisms. Time-domain sparsification enables the model to focus on the most informative local temporal patterns, while frequency-domain sparsification optimizes cross-sensor spectral relationship representations; together, they produce significant performance improvements. Notably, the complete model significantly outperformed all ablation variants in VUS-ROC (74.34%) and VUS-PR (45.42%) metrics, indicating that the sparsification mechanisms not only improve point detection accuracy but also enhance the model's detection consistency in the temporal dimension, making the start and end time localization of anomaly events more accurate.

4) Analysis from the perspective of advanced metrics: From the AUC-ROC and AUC-PR metrics, the complete model outperformed the ablation variants, particularly achieving 98.58% in AUC-ROC, approaching the performance of a perfect classifier. This further validates the effectiveness of the sparsification mechanisms in generating high-quality anomaly scores—by focusing on key features, the model learns more discriminative representations, resulting in more separable distributions of normal and anomalous samples.

F. Discussion of Practical Considerations and Limitations

While TFUL demonstrates state-of-the-art performance, several practical considerations and limitations merit discussion:

1) Concept drift: The current framework is trained offline on static data. In industrial environments, the statistical properties of "normal" behavior may evolve over time (concept drift).

TFUL does not incorporate online adaptation mechanisms; thus, its performance may degrade if deployed for extended periods without retraining. Extending TFUL to online learning with incremental model updates is a promising direction for future work.

2) Edge deployment and real-time latency: Our experiments were conducted on a high-end GPU, which is not representative of resource-constrained edge devices commonly used in industrial monitoring. The model's inference latency and memory footprint have not been optimized for such hardware. Future work could explore model quantization, pruning, or knowledge distillation to enable real-time edge deployment.

3) Sparsity threshold sensitivity: The entropy sparsification mechanism relies on a pre-defined sparsity rate ρ_d. In our experiments, we set $\rho_d = 0.3$ based on validation performance. However, an overly aggressive sparsity rate may discard subtle but informative patterns, potentially missing low-amplitude anomalies. A sensitivity analysis (varying ρ_d from 0.1 to 0.5) showed that F1-score varied by less than 2%, indicating moderate robustness, but optimal selection remains dataset-dependent. Learning ρ_d adaptively is an interesting avenue for improvement.

4) Error analysis and failure cases: Despite strong overall performance, TFUL occasionally misses anomalies that are extremely short-lived or masked by noise. For instance, on the GECCO dataset, some anomalies lasting only one time step were not detected, likely due to the smoothing effect of the multi-scale convolutions. A qualitative error analysis revealed that such missed detections often correspond to isolated spikes in a single sensor, which are difficult to distinguish from transient noise. Incorporating explicit spike-detection mechanisms could help.

5) Comparison with modern hyperparameter optimizers: We compared MsRwGWO against classic meta-heuristics; however, a comparison with modern Bayesian optimization tools like Optuna or Hyperband would provide additional context. We leave such a comparison to future work, noting that MsRwGWO's guided search may be particularly beneficial when evaluation budget is limited.

V. Conclusion

This study presented TFUL, a novel framework for time series anomaly detection that integrates entropy-sparsified time-frequency fusion with MsRwGWO meta-optimization. The key innovations include: 1) dual-domain entropy sparsification for adaptive feature selection, 2) heterogeneous feature extraction networks capturing complementary temporal and spectral patterns, and 3) an efficient meta-heuristic optimization algorithm for hyperparameter tuning.

Experimental results on five public datasets demonstrate that TFUL outperforms state-of-the-art methods in detection accuracy and robustness. The ablation studies confirm the importance of each component, particularly the sparsification mechanisms for reducing false positives and improving recall. The MsRwGWO algorithm provides efficient hyperparameter optimization, reducing tuning time while maintaining or improving performance.

Future work will focus on extending TFUL to online anomaly detection scenarios to handle concept drift, incorporating domain adaptation for cross-domain applications, exploring hardware-aware optimizations for edge deployment, and providing qualitative error analysis and case studies to further illuminate model behavior. Additionally, we plan to compare MsRwGWO with modern Bayesian optimization tools and investigate adaptive learning of sparsity thresholds.

Acknowledgment

This study is partially supported by the Natural Science Foundation of China, grant number 62562005, Gansu Provincial University Graduate Students "Innovation Star" Program, grant number 2026CXZX-973, Gansu Provincial Higher Education Teachers' Innovation Fund, grant number 2025A-124, Lanzhou Science and Technology Program, grant number 2023-1-53, and Anning District Science and Technology Program, grant number 024-JB-5.

References

[1] Wang, J., et al. "Improving bearing fault diagnosis method based on the fusion of time–frequency diagram and a novel vision transformer." *The Journal of Supercomputing* 81.1 (2025): 262.

[2] Bai, X., et al. "Open circuit fault diagnosis of wind power converter based on VMD energy entropy and time domain feature analysis." *Energy Science & Engineering* 12.3 (2024): 577-595.

[3] Ruff, L., et al. "Deep one-class classification." *International conference on machine learning.* PMLR, 2018.

[4] Seyyedabbasi, A., et al. "Optimal data transmission and pathfinding for WSN and decentralized IoT systems using I-GWO and Ex-GWO algorithms." *Alexandria Engineering Journal* 63 (2023): 339-357.

[5] Agarwal, P., and S. Kumar. "Electroencephalography based imagined alphabets classification using spatial and time-domain features." *International Journal of Imaging Systems and Technology* 32.1 (2022): 111-122.

[6] Yan, X., et al. "CDTFAFN: A novel coarse-to-fine dual-scale time-frequency attention fusion network for machinery vibro-acoustic fault diagnosis." *Information Fusion* 112 (2024): 102554.

[7] Tao, H., et al. "Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion." *Journal of the Franklin Institute* 360.2 (2023): 1454-1477.

[8] Ma, R., et al. "TFD-former: Time-frequency domain fusion decoders for effective and robust fault diagnosis under time-varying speeds." *Knowledge-Based Systems* 316 (2025): 113410.

[9] Zhao, W., and L. Fan. "Time-series representation learning via time-frequency fusion contrasting." *Frontiers in Artificial Intelligence* 7 (2024): 1414352.

[10] Xi, L., X. Meng, and H. Liu. "TFFC: time-frequency fusion consistency for semi-supervised time series classification." *Applied Intelligence* 55.11 (2025): 791.

[11] Zhao, S., X. Lu, and S. Zhao. "ScaleMixNet: Adaptive multi-scale time-frequency fusion with hybrid loss for time series forecasting." *The Journal of Supercomputing* 81.15 (2025): 1-28.

[12] Gong, C., and R. Peng. "A novel hierarchical vision transformer and wavelet time–frequency based on multi-source information fusion for intelligent fault diagnosis." *Sensors* 24.6 (2024): 1799.

[13] Wang, J., et al. "Improving bearing fault diagnosis method based on the fusion of time–frequency diagram and a novel vision transformer." *The Journal of Supercomputing* 81.1 (2025): 262.

[14] Zhang, X., et al. "Not all frequencies are created equal: Towards a dynamic fusion of frequencies in time-series forecasting." *Proceedings of the 32nd ACM International Conference on Multimedia.* 2024.

[15] Lei, T., J. Li, and K. Yang. "Time and frequency-domain feature fusion network for multivariate time series classification." *Expert Systems with Applications* 252 (2024): 124155.

[16] Zeng, S., et al. "Time–frequency fusion for enhancement of deep learning-based physical layer identification." *Ad Hoc Networks* 142 (2023): 103099.

[17] Yang, Y., Y. Zhang, and Q. Zeng. "Research on coal gangue recognition based on multi-layer time domain feature processing and recognition features cross-optimal fusion." *Measurement* 204 (2022): 112169.

[18] Islam, M. A., et al. "Time domain feature analysis for gas pipeline fault detection using LSTM." *International Journal of Science and Research Archive* [Internet] (2025): 1769-77.

[19] Bai, X., et al. "Open circuit fault diagnosis of wind power converter based on VMD energy entropy and time domain feature analysis." *Energy Science & Engineering* 12.3 (2024): 577-595.

[20] Agarwal, P., and S. Kumar. "Electroencephalography based imagined alphabets classification using spatial and time-domain features." *International Journal of Imaging Systems and Technology* 32.1 (2022): 111-122.

[21] Liu, F. T., K. M. Ting, and Z. Zhou. "Isolation forest." *2008 eighth ieee international conference on data mining.* IEEE, 2008.

[22] Zong, B., et al. "Deep autoencoding gaussian mixture model for unsupervised anomaly detection." *International conference on learning representations.* 2018.

[23] Xu, J., et al. "Anomaly transformer: Time series anomaly detection with association discrepancy." *arXiv preprint arXiv:2110.02642* (2021).

[24] Li, Z., et al. "Multivariate time series anomaly detection and interpretation using hierarchical inter-metric and temporal embedding." *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining.* 2021.

[25] Song, J., et al. "Memto: Memory-guided transformer for multivariate time series anomaly detection." *Advances in Neural Information Processing Systems* 36 (2023): 57947-57963.

[26] Chen, Y., et al. "Self-supervised spatial-temporal normality learning for time series anomaly detection." *Joint European Conference on Machine Learning and Knowledge Discovery in Databases.* Cham: Springer Nature Switzerland, 2024.

[27] Marini, F., and B. Walczak. "Particle swarm optimization (PSO). A tutorial." *Chemometrics and intelligent laboratory systems* 149 (2015): 153-165.

[28] Hatta, N. M., et al. "Recent studies on optimisation method of Grey Wolf Optimiser (GWO): a review (2014–2017)." *Artificial intelligence review* 52.4 (2019): 2651-2683.

[29] Liu, Y., et al. "Review of the grey wolf optimization algorithm: variants and applications." *Neural Computing and Applications* 36.6 (2024): 2713-2735.

[30] Xue, J., and B. Shen. "A novel swarm intelligence optimization approach: sparrow search algorithm." *Systems science & control engineering* 8.1 (2020): 22-34.

[31] Marini, F., and B. Walczak. "Particle swarm optimization (PSO). A tutorial." *Chemometrics and intelligent laboratory systems* 149 (2015): 153-165.

[32] Rashedi, E., H. Nezamabadi-Pour, and S. Saryazdi. "GSA: a gravitational search algorithm." *Information sciences* 179.13 (2009): 2232-2248.

[33] Blum, C. "Ant colony optimization: Introduction and recent trends." *Physics of Life reviews* 2.4 (2005): 353-373.

[34] Mohammed, H., Z. Abdul, and Z. Hamad. "Enhancement of GWO for solving numerical functions and engineering problems." *Neural Computing and Applications* 36.7 (2024): 3405-3413.

[35] Seyyedabbasi, A., et al. "Optimal data transmission and pathfinding for WSN and decentralized IoT systems using I-GWO and Ex-GWO algorithms." *Alexandria Engineering Journal* 63 (2023): 339-357.

[36] Abir, A. R., et al. "GTG-ACO: Graph Transformer Guided Ant Colony Optimization for learning heuristics and pheromone dynamics for combinatorial optimization." *Swarm and Evolutionary Computation* 99 (2025): 102147.

[37] Shen, Lifeng, Zhuocong Li, and James Kwok. "Timeseries anomaly detection using temporal hierarchical one-class network." *Advances in neural information processing systems* 33 (2020): 13016-13026.

[38] Yang, Yiyuan, et al. "Dcdetector: Dual attention contrastive representation learning for time series anomaly detection." *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*. 2023.

[39] Moritz, S., et al. "GECCO industrial challenge 2018 dataset." *Tech. Rep.* (2018).

[40] Abdulaal, Ahmed, Zhuanghua Liu, and Tomer Lancewicki. "Practical approach to asynchronous multivariate time series anomaly detection and localization." *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*. 2021.

[41] Su, Ya, et al. "Robust anomaly detection for multivariate time series through stochastic recurrent neural network." *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019.

[42] Hundman, Kyle, et al. "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding." *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 2018.

Blockchain-Based Multi-Chain Data Supervision Mechanism for Traditional Chinese Medicine Traceability System

Rongjun Chen[1], Yun Sun[2], Feng Xue[3], Yongzhi Ma[4],
Xinyu Wu[5], Xianxian Zeng[6], Jiawen Li[7]*, Jinchang Ren[8]*
School of Computer Science, Guangdong Polytechnic Normal University, Guangzhou, China[1,2,3,4,5,6,7,8]
School of Computing-Engineering and Technology, Robert Gordon University, Aberdeen, United Kingdom[8]

***Abstract*—Addressing the challenges of Traditional Chinese Medicine (TCM) traceability systems, including heavy data storage burdens, poor privacy protection, and susceptibility to tampering, this study establishes a highly secure and trustworthy traceability supervision system for the entire Chinese medicine supply chain, which enhances product quality and safety assurance. Centred on the Hyperledger Fabric consortium blockchain as its core architecture, a multi-chain integration framework comprising one regulatory main chain plus five organisational sub-chains is proposed to achieve permission control, data isolation, and privacy. A multi-mode encrypted data storage mechanism is designed, integrating China's national cryptographic algorithms SM4 and SM3 with CP-ABE attribute-based encryption to enable tiered management of private and non-private data. Zero-knowledge proof technology safeguards identity privacy during cross-chain data transmission, while QR codes and environmental data collection mechanisms enhance data entry efficiency and authenticity. The system achieves end-to-end traceability from cultivation and processing through transportation, warehousing, and sales. Comparative performance analysis shows that the proposed framework effectively alleviates data storage pressure, ensures data validity, enhances data security, and improves collaborative efficiency among organizations across the TCM supply chain. The proposed multi-chain integrated Chinese medicine traceability and supervision system enables efficient collaboration and trustworthy traceability across the entire Chinese medicine industry chain, while safeguarding data security and privacy, and has significant application and promotion value. Future integration with artificial intelligence and big data technologies could further enhance the system's intelligent analysis and decision-support capabilities.**

***Keywords*—Blockchain; traceability; multi-chain architecture; Hyperledger Fabric; Traditional Chinese Medicine**

I. Introduction

Several traditional Chinese herbal medicines have been demonstrated to modulate the growth, differentiation, invasion, and metastasis of Hepatocellular Carcinoma (HCC), which may be attributed to their capacity to regulate HCC progression [1]. Traditional Chinese Medicine (TCM) has considerable potential to address certain conditions that remain challenging for Western medicine. Nevertheless, current challenges in the TCM supply chain include unclear germplasm origins, genetic degradation of herbal species, pesticide residues, excessive levels of heavy metals, environmental contamination during harvesting and processing, and inadequate standardization in storage, transportation, and retail management. Therefore, it is necessary to establish a traceability system for traditional Chinese medicinal herbs that balances security and efficiency, enabling end-to-end tracking from cultivation to retail outlets, ensuring the quality of medicinal materials remains under control, and ensuring the safety of their use is traceable.

Existing blockchain-based traceability systems face significant limitations, including degraded performance as data volumes increase and insufficient data privacy, rendering them vulnerable to unauthorized tampering. For instance, Jamil et al. [2] pioneered the use of Hyperledger Fabric to record data in the pharmaceutical supply chain. Their system leveraged blockchain to log drug transactions, aiming to establish a smart healthcare ecosystem that addresses supply chain inefficiencies. While this approach capitalized on the decentralized nature of blockchain from a holistic supply chain perspective, it overlooked critical data-specific issues such as validity and privacy. Similarly, Agrawal et al. [3] proposed a blockchain-enabled network that enables manufacturers to monitor pharmaceutical products effectively throughout the supply chain, enhancing security and transparency. Although this solution improved data controllability, its oversight remained localized and failed to ensure comprehensive regulatory coverage across the entire supply chain.

To address these shortcomings in end-to-end traceability for Chinese herbal medicines, this study proposes a novel, high-security, multi-chain traceability and regulatory framework based on blockchain technology. The architecture consists of one regulatory main chain plus five organisational sub-chains, enabling multi-chain interaction and data fusion. Specifically, we adopt a permissioned consortium blockchain that incorporates the following key features: access control (only authorized participants can join and perform operations on the chain), consensus mechanisms (to guarantee node consistency), high performance and scalability (to meet industry-specific requirements), privacy preservation (to protect personal and commercial confidentiality of stakeholders), and customizability (to support future extensions) [4]. Furthermore, we implement a multi-modal data encryption and storage mechanism to enhance data security while reducing on-chain storage redundancy.

The structure of this study is as follows: Section II briefly introduces the core technology stack involved in this system. Section III presents the system design architecture and

*Corresponding authors.

highlights the core innovative components. Section IV details the system implementation. Section V conducts a comparative analysis of the system across different dimensions. Section VI concludes the study and outlines future directions.

II. Related Work

In recent years, with the frequent occurrence of food safety incidents and rising consumer demand for product transparency, traditional centralized database-based agricultural traceability systems have increasingly revealed critical limitations, including information silos, data tamperability, and a lack of trust. Blockchain technology, owing to its decentralization, immutability, and inherent traceability, has been widely adopted in supply chain traceability research across sectors such as agri-food, pharmaceuticals, and apparel. This study focuses on consortium blockchain-based system architectures. It provides a systematic review and critical analysis of recent representative studies, organized around three core dimensions: system architecture design, platform selection, and data storage with privacy preservation.

A. System Architecture Design: Consortium Blockchain

To overcome the performance, privacy, and controllability limitations of public blockchains, current research predominantly adopts consortium blockchain architectures to enable trusted data sharing among known participants. Hua et al. [5] proposed a three-role node model comprising a registration center, a data node, and a client that flexibly configures Byzantine Fault Tolerance or Paxos consensus algorithms in environments with identified stakeholders, supporting end-to-end traceability of agricultural products from cultivation to retail. Tseng et al. [6] leveraged the Gcoin consortium blockchain to assign unique digital identities to pharmaceuticals. They employed an anti-double-spending mechanism to detect duplicate circulation events, thereby aiding in the identification of counterfeit drugs. Sadri et al. [7] proposed a six-role permission model for blood supply chains, enforcing fine-grained access control to ensure operational compliance across the blood collection, testing, and transportation stages. Although Tsai et al. [8] addressed microfilm copyright protection rather than physical goods, their multi-chain parallel architecture, labeled ABC, TBC, MBC, demonstrated the scalability potential of consortium blockchains in high-concurrency scenarios. Collectively, these works illustrate that consortium blockchains, through participant restriction, enhanced permission management, and improved throughput, have become the prevailing architectural paradigm for high-assurance traceability systems.

B. Platform Selection: Hyperledger Fabric

Hyperledger Fabric has emerged as the preferred platform for traceability applications that require strong privacy and regulatory compliance, thanks to its modular design, channel architecture, pluggable consensus protocols, and fine-grained access control. Chen et al. [9] built a brand apparel anti-counterfeiting system on Fabric, utilizing ECDSA digital signatures and channel isolation to enable trustworthy data sharing across the entire supply chain from raw materials to retail, and to allow third-party arbitration entities to verify signatures and pinpoint falsification points hierarchically. Marchese et al. [10] introduced a dynamic rule engine in the agri-food domain, enabling regulators to inject specific compliance rules, such as temperature thresholds, into chaincode and automatically validate them upon batch registration, significantly enhancing regulatory compliance. Dong et al. [11] constructed a four-organization multi-channel Fabric network involving manufacturers, logistics providers, retailers, and regulators, integrating RFID tags with Golang-based chaincode to record and query food lifecycle data while ensuring enterprise data isolation. These studies consistently exploit Fabric's channel mechanism to achieve natural data partitioning: only authorized entities such as regulators can access cross-channel information, effectively balancing transparency with commercial privacy.

C. Data Storage and Privacy Protection: On-Chain Off-Chain Hybrid Architecture

Given the high storage costs and limited scalability of on-chain data, virtually all Fabric-based studies adopt a hybrid architecture in which only cryptographic hashes are stored on chain, while the original data resides off-chain. Chen et al. [9], Marchese et al. [10], and Dong et al. [11] all store primary business data, such as inspection reports, images, and contracts, in off-chain databases or on IPFS, recording only their hashes in the Fabric ledger. This approach preserves data integrity while mitigating blockchain bloat. Sensitive information, such as transaction prices and supplier identities, is encrypted using AES or ECC before being recorded on the chain, and Fabric's attribute-based access control enforces fine-grained permissions. Hua et al. [5] further recommended tiered authorization for critical operation logs, restricting data access and verification to designated roles. However, most studies inadequately address the availability of off-chain data. If an off-chain storage node fails, consumers may verify hash consistency but cannot retrieve the original content, which poses a risk of a null pointer exception.

Despite these advances in architectural design and privacy protection, several limitations remain. First, most existing Fabric-based systems rely on single-chain or multi-channel architectures and are therefore prone to storage bottlenecks and performance degradation as data volumes grow. Second, current hybrid storage schemes generally lack robust data-availability guarantees and fine-grained security control mechanisms. If an off-chain storage node fails or is maliciously tampered with, the original data may become unrecoverable, and traditional encryption approaches often incur risks of plaintext key storage on-chain or isolated key management off-chain. Third, cross-chain data interoperability and end-to-end regulatory capabilities remain insufficient, making it difficult to achieve truly comprehensive, end-to-end traceability. In addition, traditional non-blockchain traceability systems depend on centralized databases, which are inherently susceptible to data tampering, information silos, and the absence of a credible trust mechanism.

III. System Design

The proposed traceability and regulatory framework is built upon three key mechanisms. First, a multi-chain, integrated data supervision and transmission mechanism isolates organizational subchains, enabling end-to-end oversight while preserving the privacy of user accounts. Second, a multimodal

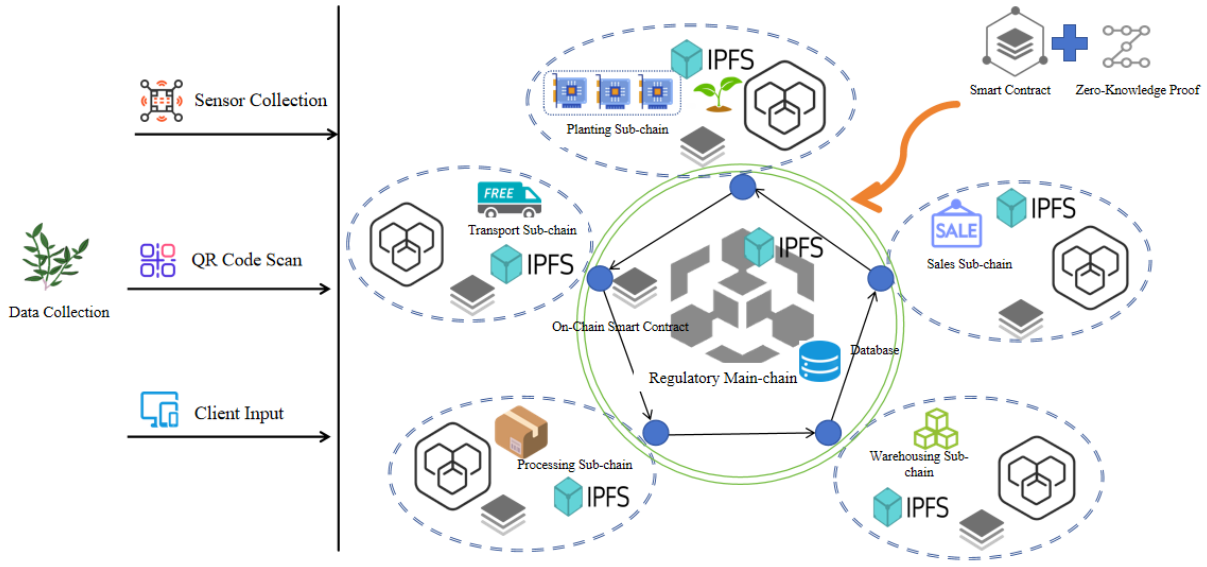


Fig. 1. Architecture diagram of a high-safety TCM traceability supervision system based on multi-chain fusion.

encrypted storage mechanism mitigates on-chain storage pressure, ensures strong data security, and supports fine-grained access control. Third, a QR code-based data entry mechanism, coupled with environmental data acquisition, enhances operational efficiency, safeguards user privacy, and ensures the validity of environmental data. Together, these mechanisms significantly improve inter-organizational collaboration and ultimately realize a highly secure, end-to-end traceability solution for the entire TCM supply chain. The overall architecture is illustrated in Fig. 1.

Five specialized sub-chains are defined: cultivation, transportation, processing, warehousing, and retail. The cultivation sub-chain, as the starting point of the supply chain, is responsible for recording detailed information on the cultivation of TCM materials. The transportation sub-chain records the movement of medicinal materials from planting sites to processing facilities, including transportation time, selected modes of transportation, and related logistics information. The processing sub-chain captures the processing of TCM materials, including the selection of processing methods, processing duration, and key process parameters. The warehousing sub-chain maintains information on the storage of medicinal materials in warehouses, including temperature and humidity conditions, storage duration, and inventory levels. The retail sub-chain records sales information, including the diversity of sales channels, the reasonableness of sales timing, and the stability of sales prices. Each organizational sub-chain thus records the key operational data of its own stage, and together the five sub-chains ensure that consumers can verify end-to-end information for each batch of Chinese medicinal products from growers to retailers.

A. Multi-Chain Integrated Data Supervision and Transmission Mechanism

As shown in Fig. 2, the system establishes a multi-chain integrated data supervision and transmission mechanism centered on the supervisory mainchain as the core hub, which connects downward to five operational subchains: cultivation, processing, transportation, packaging, and sales. All dynamic data must first undergo identity verification through a zero-knowledge proof-based authentication smart contract deployed on the supervisory main chain [12]. An operation permit is granted only upon successful verification. Subsequently, the main chain routes the relevant instruction, along with a hash digest, to the appropriate target sub-chain based on the business type, thereby implementing a closed-loop "verify before transmit" regulatory process. Sensor stream data and QR code scan records, classified as static data, are generated within each sub-chain. Since their authenticity has already been validated locally, they can be directly recorded in the respective subchain ledger. However, after block finalization, each sub-chain must periodically submit its block hash to the supervisory main chain to support global auditing. This mechanism effectively isolates direct interaction between sub-chains and end-user clients while ensuring end-to-end data trustworthiness, traceability, and privacy through centralized orchestration and backup coordination by the main chain.

B. Multi-Modal Encrypted Data Storage Mechanism

Building upon the aforementioned multi-chain organizational structure, this system addresses data security through a hybrid on-chain-off-chain multi-modal encryption and storage strategy. Data are classified according to sensitivity into privacy-sensitive and non-privacy-sensitive data, each assigned a tailored storage approach. As shown in Fig. 3, for privacy-sensitive data, the system employs the Chinese national cryptographic standard SM4 (formerly SMS4.0), a block cipher algorithm that offers high security and superior performance [13], to enhance both transmission efficiency and confidentiality. To protect the SM4 symmetric keys on the chain, the system adopts Ciphertext Policy Attribute-Based Encryption (CPABE) [14], in which user attributes, such as role or orga-

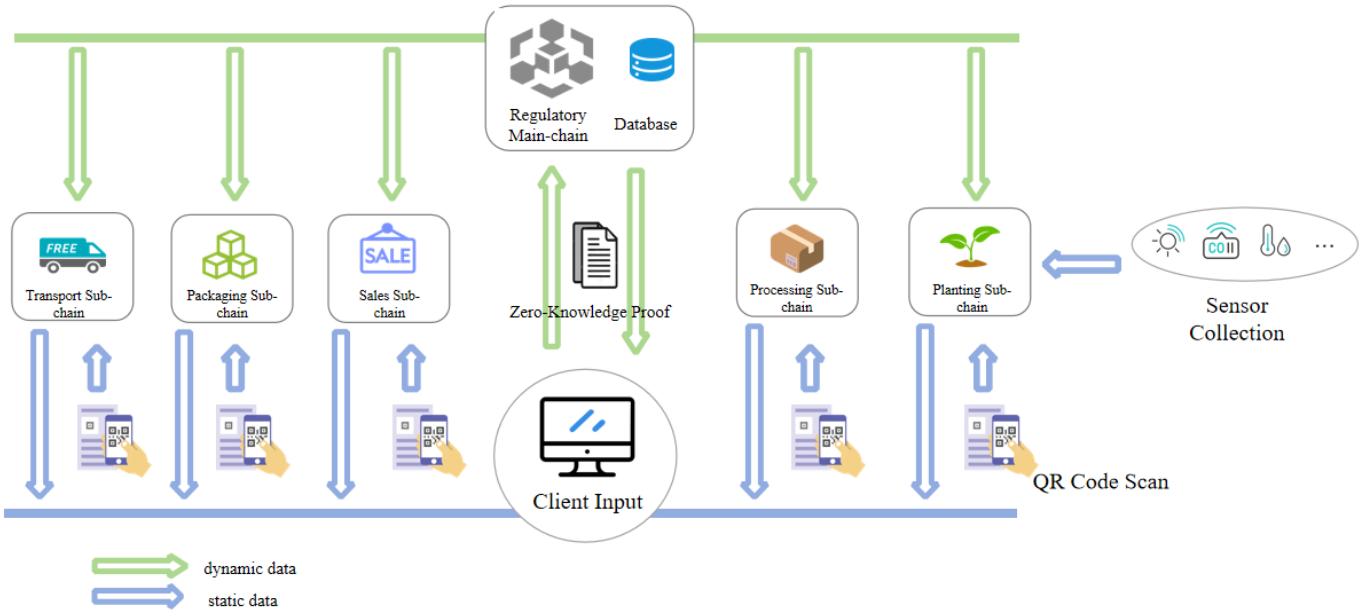


Fig. 2. A schematic diagram of the data regulatory transmission mechanism based on multi-chain fusion.

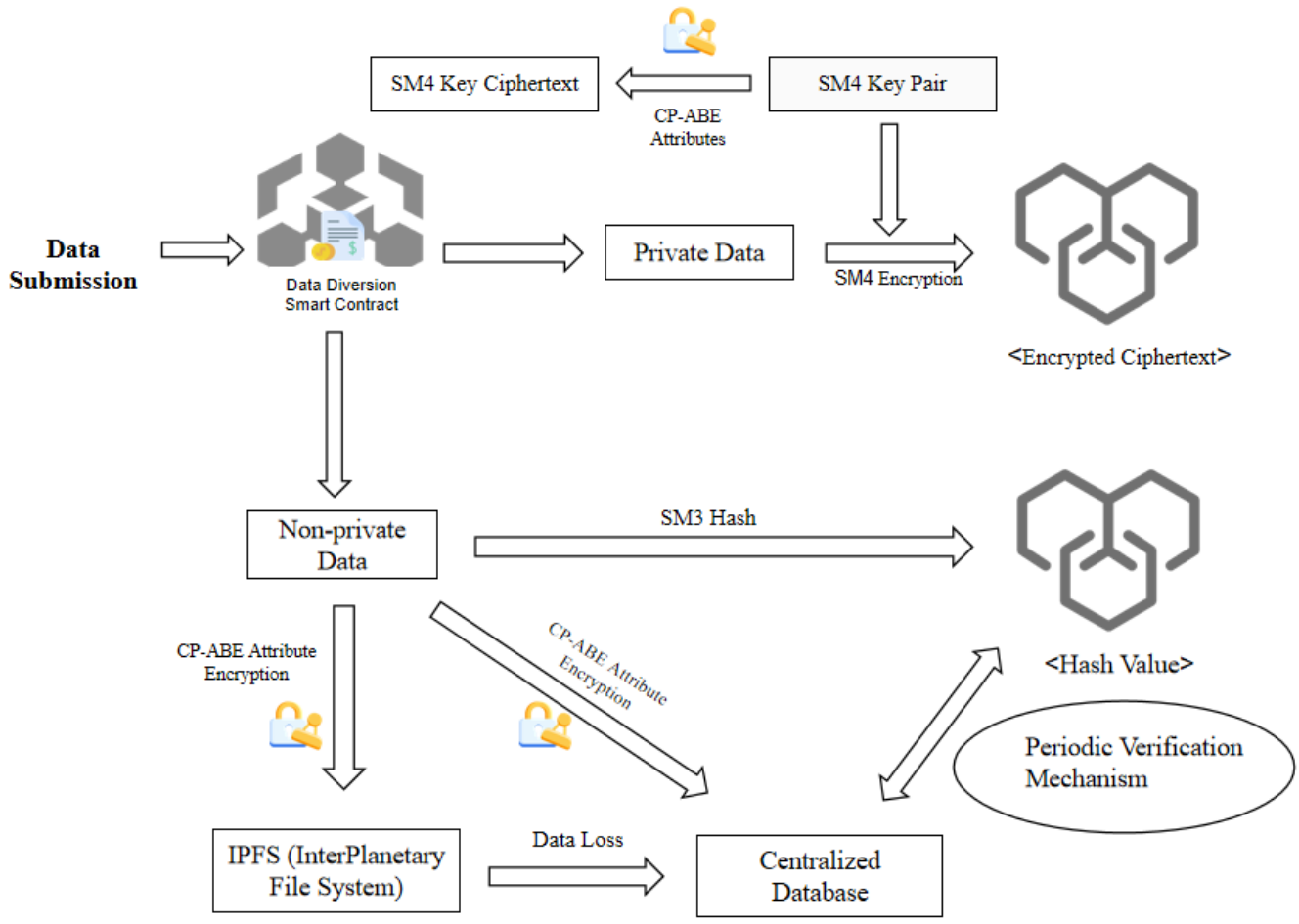


Fig. 3. Schematic diagram of data multi-mode encrypted storage.

nizational affiliation, serve as access policies. Compared with conventional symmetric or asymmetric encryption schemes, CPABE eliminates the risks associated with storing keys in plaintext on-chain or managing them in isolated off-chain repositories. Only users whose attributes satisfy the defined policy can decrypt and retrieve the SM4 key, thereby enabling fine-grained access control. For non-privacy-sensitive data, the system stores the original content in a centralized database to optimize read and write performance. Concurrently, it applies the SM3 cryptographic hash algorithm, a highly efficient national standard, to generate a digest of the data. This digest is then recorded on the corresponding sub-chain. A periodic verification mechanism between the sub-chain and the centralized database ensures the integrity and validity of the off-chain data. To further strengthen resilience against data loss or tampering, the system leverages the Inter Planetary File System (IPFS) [15] as a decentralized backup layer. If the periodic verification detects corruption, deletion, or unauthorized modification in the centralized database, the system automatically restores the affected data from the IPFS backup, ensuring continuous and reliable operation. Although the data shared within the system is accessible internally, external access remains strictly controlled. Therefore, all externally exposed data is encrypted using CPABE with a system-wide root Certificate Authority (CA) as the governing attribute policy. This configuration allows all authorized internal nodes to decrypt while denying access to external entities, thereby maintaining a robust security boundary between the internal consortium and the outside environment.

C. QR Code Generation and Environmental Data Acquisition Mechanism

In practical production scenarios, sub-chain personnel are frequently required to perform traceability-related operations, such as farmers applying fertilizer or irrigating. Conventional approaches, such as manual logbooks or direct client-side data entry, pose significant risks of information leakage or falsification. To address these vulnerabilities, the proposed design leverages the physical isolation property of QR codes. Specifically, a sub-chain user initiates a request via a dedicated mini program to the supervisory main chain, which then invokes a smart contract to generate a time-bound procedural QR code. This approach enhances both the security of traceability records and operational efficiency, as workers need only present the QR code to a scanning device. The scanner automatically captures the QR code content, appends a current timestamp and other contextual metadata, and uploads the complete record to the corresponding subchain for storage. Subsequently, a data-routing smart contract classifies and encrypts incoming data by type before storing it in the appropriate location. This mechanism assumes that all personnel affiliated with the organizational subchains have been pre-registered, with their relevant identity and role information securely collected and transmitted in advance to the supervisory main chain for encrypted storage as privacy-sensitive data. Upon completion of product packaging, the supervisory main chain receives a notification triggering the traceability query workflow. It then invokes a cross-chain traceability smart contract to aggregate traceability records associated with the product. Concretely, using the unique product traceability ID, the system queries and consolidates relevant data from cultivation, processing, transportation, and other pertinent subchains. The integrated traceability dataset is finally encoded into a QR code, known as the traceability information QR code, enabling end users to access a complete, verifiable history of the Traditional Chinese Medicine product with a simple scan.

IV. System Implementation

The chaincode implements the ChaincodeStubInterface to define multiple custom application functions, as listed in Table I. The planting sub-chain is specifically designed to monitor the seed growth environment, ensuring the validity of data submitted to the blockchain. The supervisory mainchain integrates information from all subchains to enable unified oversight and management. By continuously monitoring and analyzing data from all stages of the supply chain, the supervisory mainchain can promptly identify potential quality issues and safety hazards and initiate appropriate corrective actions. All operations executed by the chaincode generate transaction records stored on the blockchain, and any resulting state changes are updated in the state database.

Based on a multi-chain integrated architecture comprising one regulatory main chain and five organizational sub-chains, the system implements a unified identity verification process for all organizational logins. When a user accesses the login interface, the backend identifies the user's organizational affiliation based on their account credentials and directs them to the corresponding organization's main interface. This approach enhances security among organizations by requiring all inter-organizational data exchanges to request permission from the supervisory mainchain. Upon logging in to their respective main interfaces, users can clearly view detailed data for their assigned chain. Within the supervisory chain, every data operation, including creation, modification, and deletion, is meticulously recorded, enabling full traceability and auditability. This comprehensive logging mechanism empowers regulators to conduct effective, holistic oversight, enabling organizations to rapidly pinpoint the source of data anomalies and apply timely corrections. Users of organizational subchains are restricted to performing Create, Read, Update, and Delete (CRUD) operations exclusively on their own sub-chain data. Nevertheless, all such operations must generate corresponding log entries stored on the supervisory mainchain to support potential future traceability requirements. As an illustrative example, the traceability query functionality is displayed in Fig. 4.

V. Comparative Analysis

A. Storage Performance Analysis

The provenance supervision framework based on multi-chain integration not only ensures the security of private data through multi-modal encryption but also alleviates the storage burden inherent in conventional single-chain architectures by distributing data across multiple chains. Specifically, the supervision mainchain is dedicated solely to oversight and stores only a minimal amount of data, while each organizational subchain handles the storage requirements of its respective organization.

TABLE I. CHAIN CODE APPLICATION INTERFACE METHOD

	Method	Parameter	Description
Total Interface Methods of "Five Organizational Subchains + One Regulatory Mainchain"Data Supervision Scope	initLedger	ChaincodeStubInterface	Chaincode Initialization Function
	Invoke	ChaincodeStubInterface	Receive and process requests sent by clients
	createTrace	ChaincodeStubInterface[]string	Input a set of data arrays to add new transaction data records
	updateTrace	ChaincodeStubInterface[]string	Input a set of data arrays to update transaction data records
	deleteTrace	ChaincodeStubInterface[]string	Delete transaction data records
	queryTrace	ChaincodeStubInterface args []string	Input the corresponding traceability ID to query the latest data records
	queryAllTrace	ChaincodeStubInterface args []string	Input the corresponding traceability ID to query the latest data history records
	storeTraceMessage	ChaincodeStubInterface args []string	Multi-mode encrypted storage of data
Interface Methods Unique to the Planting Subchain	envrDataProcess	ChaincodeStubInterface args []string	Detect the similarity of environmental data
Interface Methods Unique to the Regulatory Mainchain	authentication	ChaincodeStubInterface args []string	Authenticate the identity of system login users
	qrCodeGeneration	ChaincodeStubInterface args []string	Generate QR code images for the corresponding data
	dataValidation	ChaincodeStubInterface args []string	Verify the validity of forwarded data
	traceQuery	ChaincodeStubInterface args []string	Integrate traceability information for user queries

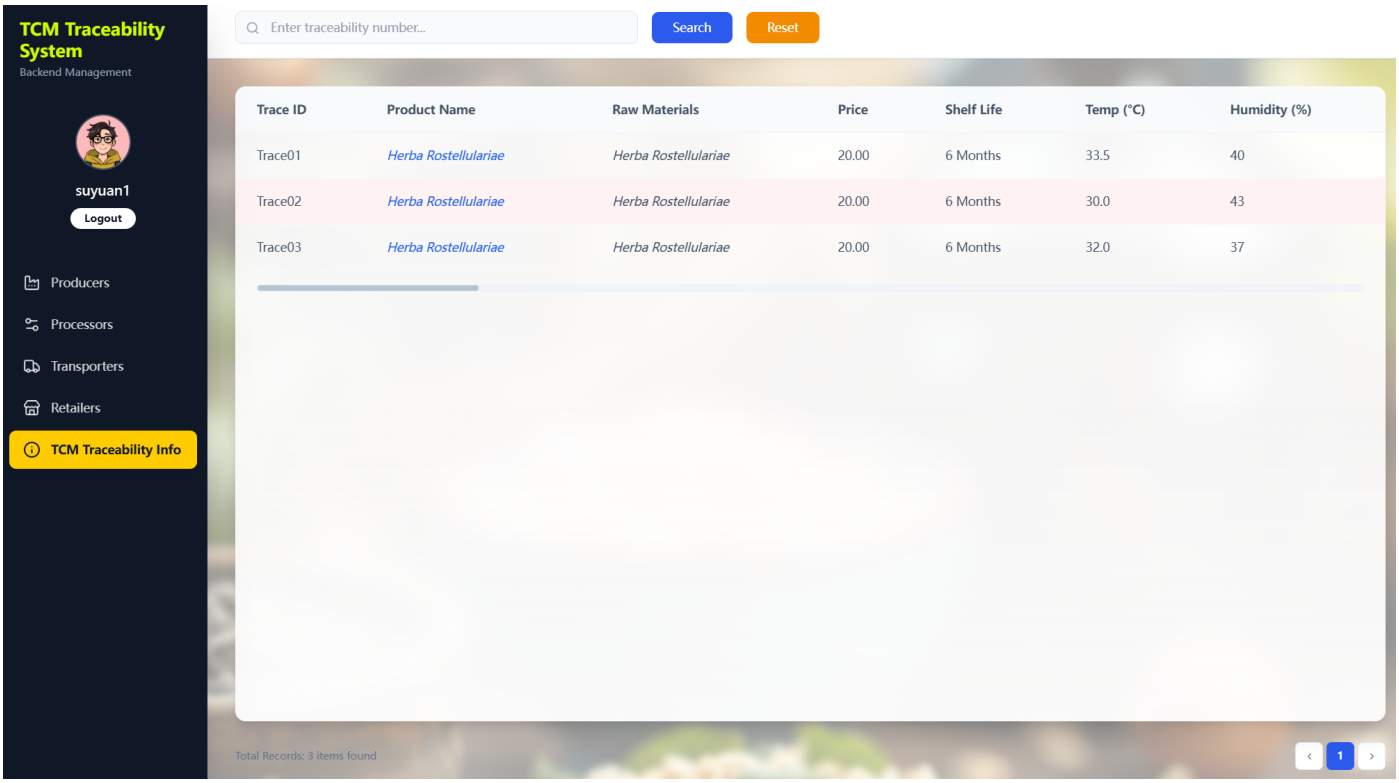


Fig. 4. Traceability information query function demonstration.

In this study, we evaluate the storage consumption of three configurations: organizational subchains (averaged across five subchains), a traditional single-chain system, and the supervision mainchain, as the number of uploaded transactions increases from 1×10^4 to 5×10^4. The results are illustrated in Fig. 5. It should be noted that, although real-world blockchain storage may involve adjustments to parameters such as block size and block generation interval based on transaction volume, this analysis focuses exclusively on the raw storage footprint attributable to the data volume itself, holding all other configuration parameters constant.

B. Comparative Evaluation of Approaches

Building upon the preceding analysis of security and storage performance, this study compares the proposed multi-chain-integrated traceability and supervision framework for Traditional Chinese Medicine with five representative existing traceability and supply chain information management systems, including a blockchain–IPFS-based monitoring system for non-perishable agricultural products that stores encrypted

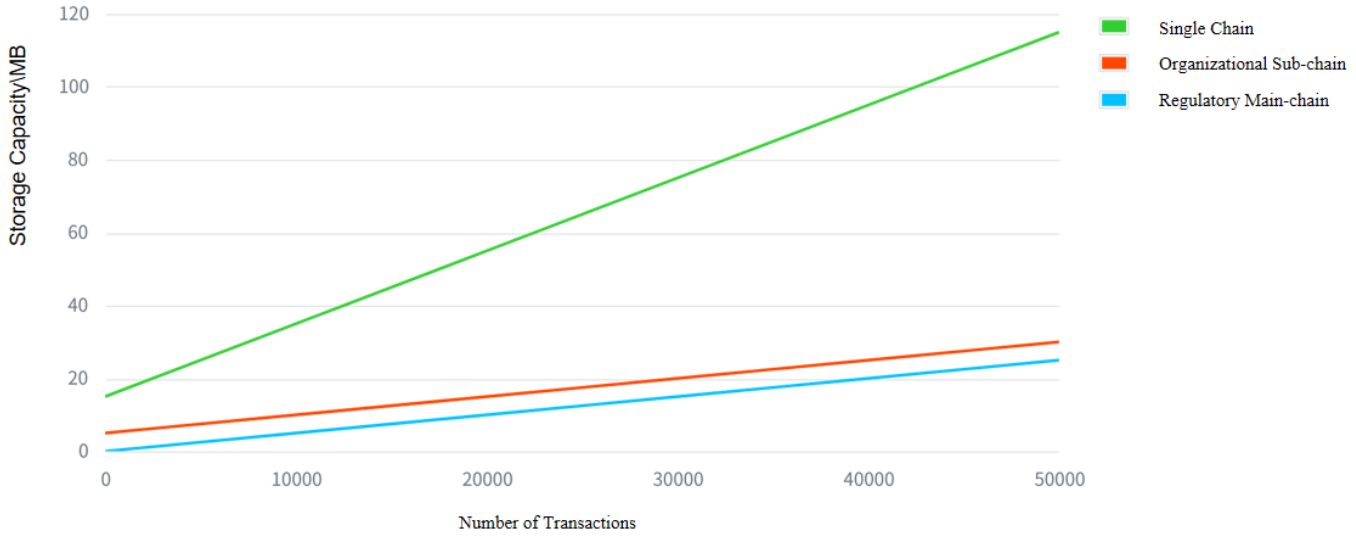


Fig. 5. Storage performance test.

TABLE II. SCHEME COMPARISON AND ANALYSIS

Research Works	[16]	[17]	[18]	[19]	[20]	**This system**
Blockchain Architecture	Single chain	Single chain	Single chain	Single chain	Single chain	Multi-chain
Data Supervision Scope	Partial info	Partial info	Partial info	Partial info	Partial info	Full lifecycle
Privacy Protection	Yes	Yes	Yes	Yes	Yes	Yes
Data Storage Performance	Medium	High	High	Medium	High	High
Data Security	High	Medium	High	High	High	Even higher
Data Validity	Medium	Medium	Medium	Medium	Medium	High
Data Access Control	None	None	None	None	None	Yes
Enterprise Collaboration	Medium	Low	Low	Low	Low	High
Main-chain Supervision Strength	Low	Low	Low	Low	Low	High
Enterprise Responsibility Differentiation	Yes	None	None	None	None	Yes

IPFS hashes on chain to alleviate centralization, reliability, and query-efficiency issues in traditional traceability architectures [16]; a blockchain-based wine supply chain traceability system that records each transaction from grape growers to retailers on chain to provide tamper-resistant, end-to-end provenance verification [17]; a Polygon-based cannabis supply chain solution that combines smart contracts with hybrid on-chain/off-chain storage to support efficient information retrieval and tamper-proof regulatory compliance across multiple stakeholders [18]; a blockchain-enabled vaccination record tracking framework that maintains immutable, privacy-preserving immunization histories across healthcare institutions [19]; and a blockchain- and IoT-driven CO_2 footprint tracking framework that enhances data integrity, transparency, and sustainability assessment in complex supply chains [20]. The comparison results are summarized in Table II. Overall, the proposed system demonstrates clear advantages across multiple dimensions, including secure cross-chain data interaction, scope of data supervision, storage efficiency, traceability and regulatory effectiveness, enterprise-level data management capabilities, strength of mainchain-based oversight, and differentiation of enterprise accountability.

VI. CONCLUSION AND FUTURE WORK

Based on existing research on traceability in TCM, this study addresses key challenges in current blockchain-based traceability systems, including single-chain storage bottlenecks, data security, data availability, and system stability. It also considers the entire TCM supply chain, encompassing cultivation, processing, transportation, sales, and regulatory oversight. To this end, we propose and implement a high-security TCM traceability and supervision system based on Hyperledger Fabric as the consortium blockchain platform. The primary contributions and innovations of this work are summarized in the following three aspects:

First, to address the storage overhead, data security, and availability challenges specific to TCM traceability scenarios, we propose a multi-chain architecture built on Hyperledger Fabric, comprising a supervising mainchain and five organiza-

tional subchains. This multi-chain framework enables secure cross-chain interaction and data fusion, effectively alleviating storage pressure, enhancing data validity, ensuring high data security, and improving inter-organizational collaboration efficiency. Furthermore, we design a multi-modal encrypted storage mechanism to strengthen data confidentiality and support fine-grained access control.

Second, we develop a zero-knowledge proof-based data supervision and transmission protocol that isolates organizational subchains while enabling end-to-end regulatory oversight without compromising user account privacy. Complementing this, we integrate QR code-based data entry and environmental data collection mechanisms, which not only improve operational efficiency but also safeguard user privacy and ensure the authenticity and reliability of environmental monitoring data. Collectively, these mechanisms guarantee the validity, traceability, and high security of data throughout the entire TCM lifecycle, from cultivation and harvesting to processing, storage, transportation, and retail. System performance evaluations confirm that the proposed framework significantly reduces storage burden, enhances data availability and security, and boosts collaborative efficiency across stakeholders in the TCM supply chain.

Third, we have constructed and successfully deployed a comprehensive TCM traceability management system that integrates the proposed multi-chain supervision architecture. The system consists of a Hyperledger Fabric consortium blockchain network, a traceability information management platform, environmental sensing devices, and QR code input terminals. We detail the functional modules, smart contract interfaces, and system demonstrations to ensure high reliability and operational efficiency. Comprehensive system testing, including storage capacity evaluation, was conducted across all stages to verify the effectiveness of each functional module in data recording and traceability. The results demonstrate that the system meets the design requirements for secure traceability management throughout the entire TCM supply chain.

For future work, we plan to incorporate artificial intelligence and big data analytics to further advance this research. By leveraging AI-driven data mining and big data analysis techniques [21–25], we aim to extract actionable insights from the vast volumes of data generated across the TCM supply chain, thereby providing scientific, intelligent, and data-driven decision support for the optimization and management of TCM production and distribution.

ACKNOWLEDGMENT

This study was supported by the Guangdong Basic and Applied Basic Research Foundation (No. 2024A1515010219), the Guangzhou Science and Technology Plan (No. 2024B03J1361, No. 2023B03J1327, and No. 2023A04J0362), the Guangdong Province Ordinary Colleges and Universities Young Innovative Talents Project (No. 2022KQNCX038), the Key Discipline Improvement Project of Guangdong Province (No. 2025ZDJS023, No. 2022ZDJS015, and No. 2021ZDJS025), the Scientific Research Capacity Improvement Project of the Doctoral Program Construction Unit of Guangdong Polytechnic Normal University (No. 22GPNUZDJS17), the Graduate Education Demonstration Base Project of Guangdong Polytechnic Normal University (No. 2023YJSY04002), and the Open Research Fund of the Guangdong Provincial Key Laboratory of Big Data Computing (No. B10120210117-OF08).

REFERENCES

[1] G. Guo, J. Zhou, X. Yang, J. Feng, Y. Shao, T. Jia, Q. Huang, Y. Li, Y. Zhong, P. Nagarkatti, and M. Nagarkatti, "Role of microRNAs induced by Chinese herbal medicines against hepatocellular carcinoma: A brief review," *Integrative Cancer Therapies*, vol. 17, no. 4, pp. 1059-1067, 2018.

[2] F. Jamil, L. Hang, K. H. Kim, and D. Kim, "A novel medical blockchain model for drug supply chain integrity management in a smart hospital," *Electronics*, vol. 8, no. 5, p. 505, 2019.

[3] D. Agrawal, S. Minocha, S. Namasudra, and A. H. Gandomi, "A robust drug recall supply chain management system using Hyperledger blockchain ecosystem," *Computers in Biology and Medicine*, vol. 140, p. 105100, 2022.

[4] O. Dib, K. L. Brousmiche, A. Durand, E. Thea, and E. B. Hamida, "Consortium blockchains: Overview, applications and challenges," *International Journal on Advances in Telecommunications*, vol. 11, no. 1, pp. 51-64, 2018.

[5] J. Hua, X. Wang, M. Kang, H. Wang, and F.-Y. Wang, "Blockchain based provenance for agricultural products: A distributed platform with duplicated and shared bookkeeping," in *Proc. 2018 IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 97-101.

[6] J.-H. Tseng, Y.-C. Liao, B. Chong, and S.-W. Liao, "Governance on the drug supply chain via Gcoin blockchain," *International Journal of Environmental Research and Public Health*, vol. 15, no. 6, p. 1055, 2018.

[7] S. Sadri, A. Shahzad, and K. Zhang, "Blockchain traceability in healthcare: blood donation supply chain," in *Proc. 2021 23rd International Conference on Advanced Communication Technology (ICACT)*, 2021, pp. 119-126.

[8] W.-T. Tsai, L. Feng, H. Zhang, Y. You, L. Wang, and Y. Zhong, "Intellectual-property blockchain-based protection model for microfilms," in *Proc. 2017 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, 2017, pp. 174-178.

[9] C.-L. Chen, X. Shang, W.-J. Tsaur, W. Weng, Y.-Y. Deng, C.-M. Wu, and J. Cui, "An anti-counterfeit and traceable management system for brand clothing with Hyperledger Fabric framework," *Symmetry*, vol. 13, no. 11, p. 2048, 2021.

[10] A. Marchese and O. Tomarchio, "An agri-food supply chain traceability management system based on Hyperledger Fabric blockchain," in *Proc. 23rd International Conference on Enterprise Information Systems (ICEIS)*, 2021, pp. 648-658.

[11] Z. Dong, C. Ma, Y. Wang, and Z. Liu, "Food information traceability system based on Fabric," in *Proc. 2020 International Conference on Aviation Safety and Information Technology (ICASIT)*, 2020, pp. 563-570.

[12] P.-W. Chi, Y.-H. Lu, and A. Guan, "A privacy-preserving zero-knowledge proof for blockchain," *IEEE Access*, vol. 11, pp. 85108-85117, 2023.

[13] J. Liu, Q. Guo, and Y. Tian, "The design of SM4 algorithm based on FPGA and its teaching application," in *Proc. 3rd International Conference on Artificial Intelligence and Education (ICAIE)*, 2025, pp. 374-379.

[14] F. Guo, Y. Mu, W. Susilo, D. S. Wong, and V. Varadharajan, "CP-ABE with constant-size keys for lightweight devices," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 5, pp. 763-771, 2014.

[15] M. Steichen, B. Fiz, R. Norvill, W. Shbair, and R. State, "Blockchain-based, decentralized access control for IPFS," in *Proc. 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, 2018, pp. 1499-1506.

[16] K. Biswas, V. Muthukumarasamy, and W. L. Tan, "Blockchain based wine supply chain traceability system," in *Proc. 2017 Future Technologies Conference (FTC)*, 2017, pp. 56-62.

[17] S. Babu and H. Devarajan, "Agro-food supply chain traceability using blockchain and IPFS," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 1, pp. 393-399, 2023.

[18] P. Nowvaratkoolchai, N. Thawesaengskulthai, W. Viriyasitavat, and P. Rangsunvigit, "Blockchain-based cannabis traceability in supply chain management," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 2, pp. 75-85, 2024.

[19] G. K. Shwetha, J. A. Rathod, G. Naveen, M. Arkachari, and M. K. Pushparani, "Blockchain-based vaccination record tracking system," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, pp. 753-759, 2024.

[20] M. Y. Mofatteh, R. Davallou, C. N. Ishimwe, S. S. Divekar, and O. F. Valilai, "Developing a blockchain based supply chain CO2 footprint tracking framework enabled by IoT," *nternational Journal of Advanced Computer Science and Applications*, vol. 15, no. 10, pp. 7-12, 2024.

[21] R. Chen, H. Huang, Y. Yu, J. Ren, H. Zhao, and Xu Lu, "Rapid detection of multi-QR codes based on multistage stepwise discrimination and a compressed MobileNet," *IEEE Internet of Things Journal*, vol. 10, no. 18, pp. 15966-15979, 2023.

[22] R. Chen, Y. Huang, K. Lan, J. Li, Y. Ren, X. Hu, L. Wang, H. Zhao, and X. Lu, "A fast adaptive binarization method for QR code images based on dynamic illumination equalization," *Electronics*, vol. 12, no. 19, p. 4134, 2023.

[23] R. Chen, C. Yao, X. Zeng, Y. Ma, J. Yuan, J. Li, H. Zhao, X. Lu, and J. Ren, "Large-scale cross-modal hashing via Kolmogorov-Arnold representation theorem and optimal transport," *Knowledge-Based Systems*, vol. 330, p. 114698, 2025.

[24] R. Chen, P. Wang, B. Lin, L. Wang, X. Zeng, X. Hu, J. Yuan, J. Li, J. Ren, and H. Zhao, "An optimized lightweight real-time detection network model for IoT embedded devices," *Scientific Reports*, vol. 15, p. 3839, 2025.

[25] J. Yuan, X. Zeng, J. Zhou, J. Li, J. Lv, R. Chen, K. Chen, W. Yang, and Y. Zhang, "Data-driven real-time home energy management system based on adaptive dynamic programming," *Electric Power Systems Research*, vol. 238, p. 111055, 2025.

Attention-Guided Fusion of EfficientNet-B0 and Swin Transformer for Cervical Cancer Classification

Twisibile Mwalughali[1], Emmanuel C. OGU[2], Evason Karanja[3]
Department of Mathematical Sciences
Pan African University Institute for Basic Sciences Technology and Innovation (PAUSTI), Nairobi, Kenya[1]
Department of Information Technology, Babcock University, Ilishan-Remo, Ogun State, Nigeria[2]
School of Computing and Information Technology (SCIT),
Jomo Kenyatta University of Agriculture and Technology (JKUAT), Nairobi, Kenya[3]

***Abstract*—The interpretation of colposcopy images is a critical yet subjective component of cervical cancer screening. To enhance this process, we propose a novel hybrid deep learning framework for the classification of cervical lesions. Our model integrates EfficientNet-B0, adept at extracting localized hierarchical features, with a Swin-Tiny Transformer, which excels at modeling long-range dependencies and global context. Moving beyond basic fusion techniques, we introduce a novel cross-attention fusion mechanism, augmented with channel and spatial attention modules. This design selectively highlights the most discriminative inter-feature relationships while maintaining computational efficiency. Evaluated on the International Agency for Research on Cancer (IARC) colposcopy image dataset, our framework achieves an accuracy of 94.76%, significantly outperforming a concatenation-based fusion model (83.99%). This represents an absolute improvement of 10.77 percentage points and captures 67.3% of the residual performance margin toward perfect accuracy. The model also demonstrates robust performance across other metrics, including a precision of 94.68%, recall of 94.82%, F1-score of 94.74%, and a Cohen's Kappa of 89.48%. These results indicate that our approach can enhance both the accuracy and reliability of cervical cancer screening, offering valuable support for clinical decision-making.**

***Keywords*—Cervical cancer classification; deep learning model; colposcopy; cross-attention fusion; EfficientNet; Swin Transformer**

I. Introduction

Cancer arises when cells in the body begin to grow uncontrollably, forming malignant tumors that can spread and affect various organs [1]. Cervical cancer specifically develops in the cervix, the narrow passage connecting the uterus to the vagina [2]. It mainly develops from long-lasting infections with certain high-risk types of human papillomavirus (HPV) [3], [4], particularly HPV-16 and HPV-18 [5]. While the immune system typically clears the virus naturally, persistent infections can lead to cellular changes that may eventually develop into cancer [4]. This disease represents a significant global health burden, particularly in developing countries where it ranks as the third deadliest cancer among women [6]. Approximately 85% of cervical cancer deaths occur in low- and middle-income countries [3], reflecting unequal access to vaccination, screening, and treatment services. The burden is overwhelmingly highest in sub-Saharan Africa, where countries like Eswatini, Zambia and Malawi report incidence rates up to 25 times greater than those in Western Asia, underscoring a critical need for targeted public health interventions [7]. In response, WHO has launched a global strategy aiming to achieve 90% HPV vaccination coverage, 70% screening participation and 90% treatment access by 2030 [8], [9].

Despite the availability of effective prevention strategies, cervical cancer continues to pose a significant global health challenge [6]. According to projections, the global burden of cervical cancer is expected to rise substantially over the coming decades due to population growth and aging. Estimates indicate that the number of cases will increase by 56.8% between 2022 and 2050, reaching nearly 1 million cases, while deaths are projected to rise by 80.7%, totaling approximately 630,000 by 2050 [7]. Patient prognosis varies significantly depending on the stage at which cervical cancer is diagnosed. When detected early, the five-year relative survival rate is approximately 91% [8]. However, survival rates decline sharply in advanced stages, with metastatic cases showing a five-year relative survival rate of about 19% . This big difference shows why it is so important to find and treat cervical cancer early, especially in places where people do not have easy access to healthcare and are often diagnosed late.

Getting an accurate and timely diagnosis is crucial to reduce mortality from cervical cancer [10]. The standard tiered approach utilizes Pap smears, HPV testing, and colposcopy [6]. Primary screening typically begins with a Pap smear to detect abnormal cells or HPV testing to identify high-risk viral strains. Positive results from these initial tests are followed by colposcopy, which provides a magnified visual examination of the cervix to locate and assess abnormalities and guide targeted biopsies for definitive diagnosis. Although Pap smears and HPV tests are effective screening tools, their utility is limited by cost, laboratory requirements, and the potential for false results [11]. Colposcopy is therefore a critical diagnostic follow-up that enables timely intervention [12]. However, its effectiveness is constrained by reliance on clinician expertise for interpreting visual findings, a challenge that is particularly pronounced in resource-limited settings [6]. This dependence on subjective interpretation and the scarcity of trained specialists underscore the need for assistive technologies, thereby motivating the integration of artificial intelligence (AI) to standardize and enhance colposcopic evaluation [6].

Recent advances in artificial intelligence, particularly deep learning models such as convolutional neural networks (CNNs) and vision transformers like the Swin Transformer, have emerged as powerful tools in medical imaging [13], [14]. These models excel at analyzing complex patterns with high accuracy

and consistency, often detecting subtle tissue changes that are imperceptible to the human eye [15]. Such capabilities make deep learning a promising pathway for developing diagnostic tools that address current limitations in early detection and support the WHO's goal of eliminating cervical cancer as a public health problem [10]. To build on this progress, hybrid architectures that integrate CNNs and Transformer-based models are gaining recognition for their effectiveness in cancer classification [16]. CNNs specialize in local spatial feature extraction [17], while Transformers capture long-range dependencies [18], [19], and their combination enables a more holistic analysis of cervical images. Despite this potential, most fusion strategies still rely on limited approaches such as concatenation or element-wise addition, which fail to fully exploit the complementarity between the two architectures. Addition assumes features can be summed directly, but this often results in features with larger magnitudes overwhelming those with smaller values, erasing critical discriminative information [20]. Conversely, concatenation preserves all information by simply appending feature vectors. While this avoids the suppression issue, it naively treats all features as equally important and substantially increases dimensionality, leading to higher computational cost and reduced model generalization [20], [21].

To address these limitations, we propose a novel hierarchical attention-guided fusion mechanism that differs fundamentally from existing fusion strategies. Unlike prior approaches that use either simple concatenation or addition, our method introduces three complementary attention modules in a carefully designed sequence, each addressing specific weaknesses of conventional fusion. Channel attention adaptively adjusts the importance of each feature channel, ensuring that channels carrying diagnostically meaningful patterns receive appropriate emphasis [27]. Spatial attention highlights the most relevant anatomical regions within the feature channels, guiding the model to focus on clinically meaningful lesion areas rather than diffuse background regions [28]. Finally, the cornerstone of our method, bidirectional cross-attention enables context-aware interaction between the CNN and Transformer branches. Unlike naïve fusion techniques, cross-attention allows each branch to selectively query and integrate information from the other, resulting in a richer and semantically aligned joint representation. Collectively, this integrated architecture minimizes redundancy, aligns semantic representations, and substantially enhances the discriminative power of the hybrid model for cervical cancer diagnosis.

Our key contributions in this study are:

- We developed a hybrid attention-guided CNN–Transformer architecture by integrating EfficientNet-B0 and the Swin Transformer, resulting in a tailored model optimized for accurate cervical cancer classification from colposcopic images.
- We developed a multi-stage attention fusion mechanism that integrates channel attention, spatial attention, and bidirectional cross-attention to effectively refine and combine complementary local and global features, resulting in enhanced predictive performance.
- We validated the effectiveness of the proposed attention-guided fusion strategy through systematic comparison with a simple concatenation-based fusion approach.

II. Literature Review

A. Deep Learning

Deep learning is a branch of machine learning that uses artificial neural networks to automatically learn patterns and features directly from raw data eliminating the need for manual feature engineering [13]. Unlike traditional methods, deep learning excels in tasks like medical imaging by detecting intricate patterns through layered abstraction [40]. CNNs are particularly effective, using trainable filters to capture spatial features and pooling layers to improve invariance to scale, rotation, and translation. Their hierarchical structure enables learning from simple to complex patterns ideal for identifying subtle tissue abnormalities or cellular irregularities [17]. Advanced architectures such as ResNet [41], DenseNet [42], EfficientNet [29], further enhance diagnostic accuracy and efficiency by incorporating residual connections, dense connectivity and compound scaling.

1) EfficientNet: EfficientNet represents a significant advancement in convolutional neural network (CNN) architecture through its innovative compound scaling methodology. Unlike traditional approaches that scale network depth, width, or input resolution independently, EfficientNet employs a principled compound scaling technique that simultaneously and proportionally scales all three dimensions [29]. This strategy, governed by a set of optimally determined scaling coefficients, ensures a balanced increase in model capacity becoming deeper and wider while processing higher-resolution images without compromising computational efficiency. The resulting architecture hierarchically captures increasingly complex features while maintaining a compact and efficient design. When benchmarked against conventional CNNs such as ResNet, DenseNet, and Inception, EfficientNet demonstrates superior performance. For instance, EfficientNet-B7 achieves state-of-the-art performance, attaining 84.3% top-1 accuracy on ImageNet with approximately 66 million parameters [29]. This outperforms deeper networks like ResNet-152, which uses a comparable parameter count but yields lower accuracy, and also surpasses both Inception-v3 and DenseNet in accuracy while requiring fewer computational resources. These characteristics make EfficientNet particularly suitable for medical imaging applications, where high diagnostic precision must often be achieved under significant computational constraints. Among the EfficientNet family (B0–B7), EfficientNet-B0 was selected as the backbone feature extractor in this study due to its optimal balance of accuracy, parameter efficiency, and computational cost. With only 5.3 million parameters and 0.39 GFLOPs, EfficientNet-B0 is capable of extracting fine-grained local features from high-resolution colposcopy images without requiring excessive memory or training time. This makes it particularly suitable for medical imaging applications where computational resources are often limited and high diagnostic precision is essential.

EfficientNet has been widely adopted in medical imaging, with successful applications in breast cancer diagnosis [43], [44], lung cancer detection [45] and cervical cancer [46], [47]. Despite these promising results, EfficientNet shares the

limitations of CNN-based models in capturing long-range dependencies, which motivates its integration with transformer-based architectures for enhanced performance in complex classification tasks.

2) Transformers: The Transformer architecture, introduced by Vaswani et al. in 2017 [35], is distinguished by its capacity to learn long-range contextual relationships within sequential data. This is achieved by segmenting input sequences into smaller units (tokens) augmented with positional encodings. The model employs a self-attention mechanism to derive feature representations, enabling every token to interact with all other tokens in the sequence. This design allows direct and parallelized modeling of relationships between all token pairs, regardless of their distance, while maintaining uniform processing across the sequence. In contrast to convolutional neural networks (CNNs), which are constrained by limited receptive fields [19], Transformers leverage self-attention to incorporate global contextual information from the entire input [18]. This capability is especially beneficial in tasks requiring holistic sequence understanding such as natural language processing and image classification laying the groundwork for their adaptation in visual recognition.

The Vision Transformer (ViT) represents a significant shift in image classification methodology by adapting the Transformer architecture for visual data [48]. Rather than using convolutional operations to induce spatial hierarchies, ViT partitions an image into fixed-size non-overlapping patches, treating each as an individual token. These patch-level tokens are processed by a standard Transformer encoder, which models relationships across the entire image [49]. A typical configuration, ViT-Base, consists of 12 encoder layers that apply multi-head self-attention to extract discriminative features. The final stages include layer normalization which aids training stability at a computational cost and a classification token that aggregates global image representations for prediction.

Building on ViT, the Swin Transformer introduces a hierarchical representation learning strategy [30]. Images are divided into non overlapping patches, which are progressively merged into larger blocks, enabling the model to capture features at multiple scales from local details to global context. A key innovation is the use of shifted windows between layers, which promotes cross-window interaction and enlarges the effective receptive field without resorting to global self-attention. This design enhances flexibility across resolutions and complexities, making Swin Transformer particularly suited for vision tasks requiring fine-grained spatial and contextual understanding. Transformers have shown promising results in medical image analysis, with successful applications in breast cancer classification [50], [51], pulmonary nodule detection [52] and cervical image analysis [53], [54]. These studies underscore the effectiveness of transformer-based models in capturing complex spatial dependencies and enhancing diagnostic accuracy across diverse clinical tasks.

The Swin-Tiny variant was chosen to construct a hybrid backbone with EfficientNet-B0. Its compact architecture provides a compelling balance of representational power and computational efficiency, enabling the effective processing of high-resolution colposcopy images. This lightweight design, which avoids the excessive demands of larger models like ViT-B or Swin-Base, ensures a practical and efficient integration for our diagnostic task.

B. CNN-Transformer Hybrid Models in Medical Imaging

Recent studies have demonstrated the promise of CNN–Transformer hybrids in cervical cancer diagnosis. For example, Mohammed et al. [22] introduced a hybrid model for colposcopy image classification that fused EfficientNet-B0 with a Swin-Tiny Transformer, achieving an AUC of 0.96 and an accuracy of 87.5%. Despite these promising results, their fusion method weighted all features equally, highlighting the need for attention-based mechanisms to better prioritize diagnostically relevant patterns. Similarly, Wang et al. [23] proposed a hybrid architecture combining a 3D convolutional neural network (CNN) and a transformer for predicting N-staging and survival outcomes in non-small cell lung cancer (NSCLC). Evaluated on the NSCLC Radiogenomics and NSCLC-Radiomics datasets, their framework leveraged the CNN to extract local spatial features from volumetric images while the transformer captured global contextual dependencies. Features from both modules were fused via simple concatenation prior to classification. Although the model achieved competitive accuracies of 0.805, 0.828, and 0.819 on the training, validation, and test sets, respectively, the reliance on direct feature concatenation may limit effective cross-modal interaction between CNN and transformer representations. This highlights the need for more sophisticated fusion strategies that can dynamically weigh and integrate complementary features for improved representation learning.

Chen et al. [24] proposed the MFEM-CIN framework, a hybrid CNN–Transformer model for colposcopic image analysis that employs multi-scale feature extraction (MSFE) and multi-scale feature fusion (MSFF) to integrate local and global features. While the multi-scale fusion aims to enrich semantic representation by allowing shallow and deep features to interact, the study does not critically evaluate how effectively the fusion mechanism prioritizes diagnostically relevant patterns. The MSFF approach, though conceptually sound, relies on heuristic fusion without attention-guided weighting, which may limit the model's ability to emphasize clinically significant regions over less informative ones. Consequently, while the framework demonstrates high overall accuracy, the fusion strategy itself may not fully exploit the complementary strengths of CNN and Transformer features, suggesting room for improvement through more advanced, attention-based fusion techniques.

Other notable work includes the research of Liu et al. [25], who developed the CVM-Cervix framework for Pap smear classification. In this approach, a CNN module extracted local cellular details, while a Vision Transformer captured global spatial dependencies within the images. Moving beyond simple fusion strategies, the authors employed a multilayer perceptron (MLP) to integrate these complementary feature sets, achieving an accuracy of 91.72%.

In another study, Abinaya and Sivak [55] proposed a deep learning-based cervical classification system integrating 3D convolutional neural networks (3D CNN) with Vision Transformer (ViT) modules. The 3D CNN extracted spatiotemporal features from cervical images, while multiple ViT models captured higher-level global representations. A 3D feature

pyramid network (FPN) was used for feature concatenation, and a 3D squeeze-and-excitation (SE) block reweighted the fused features, enhancing the discriminative power of diagnostically relevant information. Classification was then performed using a kernel extreme learning machine (KELM) with a radial basis kernel. This fusion strategy, particularly the SE-based feature reweighting, contributed to strong performance, achieving an accuracy of 98.6% and demonstrating its potential as an effective diagnostic support tool for cancer detection.

In a domain beyond human healthcare, Dula et al. [26] investigated the integration of CNNs and Transformers for muzzle-based cattle identification. The study proposed a novel Multi-Head Attention Feature Fusion (MHAFF) mechanism, which effectively combines CNN and Transformer features while preserving their complementary strengths. Experimental evaluations on two public cattle datasets yielded outstanding accuracy rates of 99.88% and 99.52%, surpassing both conventional fusion methods and existing identification systems.

While advanced attention-based fusion strategies, such as the Multi-Head Attention Feature Fusion (MHAFF) used in cattle identification [26], have achieved near-perfect accuracy by effectively balancing CNN and Transformer features, this level of sophistication remains largely untapped in cervical cancer diagnosis. Most current approaches still employ uniform or simplistic fusion schemes that cannot adaptively highlight subtle lesion characteristics. This gap highlights a significant opportunity to develop attention-guided frameworks capable of selectively prioritizing clinically relevant features, thereby enhancing the accuracy of cervical cancer classification.

III. Methodology

This section presents the proposed cervical cancer classification model, illustrated in Fig. 1. The model performs binary classification (normal vs. abnormal) by fusing EfficientNet-B0 and Swin-Tiny through attention-guided mechanisms. Both models accept inputs of 224 × 224 pixels, facilitating seamless alignment and fusion of features within the hybrid backbone. The complementary strengths of these architectures form a robust foundation for feature extraction. To further refine the representations, channel and spatial attention modules (SAM) were applied separately to each branch, enhancing informative channels and highlighting salient regions within their respective domains. Optimizing each branch independently ensures that the subsequent cross-attention mechanism receives more distinct and powerful features, enabling a more effective and informative fusion process.

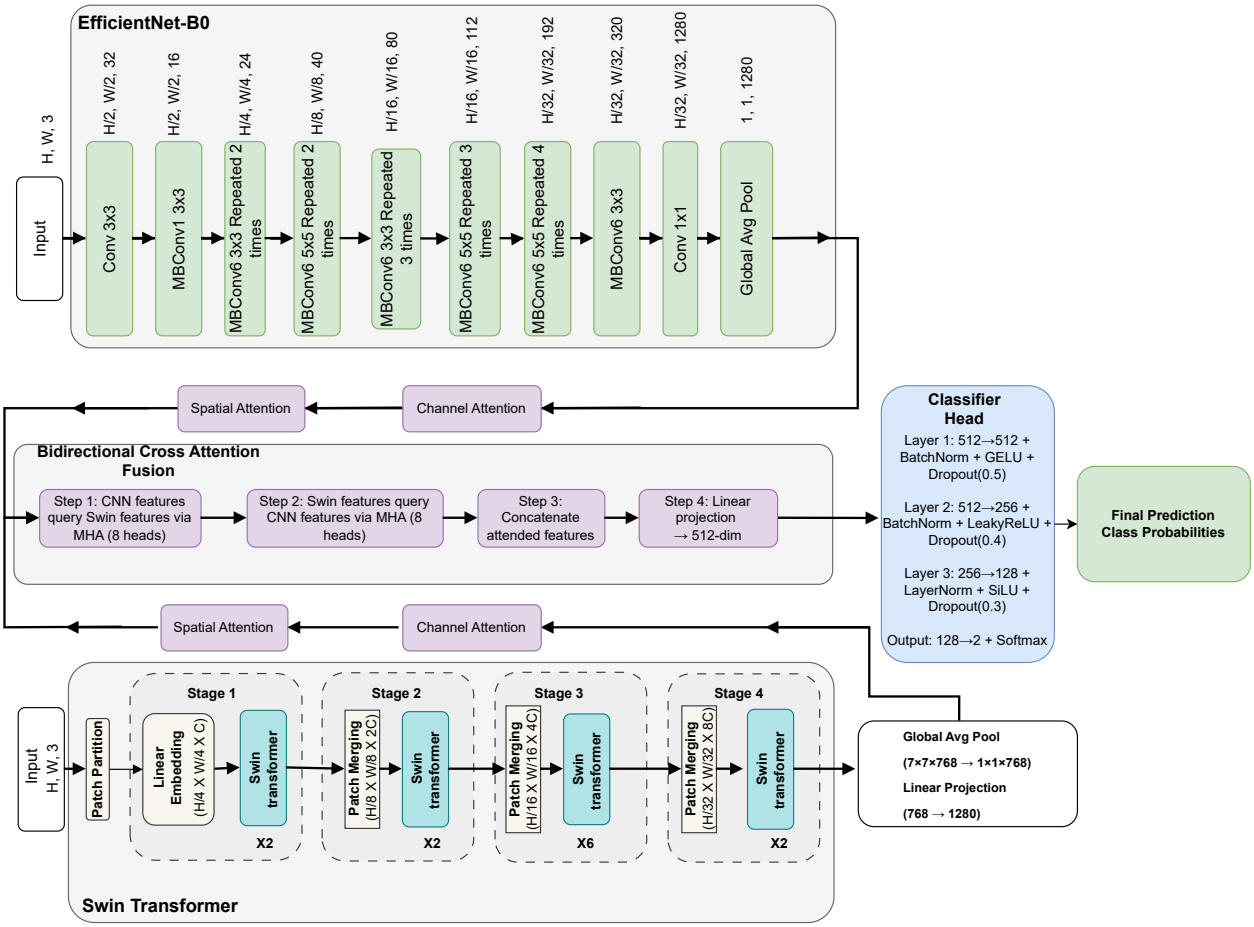
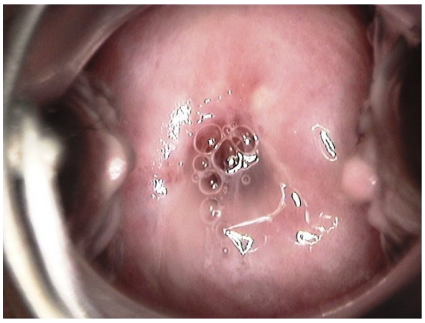


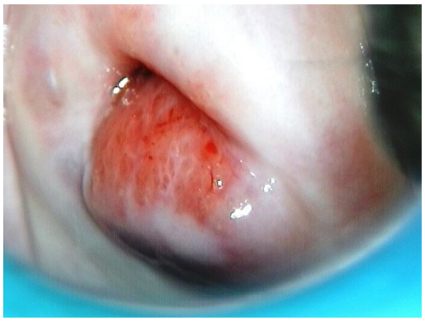
Fig. 1. Proposed dual-branch architecture combining EfficientNet-B0 and Swin Transformer. The model incorporates attention-guided feature fusion to enhance classification performance.

A. Datasets

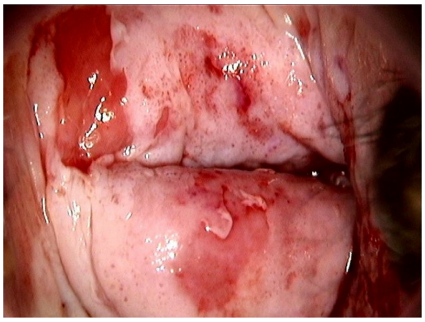
This study utilized two datasets. The first dataset was obtained upon request from the Colposcopy Image Bank of the International Agency for Research on Cancer (IARC)[31]. The original collection included 913 images from 200 patients. After excluding 15 images from three cases with inconclusive clinical assessments, 898 images from 197 cases were retained. These were classified into three categories based on histopathological metadata: normal tissue (93 cases), precancerous lesions (78 cases), and invasive cancer (26 cases). Representative examples from each category are shown in Fig. 2.



(a) Normal.



(b) Precancerous.

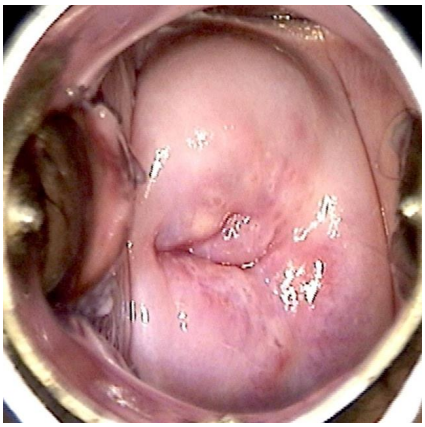


(c) Cancer.

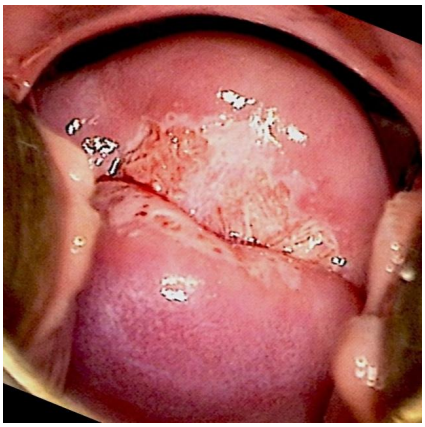
Fig. 2. Colposcopy image samples from the IARC dataset.

The second dataset was sourced from the publicly available Colposcopy Images collection on Kaggle, curated by Siti Nursyafiqah [32]. This dataset contains 2617 images with binary labels: Normal (1445 images) and Abnormal (1172 images). Sample images from this collection are also shown in Fig. 3.

The IARC dataset was reserved exclusively for testing to evaluate the model's generalizability to a new, clinically validated source. A binary classification task was created by merging the precancerous and cancer categories into a single Abnormal class, and the test set was balanced at the case level to avoid bias from patients contributing many images. Iodine-stained images were removed from the dataset to ensure consistency in image appearance. In total, the test set comprised 317 Normal images from 90 cases and 370 Abnormal images from 90 cases (including all 26 cancer cases and 64 precancerous cases).



(a) Normal.



(b) Abnormal.

Fig. 3. Representative colposcopy image samples from the Kaggle dataset: Normal and abnormal.

The Kaggle dataset, which originally contained 1445 Normal and 1172 Abnormal images, was used for training and validation. To ensure consistency with the IARC distribution, a validation set of 317 Normal and 370 Abnormal images was first selected directly from the original Kaggle dataset prior to augmentation. The remaining 1128 Normal and 802 Abnormal images were then augmented fourfold using controlled transformations: brightness adjustment (factors between 0.5× and 1.5×), random zooming up to 0.6×, and random cropping to simulate variability in image framing.All augmented images were stored separately to ensure traceability, producing an augmented pool of 4512 Normal and 3208 Abnormal images. From this augmented pool, 2536 Normal and 2960 Abnormal images were allocated to the training set. Together with the 317/370 validation images and the 317/370 IARC test images,

this produced an exact 80:10:10 split across training, validation, and testing, yielding a per class total of 3170 Normal and 3700 Abnormal images for the experiment(see Fig. 4 and Table I). To address the class imbalance, class weights were applied in the weighted categorical cross-entropy loss function, giving higher importance to the underrepresented Normal class and reducing bias toward the majority Abnormal class.

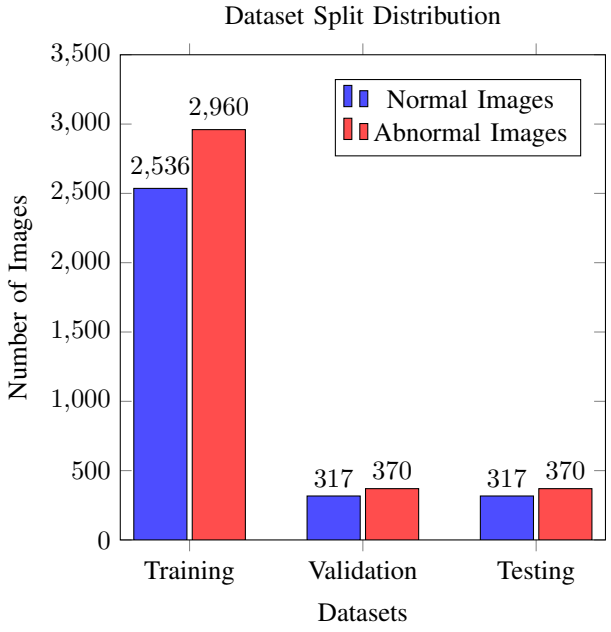


Fig. 4. Distribution of normal and abnormal images across training, validation, and testing sets.

TABLE I. NUMERICAL DISTRIBUTION OF NORMAL AND ABNORMAL IMAGES ACROSS TRAINING, VALIDATION, AND TESTING SETS.

Dataset Split	Normal Images	Abnormal Images
Training	2536	2960
Validation	317	370
Testing	317	370
Total	3170	3700

B. Data Preprocessing

All images in the dataset were processed through a standardized preprocessing pipeline to ensure consistency, improve model convergence, and enhance generalization. The pipeline consisted of the following steps:

Resizing: All images were resized to 224 × 224 pixels to match the input requirements of the ImageNet-pretrained architectures used in this study, specifically EfficientNet-B0 and Swin Transformer-Tiny [29], [30]. This standardization ensures consistent spatial dimensions across samples and facilitates seamless feature fusion between model branches.

Normalization: Pixel values were first scaled from the original 0–255 range to [0,1] by dividing by 255. They were then normalized to the range [-1,1] by subtracting 0.5 and dividing by 0.5 for each color channel. This centers the data distribution, stabilizes gradient-based learning, accelerates convergence, and aligns input statistics with those used during pretraining of the backbone models [13].

Data Augmentation: To increase the diversity of the training data and mitigate overfitting [33], several augmentation techniques were applied during training. These included brightness adjustment (random scaling of pixel intensity between 0.5× and 1.5×), random zooming (scaling factor up to 0.6× of the original image size), and random cropping to simulate variability in framing and viewpoint.

C. Feature Extraction

Feature extraction is the process of transforming complex, high-dimensional raw data, such as images or text into a compact set of informative numerical attributes known as a feature vector [36], [37]. In this study, feature extraction was performed using two parallel networks: EfficientNet-B0 and Swin Transformer (Tiny configuration with patch size 4 and window size 7), both initialized with ImageNet-pretrained weights. To adapt the networks to the colposcopy image classification task, the original classification heads were discarded and the remaining network parameters were fine-tuned during training. This design allows the models to retain their pretrained representational capacity while adjusting to domain-specific visual characteristics present in colposcopy data, forming a stable and efficient foundation for the subsequent stages of the proposed framework [34].

EfficientNet-B0 uses compound scaling to balance network depth, width, and input resolution, allowing it to extract hierarchical features efficiently while keeping computational costs low [29]. The input image $\mathbf{X} \in \mathbb{R}^{3\times224\times224}$ is processed through a sequence of convolutional layers and mobile inverted residual blocks, which progressively capture features from low-level spatial details, such as edges and textures, to high-level semantic patterns relevant for classification. After these layers, a global average pooling operation is applied, which compresses each feature map into a single value by averaging over its spatial dimensions. This produces a 1280-dimensional feature vector, where each dimension represents a summary of a distinct feature detected by the network. This vector is then refined through channel and spatial attention modules,followed by bidirectional cross-attention with the Swin Transformer features, making it suitable for integration with them for subsequent processing.

The Swin Transformer (Tiny variant, patch size 4×4, window size 7×7) processes the image as a sequence of non-overlapping patches [30]. For an input of size 224×224, the image is partitioned into $\frac{224}{4} \times \frac{224}{4} = 3136$ patches. After propagation through the transformer blocks, the final patch embeddings are averaged to produce a 768-dimensional feature vector. To ensure dimensional compatibility with the EfficientNet branch, a linear projection is applied as shown in Eq. (1):

$$\mathbf{f}_{\text{swin}} = W^{\top} \cdot \text{Swin-T}(\mathbf{X}) + b, \quad \mathbf{f}_{\text{swin}} \in \mathbb{R}^{1280}, \tag{1}$$

where, $W \in \mathbb{R}^{768\times1280}$ and $b \in \mathbb{R}^{1280}$ denote the learnable projection matrix and bias term, respectively.

D. Channel Attention Module

A Channel Attention Module, inspired by the Squeeze-and-Excitation network [27], is integrated to adaptively recalibrate channel-wise feature responses by modeling inter-dependencies between channels. Given an input feature map $\mathbf{f} \in \mathbb{R}^{B\times C\times H\times W}$, a global average pooling operation is first applied to generate a compact channel descriptor $\mathbf{z} \in \mathbb{R}^{B\times C}$. The squeeze operation aggregates spatial information into a single channel-wise statistic, formulated as Eq. (2):

$$z_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} f_c(i, j), \tag{2}$$

where, z_c represents the aggregated response of the c-th channel. Next, an 'excitation' step is performed to capture inter-channel dependencies through a bottleneck structure comprising two fully connected layers. This transformation is expressed in Eq. (3):

$$\mathbf{a} = \sigma(\mathbf{W}_2 \cdot \delta(\mathbf{W}_1 \cdot \mathbf{z})), \tag{3}$$

where, $\mathbf{W}_1 \in \mathbb{R}^{C/r\times C}$ and $\mathbf{W}_2 \in \mathbb{R}^{C\times C/r}$ are learnable parameters, $r = 16$ is the reduction ratio, δ denotes the ReLU activation, and σ is the sigmoid function. The resulting activation vector $\mathbf{a}$ serves as channel attention weights. Finally, these weights are applied to the original feature map via channel-wise multiplication, as shown in Eq. (4):

$$\tilde{\mathbf{f}} = \mathbf{a} \otimes \mathbf{f}, \tag{4}$$

where, $\otimes$ denotes element-wise channel scaling. This mechanism strengthens informative channels while suppressing less relevant ones, thereby improving the discriminative power of the learned representation.

E. Spatial Attention Module (SAM)

To complement the channel-wise focus, a SAM is incorporated to direct the model's focus towards the most informative spatial regions within the feature maps [28]. This module generates a spatial attention map that highlights where an informative part is located, regardless of channel-specific responses. Given an input feature map $\mathbf{f} \in \mathbb{R}^{B\times C\times H\times W}$, we first aggregate channel information by applying both average pooling and max pooling operations along the channel axis, as shown in Eq. (5) and Eq. (6), respectively:

$$f_{\text{avg}} = \frac{1}{C} \sum_{c=1}^{C} f_c, \quad f_{\text{avg}} \in \mathbb{R}^{B\times 1\times H\times W} \tag{5}$$

$$f_{\text{max}} = \max_{c=1,\ldots,C} f_c, \quad f_{\text{max}} \in \mathbb{R}^{B\times 1\times H\times W} \tag{6}$$

These maps are then concatenated along the channel dimension to form a composite feature descriptor:

$$f_{\text{cat}} = [f_{\text{avg}}; f_{\text{max}}], \quad f_{\text{cat}} \in \mathbb{R}^{B\times 2\times H\times W} \tag{7}$$

where, $[;]$ denotes concatenation.

A convolutional layer with a 7×7 kernel is applied to this concatenated descriptor in Eq. 7 to produce a preliminary spatial map. This is followed by a sigmoid activation function σ to generate the final spatial attention map $\mathbf{a_s}$ in Eq. (8), with values constrained between 0 and 1:

$$\mathbf{a_s} = \sigma\left(\text{Conv}^{7\times 7}(f_{\text{cat}})\right), \quad \mathbf{a_s} \in \mathbb{R}^{B\times 1\times H\times W} \tag{8}$$

The output of the module is obtained by multiplying this attention map in Eq. (8) across every channel of the original input feature map, as shown in Eq. (9):

$$\tilde{f} = \mathbf{a_s} \otimes \mathbf{f} \tag{9}$$

where, $\otimes$ denotes element-wise multiplication. This mechanism emphasizes feature responses in salient regions while suppressing information from irrelevant areas, thereby enhancing the model's spatial discriminative capabilities.

F. Bidirectional Cross-Attention Mechanism

To facilitate synergistic interaction between the feature representations from the EfficientNet-B0 and Swin Transformer branches, a bidirectional cross-attention mechanism with multi-head attention (MHA) was implemented. This allows each branch to dynamically attend to and assimilate relevant features from the other, fostering a more cohesive and discriminative fused representation.

Let $\mathbf{F}_e \in \mathbb{R}^{B\times L\times C}$ and $\mathbf{F}_s \in \mathbb{R}^{B\times N\times D}$ denote the reshaped feature tensors from the EfficientNet and Swin Transformer branches, respectively, where $L = H \times W$ represents the spatial token length, N is the number of Swin tokens, and C, D are their corresponding feature dimensions. The bidirectional cross-attention is formalized through two symmetric attention blocks:

1) EfficientNet to Swin Transformer (E → S) direction: The Swin Transformer features act as the Query, seeking information from the EfficientNet branch's Key and Value [35]:

$$Q_s = F_s W_s^Q, \qquad Q_s \in \mathbb{R}^{B\times N\times d_k}, \tag{10a}$$
$$K_e = F_e W_e^K, \qquad K_e \in \mathbb{R}^{B\times L\times d_k}, \tag{10b}$$
$$V_e = F_e W_e^V, \qquad V_e \in \mathbb{R}^{B\times L\times d_v}. \tag{10c}$$

2) Swin Transformer to EfficientNet (S → E) direction: The EfficientNet features act as the Query, incorporating contextual information from the Swin branch:

$$Q_e = F_e W_e^Q, \qquad Q_e \in \mathbb{R}^{B\times L\times d_k}, \tag{11a}$$
$$K_s = F_s W_s^K, \qquad K_s \in \mathbb{R}^{B\times N\times d_k}, \tag{11b}$$
$$V_s = F_s W_s^V, \qquad V_s \in \mathbb{R}^{B\times N\times d_v}. \tag{11c}$$

where, $\mathbf{W}^Q, \mathbf{W}^K, \mathbf{W}^V$ in Eq. (10) and Eq. (11) are learnable projection weights.

The Multi-Head Attention (MHA) mechanism with $h = 8$ heads is applied independently in each direction. For the E $\rightarrow$ S direction, the projected Query, Key, and Value tensors are first split into h distinct heads along the feature dimension. The operations for the i-th head are then defined as shown in the Eq. (12) :

$$Q_s^i = Q_s W_{Q_s}^i, \qquad Q_s^i \in \mathbb{R}^{B\times N\times d_h}, \tag{12a}$$
$$K_e^i = K_e W_{K_e}^i, \qquad K_e^i \in \mathbb{R}^{B\times L\times d_h}, \tag{12b}$$
$$V_e^i = V_e W_{V_e}^i, \qquad V_e^i \in \mathbb{R}^{B\times L\times d_h}. \tag{12c}$$

where, $d_h = d_k/h = d_v/h$ is the dimension per head, and $\mathbf{W}_{Q_s}^i, \mathbf{W}_{K_e}^i, \mathbf{W}_{V_e}^i$ are the projection weights for the i-th head. For each head, the scaled dot-product attention is computed independently. This involves calculating a compatibility score between each Query and Key vector, which determines how much focus to place on different parts of the input [see Eq. (13)]:

$$\text{Attention}(Q_s^i, K_e^i, V_e^i) = \text{softmax}\left(\frac{Q_s^i (K_e^i)^\top}{\sqrt{d_h}}\right) V_e^i. \tag{13}$$

The softmax function is applied row-wise to generate a probability distribution (attention weights) over the L spatial locations for each of the N tokens. The output for each head is consequently a weighted sum of the Value vectors, with weights determined by the Query-Key compatibility. The outputs of all h heads are then concatenated along the feature dimension, as shown in the Eq. (14):

$$\begin{aligned} H_s &= [\text{Head}_s^1; \text{Head}_s^2; \ldots; \text{Head}_s^h], \\ H_s &\in \mathbb{R}^{B\times N\times (h\cdot d_h)} = \mathbb{R}^{B\times N\times d_v}. \end{aligned} \tag{14}$$

Finally, the concatenated output is linearly projected back to the original dimension D using a learned output matrix $\mathbf{W}_O \in \mathbb{R}^{d_v\times D}$, [see Eq. (15)]:

$$\text{MHA}(Q_s, K_e, V_e) = H_s W_O. \tag{15}$$

This multi-head design allows the model to jointly attend to information from different representation subspaces at different positions, effectively capturing a diverse set of complementary features. The final refined feature for the Swin branch is obtained via a residual connection and layer normalization, as shown in Eq. (16):

$$\tilde{F}_s = \text{LayerNorm}(F_s + \text{MHA}(Q_s, K_e, V_e)). \tag{16}$$

An identical process is applied in the S $\rightarrow$ E direction to obtain $\tilde{\mathbf{F}}_e$. The enriched feature representations, $\tilde{\mathbf{F}}_s$ and $\tilde{\mathbf{F}}_e$, produced by the bidirectional cross-attention module are fused via global average pooling followed by concatenation, yielding a unified feature vector $\mathbf{z} \in \mathbb{R}^{B\times 512}$. This vector is subsequently processed by a Multi-Layer Perceptron (MLP) classifier to transform the high-level features into a probability distribution over the C target classes, where C = 2 in our case.

G. Multi-Layer Perceptron (MLP) Classifier

The fused feature vector $z \in \mathbb{R}^{512}$, obtained from the bidirectional cross-attention module, is processed by a Multi-Layer Perceptron (MLP) classifier. This network transforms the input features into a probability distribution over the target classes. The classifier comprises three hidden layers, progressively reducing the dimensionality from 512 to 128, followed by a linear output layer mapping to the number of output classes C. The dimensionality reduction follows the sequence: $512 \rightarrow 256 \rightarrow 128 \rightarrow C$.

The input to the first layer is the fused feature vector $h^{(0)} = z$. The layer applies a linear transformation using a weight matrix $W^{(1)} \in \mathbb{R}^{512\times 512}$ and bias vector $b^{(1)} \in \mathbb{R}^{512}$, as shown in Eq. (17). Batch normalization is applied to the linear output, followed by a GELU activation and dropout. The resulting output, hereafter denoted as $h^{(1)}$, serves as the input to the next layer.

$$z^{(1)} = h^{(0)} W^{(1)} + b^{(1)} \tag{17}$$

The second layer receives $h^{(1)} \in \mathbb{R}^{512}$ and projects it to 256 dimensions using $W^{(2)} \in \mathbb{R}^{512\times 256}$ and $b^{(2)} \in \mathbb{R}^{256}$, as shown in the Eq. (18).The linear output is subsequently normalized through batch normalization, followed by the application of a LeakyReLU activation function with a negative slope and dropout, producing $h^{(2)}$, which serves as the input to the third layer.

$$z^{(2)} = h^{(1)} W^{(2)} + b^{(2)} \tag{18}$$

The third hidden layer reduces the dimensionality of $h^{(2)}$ from 256 to 128 using the weight matrix $W^{(3)} \in \mathbb{R}^{256\times 128}$ and bias vector $b^{(3)} \in \mathbb{R}^{128}$, as shown in Eq. (19). The linear output is subsequently normalized through layer normalization, followed by the application of a SiLU (Swish) activation function and dropout, yielding the final hidden representation $h^{(3)}$.

$$z^{(3)} = h^{(2)} W^{(3)} + b^{(3)} \tag{19}$$

The output layer linearly projects $h^{(3)} \in \mathbb{R}^{128}$ to C dimensions using $W^{(4)} \in \mathbb{R}^{128\times C}$ and $b^{(4)} \in \mathbb{R}^C$, producing the logits u, as shown in Eq. (20). These logits are then converted into a probability distribution over the output classes using the softmax function [Eq. (21)]. Finally, the predicted class $\hat{c}$ is obtained by selecting the class with the highest probability, as defined in Eq. (22).

$$u = h^{(3)} W^{(4)} + b^{(4)} \tag{20}$$

$$\hat{y} = \text{softmax}(u) \tag{21}$$

$$\hat{c} = \arg\max_j \hat{y}_j \tag{22}$$

IV. Experimental Setup and Evaluations

A. Computing Environment

All experiments were conducted in the Google Colab environment equipped with an NVIDIA GPU, leveraging the CUDA architecture for accelerated deep learning computations. The software environment was built on Python 3.10.12 using the PyTorch deep learning framework. Key libraries included torchvision for image preprocessing and data loading, timm (Torch Image Models) for access to pre-trained backbone networks, and scikit-learn for comprehensive evaluation metrics.

B. Training Configuration and Hyperparameters

For both the baseline concatenation model and the proposed attention-guided fusion model, identical training configurations were employed to ensure a fair comparison. The training hyperparameters used in this study are summarized in Table II. The models were trained for 50 epochs with a batch size of 8 using the AdamW optimizer, a learning rate of 5×10^{-5}, and a weight decay of 0.01. These hyperparameters were selected based on prior transformer-based literature, preliminary experimental validation, and computational constraints associated with training dual-branch architectures. Model performance was evaluated using 5-fold stratified cross-validation to ensure robustness and reduce variance, and the reported metrics correspond to the aggregated performance across all folds.

AdamW was chosen because its decoupled weight decay improves generalization and helps mitigate overfitting, particularly in transformer-based networks [39]. The learning rate was determined through pilot experiments that demonstrated stable convergence during fine-tuning of pre-trained models, while a weight decay of 0.01 provided effective regularization without limiting model capacity. The batch size of 8 was necessitated by GPU memory limitations; however, smaller batch sizes have also been shown to promote better generalization in medical imaging tasks with moderately sized datasets. Training for 50 epochs ensured adequate convergence, as validation performance plateaued beyond this point during preliminary experiments.

The hidden layers utilized GELU, LeakyReLU, and SiLU activation functions to exploit their complementary strengths: GELU provides smooth non-linearity suitable for transformer components, LeakyReLU prevents the dying neuron problem in convolutional layers, and SiLU introduces self-gating behavior that enhances feature representation. A weighted categorical cross-entropy loss function was applied to address class imbalance within the dataset. Maintaining this consistent training configuration across all experiments ensures that any observed performance differences can be attributed to architectural innovations rather than variations in optimization strategy.

TABLE II. Training Hyperparameters used for Both the Baseline Concatenation Model and the Proposed Attention-Guided Fusion Model

Hyperparameter	Value
Epochs	50
Batch Size	8
Optimizer	AdamW
Learning Rate	5×10^{-5}
Weight Decay	0.01
Loss Function	Weighted Categorical Cross-Entropy
Activation Function (Hidden Layers)	GELU, LeakyReLU, SiLU
Activation Function (Output Layer)	Softmax

C. Metrics for Evaluating Performance

Model performance was assessed using standard classification metrics. For binary classification, let TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. The following metrics were computed [see Eq. (23) to Eq. (27)]:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{23}$$

Accuracy measures the overall proportion of correctly classified samples out of all predictions.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{24}$$

Precision evaluates how many of the samples predicted as positive are actually positive, reflecting reliability in identifying abnormal cases.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{25}$$

Recall (or sensitivity) measures how many actual positive samples are correctly identified, which is critical in medical diagnosis to minimize missed cases.

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{26}$$

The F1-Score balances precision and recall, providing a single measure of overall classification effectiveness, especially useful when dealing with imbalanced datasets.

$$\text{Cohen's Kappa} = \frac{P_o - P_e}{1 - P_e} \tag{27}$$

Cohen's Kappa measures the agreement between predicted and actual classifications while accounting for agreement occurring by chance. It provides a more robust evaluation than accuracy alone, particularly in imbalanced datasets.

The observed agreement P_o is defined as Eq. (28):

$$P_o = \frac{TP + TN}{TP + TN + FP + FN} \tag{28}$$

The expected agreement P_e is computed as Eq. (29):

$$P_e = \frac{(TP+FP)(TP+FN)}{(TP+TN+FP+FN)^2} + \frac{(FN+TN)(FP+TN)}{(TP+TN+FP+FN)^2} \quad (29)$$

Kappa values range from -1 to 1, where $\kappa = 1$ indicates perfect agreement, $\kappa = 0$ indicates agreement equivalent to random chance, and $\kappa < 0$ indicates disagreement worse than chance. Higher Kappa values, therefore, indicate stronger and more reliable classification performance beyond random coincidence.

To complement these quantitative metrics, performance is further analyzed using visualization techniques such as confusion matrices, Receiver Operating Characteristic (ROC) curves, and learning curves, which provide deeper insight into class-wise behavior, discriminative capability and convergence characteristics of the models.

D. Experimental setup

To rigorously assess the effectiveness of the proposed attention-guided fusion strategy, we implemented two experimental configurations. The baseline model, which used simple concatenation to combine CNN and transformer representations, provided a reference for performance without attention mechanisms. The proposed model incorporated an attention-guided fusion strategy. This approach dynamically reweights features from both branches to emphasize diagnostically relevant patterns.

1) Concatenation-based fusion model (baseline): The concatenation-based fusion model was designed as a baseline to assess the contribution of attention mechanisms. In this configuration, feature representations extracted from EfficientNet-B0 and Swin Transformer–Tiny were directly concatenated along the feature dimension without applying any attention operation. Specifically, the 1280-dimensional feature vector produced by EfficientNet-B0 and the 768-dimensional feature vector obtained from Swin Transformer–Tiny were concatenated to form a 2048-dimensional fused representation. The resulting feature vector was processed by a Multi-Layer Perceptron (MLP) classifier consisting of three sequential hidden layers with 512, 256, and 128 units, respectively. Each hidden layer incorporated normalization and nonlinear activation functions, employing GELU, LeakyReLU, and SiLU activations in successive layers, along with dropout rates of 0.5, 0.4, and 0.3, to improve generalization and mitigate overfitting. The classifier was finalized with a linear output layer for binary classification. Model optimization was performed using the AdamW optimizer with a learning rate of5×10^{-5} and a weight decay of 0.01. This baseline model provides a strong yet simple fusion strategy, serving as a reference point for evaluating the performance gains achieved by the proposed attention-based fusion framework.

2) Attention-based fusion model (proposed model): As illustrated in the Methodology section, the proposed model introduces a multi-stage attention-based fusion framework to effectively integrate features extracted from EfficientNet-B0 and Swin Transformer–Tiny. Following independent feature extraction, the 768-dimensional output of the Swin Transformer–Tiny branch is projected into a 1280-dimensional feature space via a linear transformation to ensure dimensional compatibility with the EfficientNet-B0 branch. Prior to feature fusion, each branch is refined separately using sequential Channel Attention and Spatial Attention modules. The Channel Attention module employs adaptive average pooling followed by a sigmoid-activated MLP to compute channel-wise importance weights, while the Spatial Attention module emphasizes salient spatial regions through mean and max pooling operations followed by a one-dimensional convolution. The core fusion mechanism is implemented through a CrossAttnFusion module, in which features are projected into Query (Q), Key (K), and Value (V) vectors and cross-attention is computed in both directions. The attended outputs from both directions are concatenated using the torch.cat operation and passed through a linear layer with ReLU activation to form a unified representation. This fused representation is subsequently processed by a multi-layer classifier that progressively reduces the feature dimensionality from 512 to 256 and finally 128 units. Each stage incorporates normalization and nonlinear activation functions GELU, LeakyReLU, and SiLU along with dropout rates of 0.5, 0.4, and 0.3, respectively, to enhance robustness and generalization. The final classification layer maps the 128-dimensional representation to two output units, corresponding to the normal and abnormal classes. Training was conducted using the Weighted Categorical Cross-Entropy loss function, implemented via PyTorch's `nn.CrossEntropyLoss` with class weights to account for dataset imbalance. The model was optimized using the AdamW optimizer with a learning rate of 5×10^{-5} and a weight decay of 0.01. During inference, class predictions were obtained using `argmax(1)`, while class probabilities for ROC analysis and other evaluation metrics were computed using `torch.softmax(outputs, 1)`.

This experimental design isolates the effect of the attention mechanism, allowing direct attribution of any performance differences to the fusion strategy rather than extraneous factors.

V. Results and Discussion

A. Quantitative results

As shown in Table III, the proposed model demonstrates consistent and substantial improvements over baseline model across all evaluation metrics. Classification accuracy increases by 10.77 percentage points, indicating a marked enhancement in overall predictive performance. Similarly, the F1-score improves by 10.76 percentage points, reflecting more balanced classification across classes. Notably, Cohen's Kappa exhibits a substantial increase of 21.45 percentage points, highlighting a significantly stronger agreement beyond chance and confirming the robustness of the proposed attention-guided fusion strategy.

B. Qualitative Results

The confusion matrices presented in Fig. 5 and Fig. 6 provide a detailed class-wise evaluation of the models on the test dataset.The results demonstrate a clear performance improvement when using the attention-guided fusion model compared to the simple concatenation model. In the simple

TABLE III. PERFORMANCE COMPARISON ON THE TEST DATASET

Metric	Baseline Model (%)	Proposed Model (%)	Absolute Improvement(%)
Accuracy	83.99	94.76	+10.77
Precision	84.09	94.68	+10.59
Recall	84.28	94.82	+10.54
F1-score	83.98	94.74	+10.76
Cohen's Kappa	68.03	89.48	+21.45

concatenation approach, although the model achieved a reasonable number of correct classifications (298 abnormal and 279 normal), it produced a relatively high number of false negatives (72 abnormal cases misclassified as normal), which is particularly concerning in a medical diagnostic context where missed abnormal cases can delay treatment. In contrast, the attention-guided model substantially reduced false negatives from 72 to 22 and false positives from 38 to 14, while simultaneously increasing correct abnormal predictions (from 298 to 348) and correct normal predictions (from 279 to 303).These improvements indicate that the attention mechanism enhances feature discrimination by prioritizing more informative representations within the hybrid architecture, thereby improving overall classification reliability.

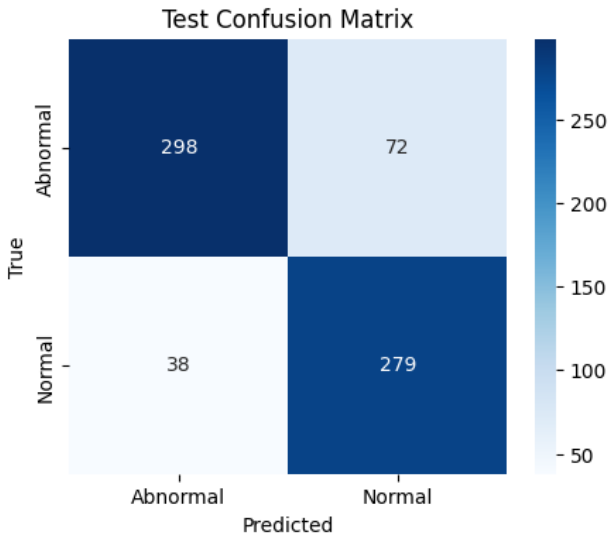


Fig. 5. Confusion matrix for the concatenation-based fusion model on the test dataset.

This finding is strongly corroborated by the Receiver Operating Characteristic (ROC) curves in Fig. 7 and Fig. 8. While the baseline model achieved a respectable Area Under the Curve (AUC) of 0.90 for both classes, the proposed model attained a near-perfect AUC of 0.99. This significant increase in AUC, consistent with the sharp reduction in confusion matrix errors, underscores that the attention mechanism not only boosts overall accuracy but also substantially improves the model's confidence and precision in separating the two classes.

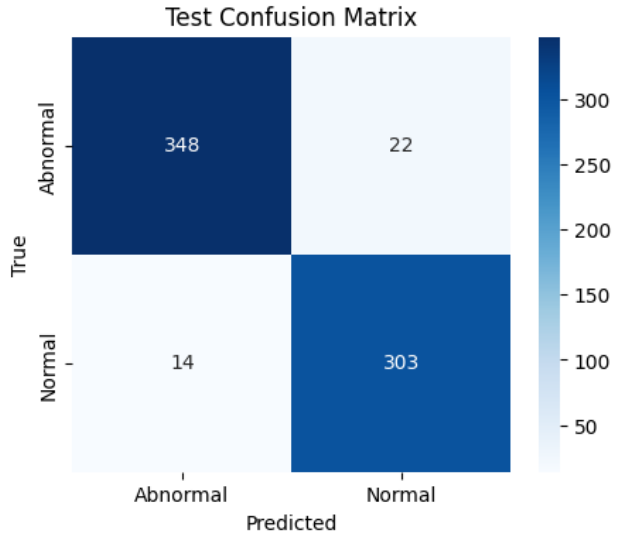


Fig. 6. Confusion matrix for the proposed attention-based fusion model on the test dataset.

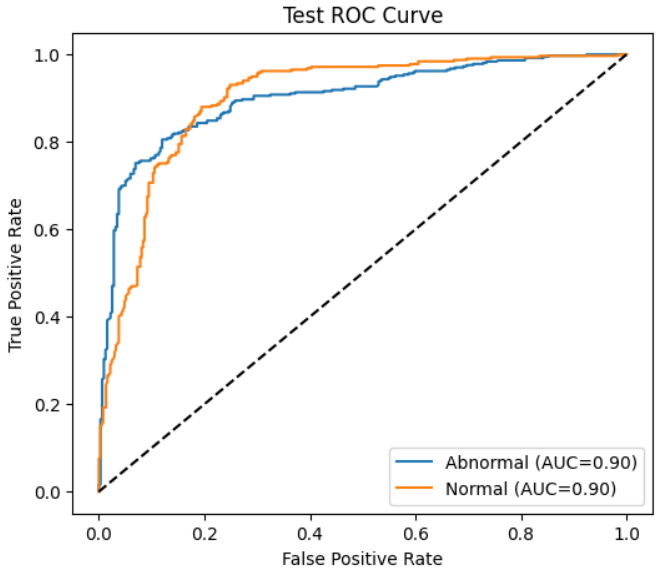


Fig. 7. ROC curve for the concatenation-based fusion model on the test dataset.

C. Learning Curve Analysis

The learning dynamics of the two approaches, illustrated in Fig. 9 and Fig. 10, provide additional insight into their behavior during training. The concatenation-based fusion model exhibits slower convergence, plateauing around epoch 30, and shows variability in generalization during intermediate epochs, which may indicate that the model struggles to locate the most critical regions in the images. By contrast, the proposed model stabilizes much earlier, between epochs 10 and 15, with closely aligned training and validation accuracies, reflecting faster and more consistent learning. This efficiency in convergence, combined with the improved predictive performance, underscores the effectiveness of the attention-guided fusion strategy in

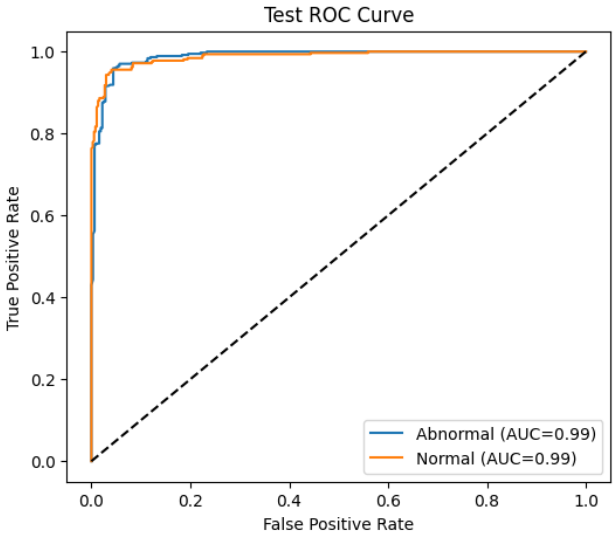


Fig. 8. ROC curve for the proposed attention-based fusion model on the test dataset.

prioritizing clinically relevant features while reducing training time. Collectively, these results highlight the potential of the proposed model as a robust and reliable tool for automated colposcopy image analysis. By improving classification accuracy, reducing misclassifications, and demonstrating stable learning, the model supports safer and more efficient cervical cancer screening workflows, complementing existing diagnostic procedures while minimizing the risk of missed diagnoses.

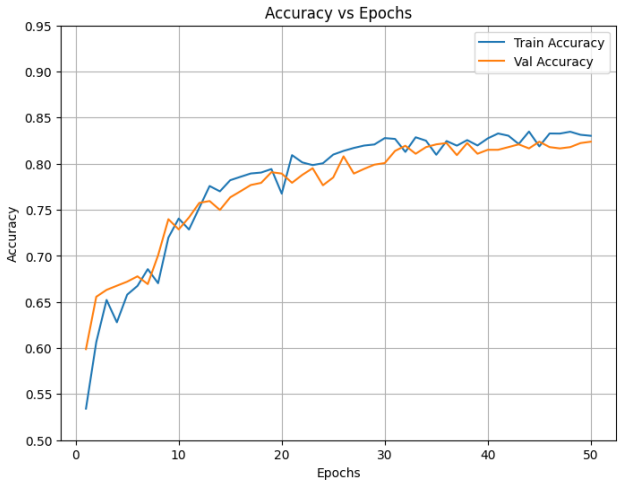


Fig. 9. Training and validation accuracy curve for the baseline model.

D. Model Classification Decisions

The Grad-CAM [38] visualizations in Fig. 11 illustrate that the model effectively highlights the image regions most influential to its classification decisions. The figure presents four representative samples (two normal and two abnormal), with the top row showing the original images and predicted confidence scores, and the bottom row displaying the corresponding Grad-CAM heatmaps. The model achieves confidence scores

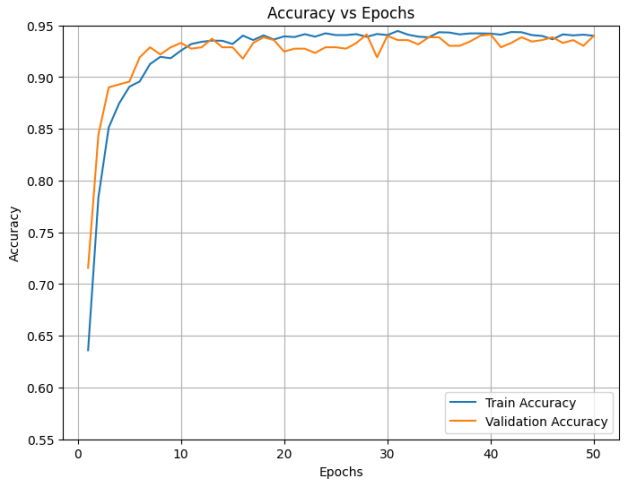


Fig. 10. Training and validation accuracy curve for the proposed attention-based fusion model.

of 99.1% and 99.6% for normal samples and 88.5% and 99.1% for abnormal samples, indicating strong certainty in distinguishing healthy from diseased tissue.

In the heatmaps, red regions denote areas of highest importance, yellow indicate moderate importance, and blue correspond to regions with minimal influence. The strongest activations are consistently concentrated on irregular tissue patterns, including lesion margins, atypical epithelial structures, and abnormal coloration. This focused localization suggests that the model bases its predictions on clinically relevant pathological features rather than background artifacts.

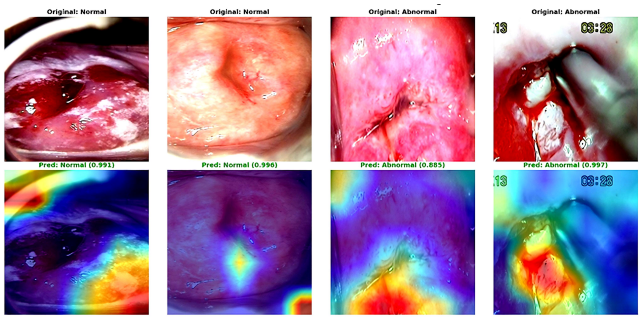


Fig. 11. Grad-CAM visualizations highlighting regions of high and low importance in colposcopy images.

E. Discussion

The attention mechanism fundamentally changes how the hybrid model integrates information from its two component architectures. Rather than simply combining features, it learns to prioritize different types of visual information depending on the input. This selective weighting explains the observed error reduction: false negatives dropped from 72 to 22, and false positives fell from 38 to 14. This pattern is particularly significant in screening contexts, where missed abnormalities carry greater clinical risk than over-calling normal tissue.

The training dynamics further demonstrate the value of attention. The baseline model required more epochs to plateau,

converging around epoch 30, whereas the proposed model stabilized near epoch 10 with closely aligned training and validation curves. This efficiency suggests that attention focuses learning on informative regions from the outset.

The near-perfect AUC of 0.99 reflects how cleanly the model separates classes in feature space. This indicates that the internal representations of normal and abnormal cases form two distinct clusters with minimal overlap, enabling confident predictions across any decision threshold. The attention mechanism achieves this separation by ensuring that only relevant features such as lesion margins, epithelial changes, and vascular patterns contribute to the final representation, while effectively suppressing background noise.

Despite the promising performance, several limitations should be acknowledged. First, the model was trained and evaluated on a relatively small dataset, which may limit its generalizability to larger or more diverse populations. Second, variations in imaging devices, staining protocols, or patient demographics could affect performance when applied to external cohorts. Third, the proposed hybrid model is computationally demanding due to the large size of the EfficientNet and Swin Transformer branches, which constrained the training batch size to 8. Future work should focus on validating the model on larger, more diverse datasets, optimizing computational resource usage, and integrating complementary modalities, such as histopathology images and clinical metadata, to further enhance robustness and predictive performance. These findings demonstrate that the proposed attention-based fusion model outperforms the baseline in accuracy and reliability, providing a clinically meaningful approach for precise and efficient cervical cancer screening.

F. Comparative Analysis with Prior Studies

The proposed model was comparatively evaluated against several representative CNN–Transformer–based cervical image analysis frameworks reported in the literature. The CVM-Cervix model [25] demonstrated the benefit of integrating CNN-extracted cellular features with Vision Transformer–captured global contextual representations; Feature fusion was performed using a multilayer perceptron (MLP), which enables joint feature learning but lacks explicit attention mechanisms to selectively emphasize diagnostically salient regions. Similarly, the hybrid EfficientNet-B0–Swin-Tiny framework [22] highlighted the potential of CNN–Transformer integration but employed a simple concatenation-based fusion mechanism, achieving an accuracy of 87.5% after 30 training epochs. The MFEM-CIN framework [24] further contributed toward lightweight, multi-scale colposcopic image analysis; nevertheless, its multi-scale feature fusion (MSFF) strategy was primarily based on concatenation and linear projection, which may constrain the modeling of complex inter-feature dependencies. Beyond cervical imaging, a related hybrid approach combining a ResNet module with a transformer was proposed in [23] for non-small cell lung cancer (NSCLC) analysis. While the ResNet branch captured local volumetric features and the transformer modeled long-range contextual information, the use of simple feature concatenation resulted in a maximum testing accuracy of 81.9%, underscoring the limitations of uniform feature aggregation. In contrast, the baseline model implemented in this study, which also employed concatenation-based fusion, achieved an accuracy of 83.99%, aligning with the performance trends observed in prior concatenation-driven hybrid architectures. Notably, the proposed attention-guided fusion model substantially outperformed both the baseline and existing methods, achieving an accuracy of 94.76%. This improvement highlights the effectiveness of attention mechanisms in selectively emphasizing diagnostically relevant regions and suppressing redundant features during fusion (see Table IV).

TABLE IV. COMPARISON BETWEEN THE PROPOSED MODEL AND RELATED STUDIES

Study	Model	Image type	Fusion Mechanism	Accuracy (%)
[25]	CNN + Transformer	Pap smear images	Multilayer Perceptron (MLP)	91.72
[22]	EfficientNet-B0 + Swin-Tiny	Colposcopy images	Concatenation	87.50
[24]	Multi-scale CNN + Transformer	Colposcopy images	Concatenation	89.20
[23]	ResNet + Transformer	NSCLC CT images	Concatenation	81.90
Our baseline	EfficientNet-B0 + Swin-T	Colposcopy images	Concatenation	83.99
Our proposed	**EfficientNet-B4 + Swin-T**	**Colposcopy images**	**Attention-guided fusion (Cross attention)**	**94.76**

G. Clinical Integration and Practical Implications

The proposed hybrid EfficientNet-B0 and Swin Transformer model with cross-attention fusion offers meaningful potential to support cervical cancer screening and diagnostic workflows. By processing colposcopy images in real-time, the model can assist clinicians during routine gynecological examinations, delivering rapid and data-informed interpretations. This capability helps mitigate the subjectivity and inter-observer variability common in visual assessment, contributing to more standardized and reproducible evaluations across practitioners.

With its high predictive accuracy and robust generalization, the model serves as a reliable decision-support tool that complements established diagnostic techniques such as Pap smears, HPV testing, and biopsy. Although the current implementation is best suited for systems with GPU support, future optimizations may enable broader adoption even in resource-constrained clinical settings.

Importantly, the model produces interpretable attention maps that highlight key regions in the images, including lesion margins, abnormal epithelial pattern, and areas of unusual coloration. These visual explanations assist pathologists and radiologists in focusing on clinically relevant areas, prioritizing critical casesand making more confident, informed diagnostic decisions. Overall, the system is intended to support rather than replace expert judgment, improving both the speed and reliability of cervical cancer diagnosis in clinical practice.

VI. Conclusion

This study proposed a novel attention-guided hybrid framework for cervical cancer classification, integrating features from EfficientNet-B0 and Swin Transformer. By applying channel-wise and spatial attention prior to fusion, the model effectively captured both global contextual information and localized fine-grained details, addressing the limitations of simple concatenation. The framework achieved strong performance, with a test accuracy of 94.76%, precision of 94.68%, recall of 94.82%, F1-score of 94.74%, and a Cohen's Kappa of 89.48%. Attention mechanisms were crucial in prioritizing salient features, enhancing discriminative power, and reducing misclassifications, while the accelerated and stable convergence during training underscores the model's efficiency.

Although these results are promising, future work should focus on validating the framework on large-scale datasets from diverse clinical institutions to ensure robust generalizability. Incorporating complementary patient information, such as clinical history and histopathology reports, could further improve diagnostic accuracy and utility. Additionally, the flexible design of the framework presents an opportunity to extend its application to other gynecological cancers and imaging modalities, broadening its potential clinical impact.

Acknowledgment

We express our sincere gratitude to the Pan African University, Institute for Basic Sciences, Technology, and Innovation (PAUSTI), Nairobi, Kenya, and the African Union for providing the essential resources that made this research possible. Their support and encouragement were instrumental in enabling the successful completion of this work.

References

[1] S. K. Singh, R. Singh, and A. Goyal, "Semi-automatic segmentation of overlapping cells in pap smear image," in *Proceedings of the 4th International Conference on Computer Science (ICCS)*, Aug. 2018, pp. 161–165. https://doi.org/10.1109/ICCS.2018.00034

[2] B. Nithya and V. Ilango, "Evaluation of machine learning based optimized feature selection approaches and classification methods for cervical cancer prediction," *SN Applied Sciences*, vol. 1, no. 6, p. 641, 2019. doi: 10.1007/s42452-019-0645-7. [Online]. Available: https://doi.org/10.1007/s42452-019-0645-7

[3] M. Arbyn, E. Weiderpass, L. Bruni, S. de Sanjosé, M. Saraiya, J. Ferlay, and F. Bray, "Estimates of incidence and mortality of cervical cancer in 2018: a worldwide analysis," *The Lancet Global Health*, vol. 8, no. 2, pp. e191–e203, 2020. doi: 10.1016/S2214-109X(19)30482-6. [Online]. Available: https://doi.org/10.1016/S2214-109X(19)30482-6

[4] J. Jin, "HPV Infection and Cancer," *JAMA*, vol. 319, no. 10, p. 1058, 2018. doi: 10.1001/jama.2018.0687. [Online]. Available: https://doi.org/10.1001/jama.2018.0687

[5] M. Das, "WHO launches strategy to accelerate elimination of cervical cancer," *The Lancet Oncology*, vol. 22, no. 1, pp. 20–21, 2021. doi: 10.1016/S1470-2045(20)30729-4. [Online]. Available: https://doi.org/10.1016/S1470-2045(20)30729-4

[6] N. Youneszade, M. Marjani, and C. P. Pei, "Deep Learning in Cervical Cancer Diagnosis: Architecture, Opportunities, and Open Research Challenges," *IEEE Access*, vol. 11, pp. 6133–6149, 2023. https://doi.org/10.1109/ACCESS.2023.3235833

[7] J. Wu, Q. Jin, Y. Zhang, Y. Ji, J. Li, X. Liu, and Y. Huang, "Global Burden of Cervical Cancer: Current Estimates, Temporal Trend and Future Projections Based on the GLOBOCAN 2022," *Journal of the National Cancer Center*, 2025. doi: 10.1016/j.jncc.2024.11.006. [Online]. Available: https://doi.org/10.1016/j.jncc.2024.11.006

[8] P. N. Simelela, "WHO Global Strategy to Eliminate Cervical Cancer as a Public Health Problem: An Opportunity to Make It a Disease of the Past," *International Journal of Gynecology & Obstetrics*, vol. 152, pp. 1–3, 2021. doi: 10.1002/ijgo.13484. [Online]. Available: https://doi.org/10.1002/ijgo.13484

[9] K. T. Simms, J. Steinberg, M. Caruana, M. A. Smith, J. B. Lew, I. Soerjomataram, *et al.*, "Impact of scaled up human papillomavirus vaccination and cervical screening and the potential for global elimination of cervical cancer in 181 countries, 2020–99: a modelling study," *The Lancet Oncology*, vol. 20, no. 3, pp. 394–407, 2019. https://doi.org/10.1016/S1470-2045(18)30836-2

[10] X. Hou, G. Shen, L. Zhou, Y. Li, T. Wang, and X. Ma, "Artificial intelligence in cervical cancer screening and diagnosis," *Frontiers in Oncology*, vol. 12, p. 851367, 2022. https://doi.org/10.3389/fonc.2022.851367

[11] V. Chandran, M. G. Sumithra, A. Karthick, T. George, M. Deivakani, B. Elakkiya, *et al.*, "Diagnosis of cervical cancer based on ensemble deep learning network using colposcopy images," *BioMed Research International*, vol. 2021, no. 1, p. 5584004, 2021. doi: 10.1155/2021/5584004. [Online]. Available: https://doi.org/10.1155/2021/5584004

[12] G. Purwoto, H. D. Dianika, A. Putra, S. Purbadi, and L. Nuranna, "Modified cervicography and visual inspection with acetic acid as an alternative screening method for cervical precancerous lesions," *Journal of Cancer Prevention*, vol. 22, no. 4, p. 254, 2017. https://doi.org/10.15430/JCP.2017.22.4.254

[13] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. doi: 10.1038/nature14539. [Online]. Available: https://doi.org/10.1038/nature14539

[14] P. Sahoo, S. Saha, S. Mondal, M. Seera, S. K. Sharma, and M. Kumar, "Enhancing Computer-Aided Cervical Cancer Detection Using a Novel Fuzzy Rank-Based Fusion," *IEEE Access*, vol. 11, pp. 145281–145294, 2023. doi: 10.1109/ACCESS.2023.3346764. [Online]. Available: https://doi.org/10.1109/ACCESS.2023.3346764

[15] A. Khan, Z. Rauf, A. Sohail, A. R. Khan, H. Asif, A. Asif, and U. Farooq, "A Survey of the Vision Transformers and their CNN-Transformer Based Variants," *Artificial Intelligence Review*, vol. 56, no. Suppl 3, pp. 2917–2970, 2023. doi: 10.1007/s10462-023-10595-0. [Online]. Available: https://doi.org/10.1007/s10462-023-10595-0

[16] M. Kalbhor, S. Shinde, H. Joshi, and P. Wajire, "Pap smear-based cervical cancer detection using hybrid deep learning and performance evaluation," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization*, vol. 11, no. 5, pp. 1615–1624, 2023. https://doi.org/10.1080/21681163.2023.2219469

[17] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 6999–7019, 2021. doi: 10.1109/TNNLS.2021.3084827. [Online]. Available: https://doi.org/10.1109/TNNLS.2021.3084827

[18] I. Zimerman and L. Wolf, "On the Long Range Abilities of Transformers," *arXiv preprint arXiv:2311.16620*, 2023. doi: 10.48550/arXiv.2311.16620. [Online]. Available: https://doi.org/10.48550/arXiv.2311.16620

[19] M. M. Naseer, K. Ranasinghe, S. H. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Intriguing Properties of Vision Transformers," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 23296–23308. doi: 10.48550/arXiv.2105.10497. [Online]. Available: https://doi.org/10.48550/arXiv.2105.10497

[20] Y. Dai, F. Gieseke, S. Oehmcke, Y. Wu, and K. Barnard, "Attentional Feature Fusion," in *Proc. IEEE/CVF Winter Conf. on Applications of Computer Vision (WACV)*, 2021, pp. 3560–3569. doi: 10.1109/WACV48630.2021.00360. [Online]. Available: https://doi.org/10.1109/WACV48630.2021.00360

[21] J. Wei, S. Wang, and Q. Huang, "F³Net: Fusion, Feedback and Focus for Salient Object Detection," in *Proc. AAAI Conf. on Artificial Intelligence*, vol. 34, no. 07, 2020, pp. 12321–12328. doi: 10.1609/aaai.v34i07.6916. [Online]. Available: https://doi.org/10.1609/aaai.v34i07.6916

[22] F. A. Mohammed, K. K. Tune, J. A. Mohammed, T. A. Wassu, and S. Muhie, "Early Cervical Cancer Diagnosis with SWIN-Transformer and Convolutional Neural Networks," *Diagnostics*, vol. 14, no. 20, p. 2286, 2024. doi: 10.3390/diagnostics14202286. [Online]. Available: https://doi.org/10.3390/diagnostics14202286

[23] L. Wang, C. Zhang, and J. Li, "A Hybrid CNN-Transformer Model for Predicting N Staging and Survival in Non-Small Cell Lung Cancer Patients Based on CT-Scan," *Tomography*, vol. 10, no. 10, pp. 1676–1693, Oct. 2024. doi: 10.3390/tomography10100123. [Online]. Available: https://doi.org/10.3390/tomography10100123

[24] P. Chen, F. Liu, J. Zhang, and B. Wang, "MFEM-CIN: a lightweight architecture combining CNN and transformer for the classification of Pre-cancerous lesions of the cervix," *IEEE Open Journal of Engineering in Medicine and Biology*, vol. 5, pp. 216–225, 2024. doi: 10.1109/OJEMB.2024.3367243. [Online]. Available: https://doi.org/10.1109/OJEMB.2024.3367243

[25] W. Liu, C. Li, N. Xu, T. Jiang, M. M. Rahaman, H. Sun, and M. Grzegorzek, "CVM-Cervix: A hybrid cervical Pap-smear image classification framework using CNN, visual transformer and multilayer perceptron," *Pattern Recognition*, vol. 130, p. 108829, 2022. https://doi.org/10.1016/j.patcog.2022.108829.

[26] R. Dulal, L. Zheng, and M. A. Kabir, "MHAFF: Multihead Attention Feature Fusion of CNN and Transformer for Cattle Identification," *IEEE Transactions on AgriFood Electronics*, 2025. doi: 10.1109/TAFE.2025.3574708. [Online]. Available: https://doi.org/10.1109/TAFE.2025.3574708

[27] J. Hu, L. Shen, and G. Sun, "Squeeze-and-Excitation Networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 7132–7141. https://doi.org/10.48550/arXiv.1709.01507.

[28] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19. https://doi.org/10.48550/arXiv.1807.06521.

[29] M. Tan and Q. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 6105–6114. https://doi.org/10.48550/arXiv.1905.11946.

[30] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2021, pp. 10012–10022. https://doi.org/10.1109/ICCV48922.2021.00986.

[31] International Agency for Research on Cancer, "Colposcopy Image Bank," 2024. [Online]. Available: https://screening.iarc.fr/cervicalimagebank.php. Accessed: May 17, 2025.

[32] S. Nursyafiqah, "Dataset: Colposcopy Images," Kaggle, 2025. [Online]. Available: https://www.kaggle.com/datasets/sitinursyafiqahba/dataset-colposcopy-images.

[33] C. Shorten and T. M. Khoshgoftaar, "A Survey on Image Data Augmentation for Deep Learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, 2019. https://doi.org/10.1186/s40537-019-0197-0.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017. https://doi.org/10.1145/3065386.

[35] A. Vaswani *et al.*, "Attention Is All You Need," arXiv preprint arXiv:1706.03762, 2017. https://doi.org/10.48550/arXiv.1706.03762.

[36] G. Kumar and P. K. Bhatia, "A Detailed Review of Feature Extraction in Image Processing Systems," in *2014 Fourth International Conference on Advanced Computing & Communication Technologies (ACCT)*, February 2014, pp. 5–12, IEEE. doi: 10.1109/ACCT.2014.74.

[37] S. Khalid, T. Khalil, and S. Nasreen, "A Survey of Feature Selection and Feature Extraction Techniques in Machine Learning," in *2014 Science and Information Conference*, 2014, pp. 372–378, IEEE. https://doi.org/10.1109/SAI.2014.6918213. [Online]. Available:https://doi.org/10.1109/SAI.2014.6918213

[38] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 618–626, 2017. doi: 10.1109/ICCV.2017.74. [Online]. Available: https://doi.org/10.1109/ICCV.2017.74

[39] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," in *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, New Orleans, LA, USA, May 6–9, 2019. [Online]. Available: https://dblp.org/rec/conf/iclr/LoshchilovH19.html

[40] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017. doi: 10.1016/j.media.2017.07.005.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.48550/arXiv.1512.03385. [Online]. Available: https://doi.org/10.48550/arXiv.1512.03385

[42] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4700–4708, 2017. doi: 10.48550/arXiv.1608.06993. [Online]. Available: https://doi.org/10.48550/arXiv.1608.06993

[43] O. S. Oyebanji, A. R. Apampa, P. I. Idoko, A. Babalola, O. M. Ijiga, O. Afolabi, and C. I. Michael, "Enhancing Breast Cancer Detection Accuracy through Transfer Learning: A Case Study Using EfficientNet," *World Journal of Advanced Engineering Technology and Sciences*, vol. 13, no. 01, pp. 285–318, 2024. doi: 10.30574/wjaets.2024.13.1.0415. [Online]. Available: https://doi.org/10.30574/wjaets.2024.13.1.0415

[44] M. Behzadpour, B. L. Ortiz, E. Azizi, and K. Wu, "Breast Tumor Classification Using EfficientNet Deep Learning Model," *arXiv preprint arXiv:2411.17870*, 2024. doi: 10.48550/arXiv.2411.17870. [Online]. Available: https://doi.org/10.48550/arXiv.2411.17870

[45] A. A. Nafea, M. S. Ibrahim, M. M. Shwaysh, K. Abdul-Kadhim, H. R. Almamoori, and M. M. Al-Ani, "A Deep Learning Algorithm for Lung Cancer Detection Using EfficientNet-B3," *Wasit Journal of Computer and Mathematics Science*, vol. 2, no. 4, pp. 68–76, 2023. doi: 10.31185/wjcms.209. [Online]. Available: https://doi.org/10.31185/wjcms.209

[46] S. Aouadi, T. Torfeh, O. Bouhali, S. A. Yoganathan, S. Paloor, S. Chandramouli, and N. Al-Hammadi, "Prediction of Cervix Cancer Stage and Grade from Diffusion Weighted Imaging Using EfficientNet," *Biomedical Physics & Engineering Express*, vol. 10, no. 4, p. 045042, 2024. doi: 10.1088/2057-1976/ad5207. [Online]. Available: https://doi.org/10.1088/2057-1976/ad5207

[47] A. Suphalakshmi, A. Ahilan, A. Jeyam, and M. Subramanian, "Cervical Cancer Classification Using EfficientNet and Fuzzy Extreme Learning Machine," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 5, pp. 6333–6342, 2022. doi: 10.3233/JIFS-220296. [Online]. Available: https://doi.org/10.3233/JIFS-220296

[48] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. doi: 10.48550/arXiv.2010.11929. [Online]. Available: https://doi.org/10.48550/arXiv.2010.11929

[49] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, *et al.*, "A survey on vision transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 87–110, 2022. doi: 10.1109/TPAMI.2022.3152247.

[50] J. Zhang, Z. Zhang, H. Liu, and S. Xu, "SaTransformer: Semantic-aware transformer for breast cancer classification and segmentation," *IET Image Processing*, vol. 17, no. 13, pp. 3789–3800, 2023. doi: 10.1049/ipr2.12897. [Online]. Available: https://doi.org/10.1049/ipr2.12897

[51] D. R. Nayak, "RDTNet: A residual deformable attention based transformer network for breast cancer classification," *Expert Systems with Applications*, vol. 249, p. 123569, 2024. doi: 10.1016/j.eswa.2024.123569. [Online]. Available: https://doi.org/10.1016/j.eswa.2024.123569

[52] D. P. Yadav, B. Sharma, J. L. Webber, A. Mehbodniya, and S. Chauhan, "EDTNet: A spatial aware attention-based transformer for the pulmonary nodule segmentation," *PLOS ONE*, vol. 19, no. 11, p. e0311080, 2024. doi: 10.1371/journal.pone.0311080. [Online]. Available: https://doi.org/10.1371/journal.pone.0311080

[53] M. Darwish, M. Z. Altabel, and R. H. Abiyev, "Enhancing cervical precancerous classification using advanced vision transformer," *Diagnostics*, vol. 13, no. 18, p. 2884, 2023. doi: 10.3390/diagnostics13182884. [Online]. Available: https://doi.org/10.3390/diagnostics13182884

[54] N. Sharma, K. Gaurav, and T. K. R. Bollu, "CerviTransX: Explainable Transformer-Based Cervical Cancer Classification," in *Proceedings of the 2025 National Conference on Communications (NCC)*, pp. 1–6, Mar. 2025. doi: 10.1109/NCC63735.2025.10983448. [Online]. Available: https://doi.org/10.1109/NCC63735.2025.10983448

[55] K. Abinaya and B. Sivakumar, "A Deep Learning-Based Approach for Cervical Cancer Classification Using 3D CNN and Vision Transformer," *Journal of Imaging Informatics in Medicine*, vol. 37, no. 1, p. 280, 2024. doi: 10.1007/s10278-023-00911-z. [Online]. Available: https://doi.org/10.1007/s10278-023-00911-z

Optimizing Resource Allocation for Crisis-Resilient Healthcare Robotics: An Integrated MLR, MDP, and Petri Net Approach

Ikram Dahamou[1]*, Ayoub Elbazzazi[2], Cherki Daoui[3]
Faculty of Science and Technology, Computer Engineering,
Sultan Moulay Slimane University, B.P. 523, Beni Mellal, Morocco[1,2]
Faculty of Science and Technology, Mathematics and Computer Science
Sultan Moulay Slimane University, B.P. 523, Beni Mellal, Morocco[3]

***Abstract*—Global crises, such as pandemics and climate-related disasters, place unprecedented strain on healthcare systems, exposing weaknesses in resource management and patient care. This study aims to address these challenges by developing an integrated computational framework for crisis-resilient healthcare robotics. We propose a unified approach that combines Multiple Linear Regression (MLR), Markov Decision Processes (MDPs), and Petri Nets. MLR is applied to predict the Average Length of Stay (ALOS) using patient and hospital data. These forecasts inform MDPs, which guide admission and triage decisions under uncertainty. Petri Nets are employed to model and validate patient flow and hospital workflows, ensuring feasibility and efficiency. Case studies, including ICU bed prioritization and disaster logistics, demonstrate that the proposed framework improves adaptability and resource utilization while supporting structured decision guidance. Simulation results highlight enhanced system efficiency, better patient prioritization, and reduced congestion during surge conditions. The integration of predictive analytics, probabilistic optimization, and workflow modeling provides a robust decision-support system for healthcare robotics in crisis scenarios. This interdisciplinary framework offers practical solutions for improving resilience, scalability, and patient outcomes, providing a structured foundation for enhancing resilience and coordination in healthcare systems facing future emergencies.**

***Keywords*—Healthcare robotics; Markov Decision Processes; Petri Nets; Multiple Linear Regression; crisis management; ICU resource allocation**

I. Introduction

Global healthcare systems are increasingly challenged by recurring crises such as pandemics, extreme weather events, and large-scale disasters [1], [2]. These events expose persistent weaknesses in continuity of care, inefficiencies in resource management, and risks to patient outcomes.

A. Motivation and Contribution Perspective

Rather than introducing new individual algorithms, this work focuses on how existing predictive, decision, and workflow models can be structurally combined to support crisis-aware hospital operations. The core contribution lies in a decision-oriented modeling chain (MLR $\rightarrow$ MDP $\rightarrow$ Petri Net), which constrains hospital dynamics to decision-feasible trajectories. This structured coupling enables tractable reachability analysis and formal verification of admission policies under severe resource constraints, which cannot be achieved using predictive or decision models alone.

To address this gap, we propose an integrated computational framework that combines:

- Multiple Linear Regression (MLR) for forecasting critical indicators such as the Average Length of Stay (ALOS) [5], [6],
- Markov Decision Processes (MDPs) for guiding triage and admission policies under uncertainty [7], [8],
- Petri Nets for validating hospital workflows and ensuring feasibility of robotic interventions under operational constraints [10], [11].

This unified approach integrates statistical forecasting, probabilistic decision-making, and workflow simulation into a single coherent system.

B. Main Contributions

The main contributions of this work are summarized as follows:

- We propose a decision-constrained modeling chain that links predictive analytics, sequential decision-making, and workflow verification, reducing the reachable state space of hospital systems during crises.
- We introduce an explainable approximate MDP-based admission policy, designed for environments where full transition probabilities cannot be reliably identified.
- We demonstrate how Petri Net reachability analysis can be used to formally verify the feasibility, boundedness, and safety of data-driven admission decisions.

C. Paper Organization

The remainder of this study is organized as follows:

- Section II reviews related work on healthcare robotics and computational modeling, discusses applications

*Corresponding author.

and current limitations of robotic systems in healthcare, presents the proposed improvements and analyzes resource management during global crises.

- Section III details the computational models (MLR, MDPs, Petri Nets).
- Section IV provides case analysis and intermediate results.
- Section V develops Petri Net modeling and analysis.
- Section VI defines the initial marking and reachability analysis.
- Section VII describes integration with robotic systems through an illustrative workflow and offers a comparative analysis of reachability approaches.
- Section VIII concludes the study with perspectives for future research.

II. Related Work

The integration of computational models in healthcare robotics has attracted increasing attention in recent years. Several studies have applied Markov Decision Processes (MDPs) to model clinical decision-making under uncertainty, particularly for triage optimization and ICU resource allocation [7], [8]. For example, Hauskrecht (2000) demonstrated the use of MDPs in managing treatment policies for chronic diseases. More recent works have explored reinforcement learning and optimization-based strategies for hospital resource allocation under pandemic-induced surge conditions [24], [25], [26], highlighting the importance of adaptive decision-making in crisis scenarios.

Petri Nets have been widely used to model complex hospital workflows, including patient flow in emergency departments and coordination of clinical tasks [10], [11], [17]. They provide powerful tools for identifying bottlenecks and ensuring safety constraints in distributed systems. Recent probabilistic and time-dependent Petri Net extensions have further enabled formal modeling of clinical pathways under uncertainty [27].

Multiple Linear Regression (MLR) has been applied for forecasting hospital metrics such as length of stay (LOS), readmission risks, and cost estimation [5]. However, these studies generally focus on prediction as a standalone task and do not embed regression outputs into real-time robotic or autonomous decision workflows.

In contrast to prior works that treat predictive modeling, decision optimization, and workflow analysis as separate components, the present framework integrates MLR, MDPs, and Petri Nets within a unified decision-support modeling chain. While existing literature addresses forecasting, optimization, or workflow verification independently, relatively few approaches structurally couple predictive analytics with sequential decision-making and formal system validation.

This methodological gap motivates the integrated approach proposed in this study, where predictive estimates inform admission control policies, and decision-constrained transitions are subsequently verified through Petri Net reachability analysis.

Petri Nets have been widely used to model complex hospital workflows, including patient flow in emergency departments and coordination of clinical tasks [10], [11]. They provide powerful tools for identifying bottlenecks and ensuring safety constraints in distributed systems. Recent probabilistic and time-dependent Petri Net extensions have further enabled formal modeling of clinical pathways under uncertainty [27]. Recent Petri Net models have also been applied to healthcare systems under COVID-19 surge conditions [18].

A. Main Applications of Healthcare Robotics

- Surgical Robots: Systems like *da Vinci* enable minimally invasive surgery with higher precision and reduced recovery time [15].
- Assistive Robots: Devices such as PARO or Robear help elderly and disabled patients with daily activities and emotional support [12].
- Diagnostic and Rehabilitation Robots: AI-based robots analyze medical images or assist physical recovery, improving outcomes [22], [23].
- Logistics Support: Autonomous systems (e.g., TUG robots) deliver drugs and supplies, increasing hospital efficiency [1].

B. Challenges in Crisis Contexts

Despite these advances, crises such as pandemics expose key weaknesses:

- Resource rigidity: Robots are often task-specific and struggle to adapt to new priorities [3], [4].
- Limited real-time adaptation: Pre-programmed workflows cannot manage sudden surges in patient volume [8].
- Poor integration: Robotic subsystems rarely connect seamlessly to hospital information systems or triage networks [10].
- High cost and scalability issues: Advanced units remain inaccessible to resource-limited hospitals [2].

C. Resource Management Under Global Crises

Pandemics and disasters further aggravate shortages of ICU beds, staff, and equipment [19], [5]. Traditional allocation approaches remain static, manually driven, and fragmented, lacking predictive and adaptive capacity [7], [2]. Integrating data-driven models such as Petri Nets, Markov Decision Processes, and Multiple Linear Regression can improve crisis resource allocation [20].

- Real-time decision-making and prioritization of tasks,
- Interoperability with hospital information systems,
- Scalability and coordination during resource surges.

This hybrid framework promotes proactive, self-adjusting robotic systems capable of supporting hospitals under both routine and emergency conditions. The constraints and challenges outlined in this section motivate the need for an integrated decision-support framework capable of anticipating resource usage, regulating admissions, and ensuring operational safety.

III. Computational Models for Crisis-Aware Optimization

This work relies on the integration of three complementary modeling paradigms—Multiple Linear Regression (MLR), Markov Decision Processes (MDPs), and Petri Nets—to support crisis-aware decision-making in hospital systems. Each model plays a distinct and clearly delimited role: prediction, decision guidance, and formal workflow validation.

A. Predictive Layer: Multiple Linear Regression for ALOS Estimation

Multiple Linear Regression (MLR) is used to forecast the Average Length of Stay (ALOS), a key operational indicator for hospital capacity planning [5]. Accurate ALOS prediction enables anticipation of future bed occupancy and resource commitment, which is critical during crisis situations such as pandemics or mass-casualty events [6].

1) Model formulation: The MLR model estimates ALOS as:

$$Y = \beta_0 + \sum_{i=1}^{n} \beta_i X_i + \epsilon \tag{1}$$

where, Y denotes ALOS, X_i represent patient- and admission-related variables, β_i are regression coefficients, and ϵ is the error term.

2) Why Multiple Linear Regression?: MLR is selected as a transparent baseline that supports interpretability and auditability in healthcare operations. Unlike complex nonlinear predictors, MLR allows direct inspection of coefficients, facilitates sensitivity analysis, and reduces integration friction when predictions are embedded into the downstream decision layer.

Although nonlinear models may achieve marginally higher predictive accuracy, the primary objective of this work is to ensure explainability and seamless coupling with the decision-theoretic layer. Therefore, MLR is adopted as a structurally compatible and interpretable predictive backbone.

3) Data and predictive performance: The model is trained on data derived from the MIMIC-III database [6], including demographic information, clinical severity, emergency admission rate, satisfaction score, and total hospital charges.

4) Preprocessing and reproducibility details: We used MIMIC-III structured variables consistent with ALOS modeling. Missing values were handled by: 1) removing features with $> 30\%$ missingness, and 2) median imputation for continuous variables and mode imputation for categorical variables. Continuous predictors were standardized using z-score normalization: $x' = (x - \mu)/\sigma$ computed on the training set only. Categorical variables were one-hot encoded.

Feature selection followed a two-step procedure: a) correlation screening and variance inflation factor (VIF) control to mitigate multicollinearity, and b) stepwise regression using AIC to retain parsimonious predictors. We report the final set of predictors in Table I.

TABLE I. Key Predictors Impacting Average Length of Stay (ALOS)

Variable	Coef.	Interpretation
Emergency_Rate	+0.043	Increases ALOS
Patient_Severity_Index	+0.037	Prolongs stay
Satisfaction_Score	-0.031	Reduces stay duration
Total_Hospital_Charges	+0.022	Associated with longer stay

We evaluated the model using 5-fold cross-validation and report the average values of R^2, RMSE, and MAE across folds. The coefficient of determination ($R^2 = 0.86$) indicates that 86% of the variability in ALOS is explained by the model. The RMSE of 0.38 days reflects limited quadratic prediction error, while the MAE of 0.48 days suggests that the average absolute deviation between predicted and observed ALOS remains below half a day, indicating stable and clinically acceptable predictive performance.

The feature correlations are illustrated in Fig. 1.

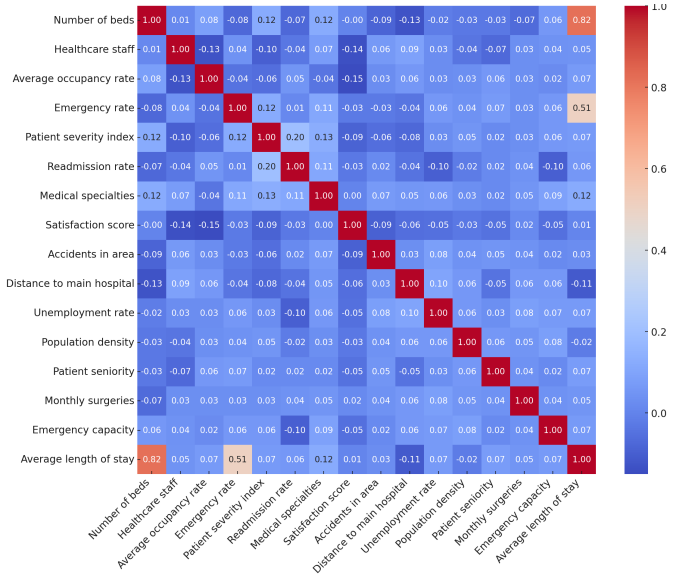


Fig. 1. Correlation matrix between input features and ALOS.

5) Operational role of MLR: Rather than acting as a decision-maker, MLR provides anticipatory information that feeds the downstream decision layer. Predicted ALOS values quantify expected resource occupation and are explicitly embedded into the MDP state representation.

B. Decision Layer: Crisis-Aware Admission Control via MDP

The Markov Decision Process (MDP) is not used to compute an optimal policy in the classical sense. Instead, it represents a constrained, explainable approximation of an optimal MDP policy, suitable for crisis contexts where transition probabilities are uncertain or rapidly evolving [7], [8].

1) State and action representation: Each MDP state captures the current operational condition of the hospital, including bed availability, ICU load, patient severity, and predicted ALOS from the MLR model. The action space is deliberately

restricted to three clinically meaningful decisions: *admit*, *wait-list*, or *reject*.

2) Formal MDP definition and coupling with MLR: We model admission control as a Markov Decision Process (MDP) defined by the tuple $M = (S, A, P, R, \gamma)$, following the classical MDP formulation [9]. Each state $s_t \in \mathcal{S}$ summarizes the hospital operational condition at decision time t:

$$s_t = \left(b_t, o_t, \bar{a}_t, \hat{\ell}_t, \sigma_{\hat{\ell},t}, u_t\right) \tag{2}$$

where, b_t is the number of available beds, o_t is the ICU occupancy level, $\bar{a}_t$ denotes arrival pressure (e.g., emergency rate), $\hat{\ell}_t$ is the MLR-predicted ALOS for the incoming patient (or cohort), $\sigma_{\hat{\ell},t}$ is the prediction uncertainty (estimated from residual variance), and u_t denotes patient acuity/severity.

The action space is $\mathcal{A} = \{\textit{Admit}, \textit{Waitlist}, \textit{Reject}\}$.

Transition dynamics are expressed as:

$$\mathcal{P}(s_{t+1} \mid s_t, a_t) \tag{3}$$

where, bed availability evolves according to admissions/discharges:

$$b_{t+1} = b_t - \mathbb{I}[a_t = \textit{Admit}] + d_t \tag{4}$$

and d_t represents discharges, approximated using the predicted LOS distribution induced by $\hat{\ell}_t$.

The reward function encodes clinical utility and congestion penalties:

$$\begin{aligned}\mathcal{R}(s_t, a_t) = \alpha u_t \mathbb{I}[a_t = \textit{Admit}] - \beta \mathbb{I}[b_t = 0 \wedge a_t = \textit{Admit}] \\ - \gamma_1 \hat{\ell}_t \mathbb{I}[a_t = \textit{Admit}] - \gamma_2 \sigma_{\hat{\ell},t} \mathbb{I}[a_t = \textit{Admit}]\end{aligned} \tag{5}$$

where, $\alpha, \beta, \gamma_1, \gamma_2 \geq 0$ balance severity prioritization, overflow risk, expected resource occupation, and robustness to prediction uncertainty.

3) Decision logic and interpretation: The reward structure reflects institutional priorities rather than numerical optimality. Admissions associated with long predicted stays are penalized under high occupancy, while high-severity patients are prioritized. As a result, the derived policy acts as a decision-guidance mechanism that balances urgency and sustainability.

4) Illustrative admission decisions: Algorithm 1 presents a simplified rule-based representation of the MDP-derived logic, introduced for interpretability only.

Algorithm 1 MDP-Based Admission Policy

Require: CurrentBeds, TotalBeds,
SeverityIndex, PredictedALOS
Ensure: AdmissionDecision
if CurrentBeds $\geq$ TotalBeds **then**
  AdmissionDecision $\leftarrow$ Reject
else
  if SeverityIndex ≥ 8 **then**
    AdmissionDecision $\leftarrow$ Admit
  else if PredictedALOS ¡ 5 **then**
    AdmissionDecision $\leftarrow$ Admit
  else
    AdmissionDecision $\leftarrow$ Waitlist
  end if
end if
return AdmissionDecision

Representative admission outcomes are reported in Table II, illustrating how severity and predicted ALOS jointly shape decisions under constrained capacity.

TABLE II. MDP DECISIONS BASED ON PRIORITY SCORE COMBINING SEVERITY INDEX AND PREDICTED ALOS.

Patient	Severity Index	Predicted ALOS	Priority Score	MDP Decision
P1	7	8.29	9.86	Waitlist
P2	4	5.98	5.01	Waitlist
P3	8	4.46	13.77	Admit
P4	5	6.97	6.51	Waitlist
P5	7	3.22	12.39	Admit
P6	3	8.90	1.55	Reject
P7	7	6.15	10.93	Admit
P8	8	5.77	13.12	Admit
P9	5	9.49	5.26	Waitlist
P10	4	8.09	3.95	Reject

C. Formal Validation Layer: Petri Net Modeling

Petri Nets are used as a formal verification and simulation layer to validate the feasibility and safety of MDP-driven decisions within hospital workflows [10], [11].

1) Reachability analysis: A Petri Net is defined as:

$$PN = (P, T, F, W, M_0)$$

and its reachability graph enumerates all possible system states reachable from the initial marking [13], [14]. Reachability analysis enables the verification of critical properties such as deadlock-freeness, boundedness, and guaranteed admission of high-priority patients. This positions Petri Nets as the scientific validation core of the proposed framework.

D. Methodological Complementarity

In summary:

- MLR anticipates future resource demand,
- The MDP provides explainable decision guidance under uncertainty,
- Petri Nets formally validate workflow consistency.

This layered architecture ensures predictive, decision-oriented, and formally verified crisis-aware optimization.

IV. Comparative Example Analysis of Decision-Aware and Direct Petri Net Modeling

This section presents a comparative example analysis between two hospital workflow modeling approaches under crisis conditions, highlighting the analytical benefits of introducing a decision layer prior to Petri Net construction.

A. Method A: Direct Petri Net Modeling

In the baseline approach, the hospital workflow is modeled directly as a Petri Net without explicit decision constraints. Patient arrivals, admissions, treatments, and discharges are represented as transitions, while hospital resources are modeled as bounded places.

The initial marking is defined as:

$$M_0 = (1, 0, 0, 0, 0, 0),$$

Indicating that one patient enters the system while all other places are initially empty. By firing all enabled transitions, a corresponding reachability graph is obtained. Although this approach accurately captures workflow dynamics, it generates a large and weakly constrained state space that may include clinically unrealistic configurations.

B. Method B: MLR $\rightarrow$ PDM $\rightarrow$ Petri Net Modeling (Proposed)

The proposed approach introduces a decision layer prior to Petri Net construction. Admission decisions are regulated by the Resource Logic Model (RLM) and the Process Decision Model (PDM), derived from the MDP framework and informed by predicted ALOS. Only policy-compliant decisions are translated into Petri Net transitions.The resulting decision-aware Petri Net structure is shown in Fig. 2.

In particular, transitions from waitlist to admission are enabled exclusively when priority and resource availability conditions are satisfied.

C. Reachability Graph Derived from the Decision-Aware Model

The reachability graph associated with the proposed modeling chain is shown in Fig. 3. Each node corresponds to a feasible system configuration, while edges represent authorized transition firings. Compared to direct Petri Net modeling, the reachability graph is significantly more compact and structured, containing only clinically admissible and resource-feasible states.

D. Comparative Discussion

The comparison highlights three major advantages of the proposed approach:

- Controlled State-Space Complexity: Decision constraints limit unnecessary reachability expansion.
- Decision Alignment: Petri Net execution remains consistent with MDP-based admission priorities.
- Formal Verifiability: Boundedness, liveness, and deadlock-freeness are easier to establish.

E. Interpretation and Practical Implications

This comparative example confirms that integrating decision intelligence prior to Petri Net modeling produces a more interpretable and analytically tractable system. The proposed modeling chain preserves workflow realism while preventing state-space explosion, making it particularly suitable for crisis-aware healthcare management.

F. Computational Complexity and Runtime Considerations

The predictive layer (MLR) performs inference in $O(pd)$ per time step, where p denotes the number of predictors and d the number of patients scored at that step. The decision layer relies on a compact discrete-state representation; under the proposed rule-based approximation, policy evaluation is $O(1)$ per patient decision.

Petri Net reachability analysis is exponential in the worst case. To mitigate state-space explosion, we constrain both the initial marking and the set of policy-compliant transitions, thereby limiting the number of reachable states.

We report runtimes measured on a standard workstation for: 1) MLR inference per patient, 2) decision computation per patient, and 3) reachability graph construction for the considered markings. Finally, we discuss scalability to multi-department settings by decomposing the global Petri Net into interacting subnetworks and evaluating departments in parallel.

V. Petri Net Modeling and Analysis

Fig. 4 presents a Petri Net model designed to represent patient flow and hospital resource management under global crisis conditions. The model follows the classical Petri Net formalism, which has been widely applied to healthcare process modeling [13], [14], [16].

A. Places and Transitions

Places represent critical system states and hospital resources, including Emergency Capacity, Emergency Rate, Personnel, Beds, Operations, and Satisfaction. Each place captures the availability or status of a resource or patient group at a given time.

Transitions correspond to operational events that modify the system state, such as Emergency Visits, Admission, Treatment, Operations, and Evaluation. Transition firings consume and generate tokens, which represent patients or resources flowing through the hospital system.

The model dynamics explicitly encode causal dependencies:

- Admission depends on emergency capacity, emergency rate, population density, patient age, accident zone proximity, and distance to the hospital, reflecting a multi-criteria admission decision process.
- Treatment requires the simultaneous availability of beds and medical personnel.
- Patients requiring surgery pass through the Operations transition, mobilizing specialized resources.

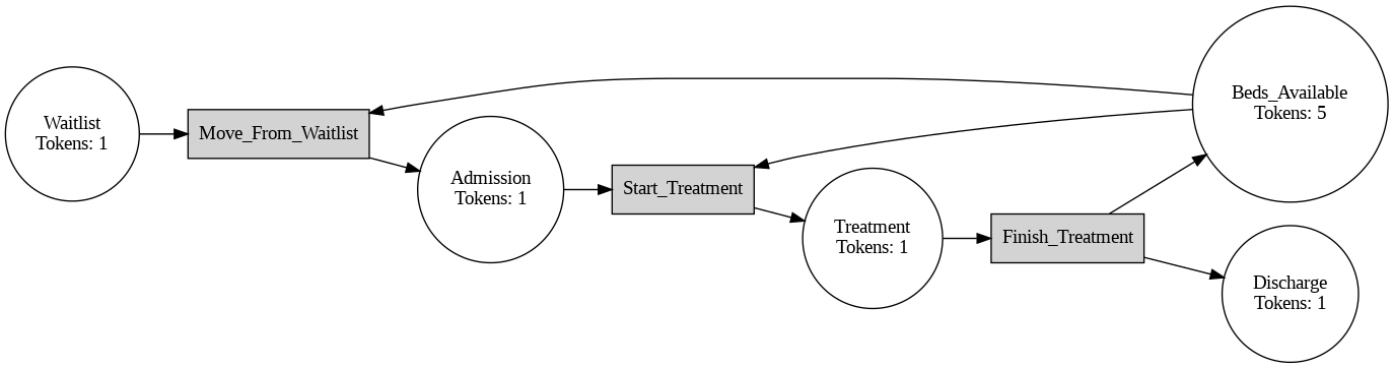


Fig. 2. Petri Net representation of patient flow derived from PDM and MDP policy.

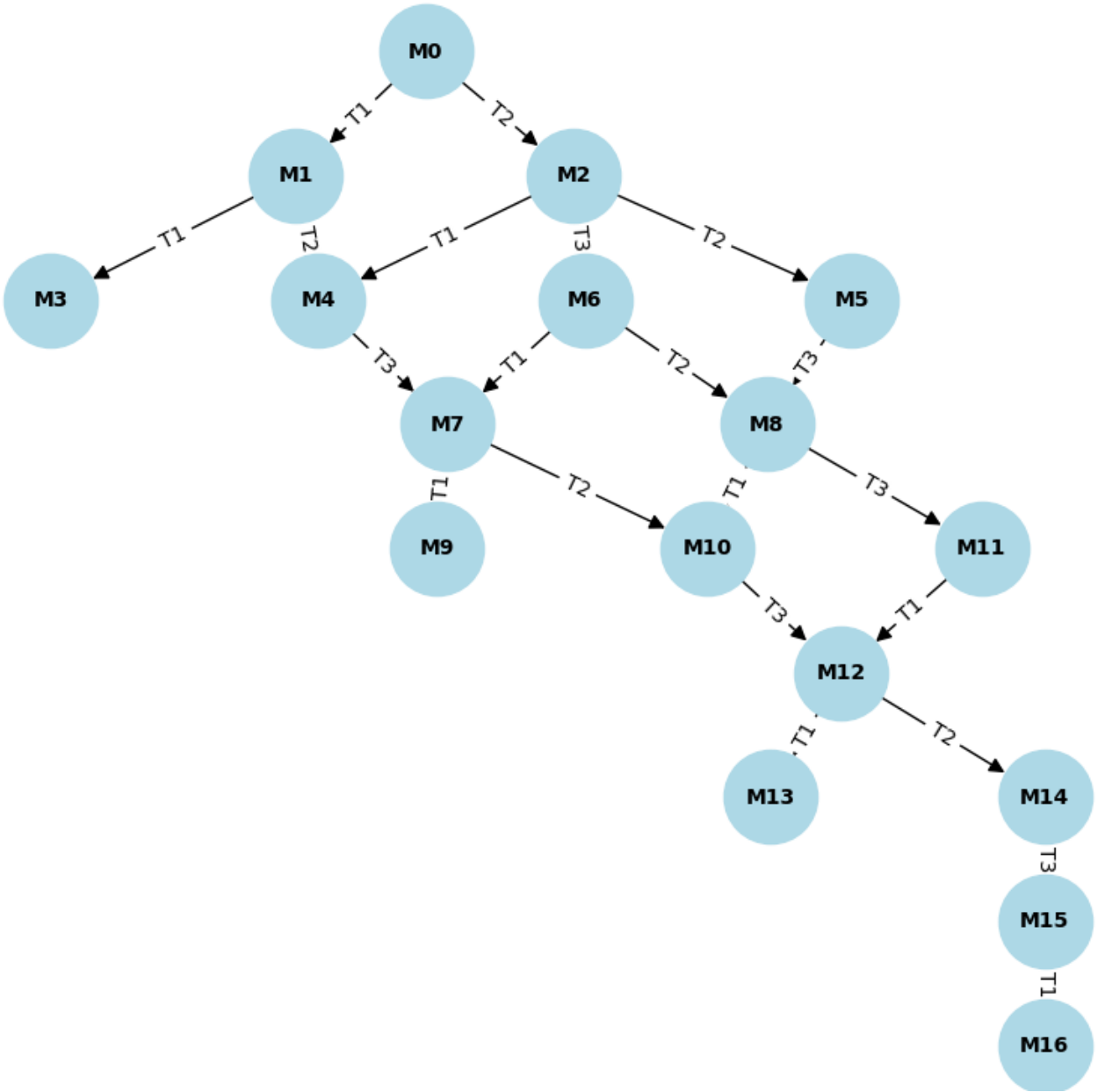


Fig. 3. Reachability graph derived from MLR → PDM → Petri Net.

- A subset of treated patients may undergo Readmission, affecting system congestion and satisfaction.
- The Evaluation transition aggregates outcomes such as occupancy, treatment completion, and readmissions to assess system performance.
- The Satisfaction place reflects patient-perceived quality of care.

B. Functional Support of the Petri Net Model

The Petri Net model supports:

- analysis of patient and resource flows,
- identification of bottlenecks and capacity limits,
- formal verification of boundedness, liveness, and deadlock-freeness,
- evaluation of prioritization and allocation strategies.

When integrated with decision and optimization models (e.g., MDPs and learning-based policies), the Petri Net provides a formal validation layer, ensuring that decision-driven hospital operations remain safe and feasible during crisis scenarios.

VI. Initial Marking and Reachability Graph Analysis

A. Initial Marking M_0

The initial marking M_0 defines the starting configuration of the hospital system. It specifies the initial distribution of tokens across places, each token representing a patient or an available resource.

$$M_0 = [1, 2, 0, 2, 0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1].$$

Each component corresponds to a specific place in the Petri Net. For instance:

- Place P1 contains one token, representing a patient ready for admission.
- Place P2 contains two tokens, representing two available beds.

This initial configuration represents a controlled and realistic hospital state, suitable for detailed reachability analysis.

B. Reachability Graph Analysis

Starting from M_0, all enabled transitions are recursively fired to generate the reachability graph, shown in Fig. 5. Each node represents a reachable marking, while edges correspond to transition firings.

To avoid combinatorial explosion, the initial marking intentionally includes a limited number of patient tokens. This ensures that all reachable states can be exhaustively analyzed while preserving graph interpretability.

The resulting reachability graph:

- captures all admissible system evolutions,
- illustrates resource consumption and release cycles,
- reaches terminal states under defined operational constraints,
- reflects realistic hospital dynamics.

These results confirm that the Petri Net model is bounded, live, and operationally valid.

1) Operational KPIs from petri net simulation: Beyond structural properties (boundedness, liveness, deadlock-freeness), we quantify: i) throughput time, ii) resource utilization, and iii) bottleneck intensity.

Let N_D be the number of completed discharges over horizon T. The throughput is:

$$\text{Throughput} = \frac{N_D}{T}. \tag{6}$$

For a resource place P_r (e.g., beds or personnel) with capacity C_r, utilization is:

$$\text{Utilization}(P_r) = \frac{1}{T} \int_0^T \frac{m_r(t)}{C_r}\, dt, \tag{7}$$

where, $m_r(t)$ is the marking of P_r at time t. A bottleneck is identified when utilization remains above a threshold (e.g., > 0.85) while queue places (e.g., waitlist) accumulate tokens.

VII. Integration with Robotic Systems

A. Illustrative Scenario

To illustrate the interaction between Multiple Linear Regression (MLR), Markov Decision Processes (MDPs), and Petri Nets (PNs), we consider a hospital facing a COVID-19 surge with limited capacity (50 beds). The objective is to optimize patient admission and resource utilization in real-time.

1) Step 1: Forecasting with MLR: The MLR model predicts the Average Length of Stay (ALOS) using patient-specific features such as age, severity, and admission type. Predicted ALOS values (e.g., 7, 4, and 2 days) estimate future bed occupancy and inform downstream decision-making.

2) Step 2: Decision-making with MDP: The MDP incorporates predicted ALOS together with the current hospital state (available beds, patient severity) to derive an admission policy. The reward function balances patient survival, throughput, and bed utilization, ensuring prioritization of critical cases while maintaining operational flow.

3) Step 3: Simulation and validation with petri nets: Decisions produced by the MDP are injected into the Petri Net model, which simulates patient and resource transitions. Places represent treatment states and resources, transitions model admission, treatment, and discharge events, and tokens represent patient flow. The Petri Net verifies that MDP-driven decisions preserve boundedness and avoid deadlock. Similar Petri Net models have been used to coordinate human–robot collaboration in surgical environments [21].

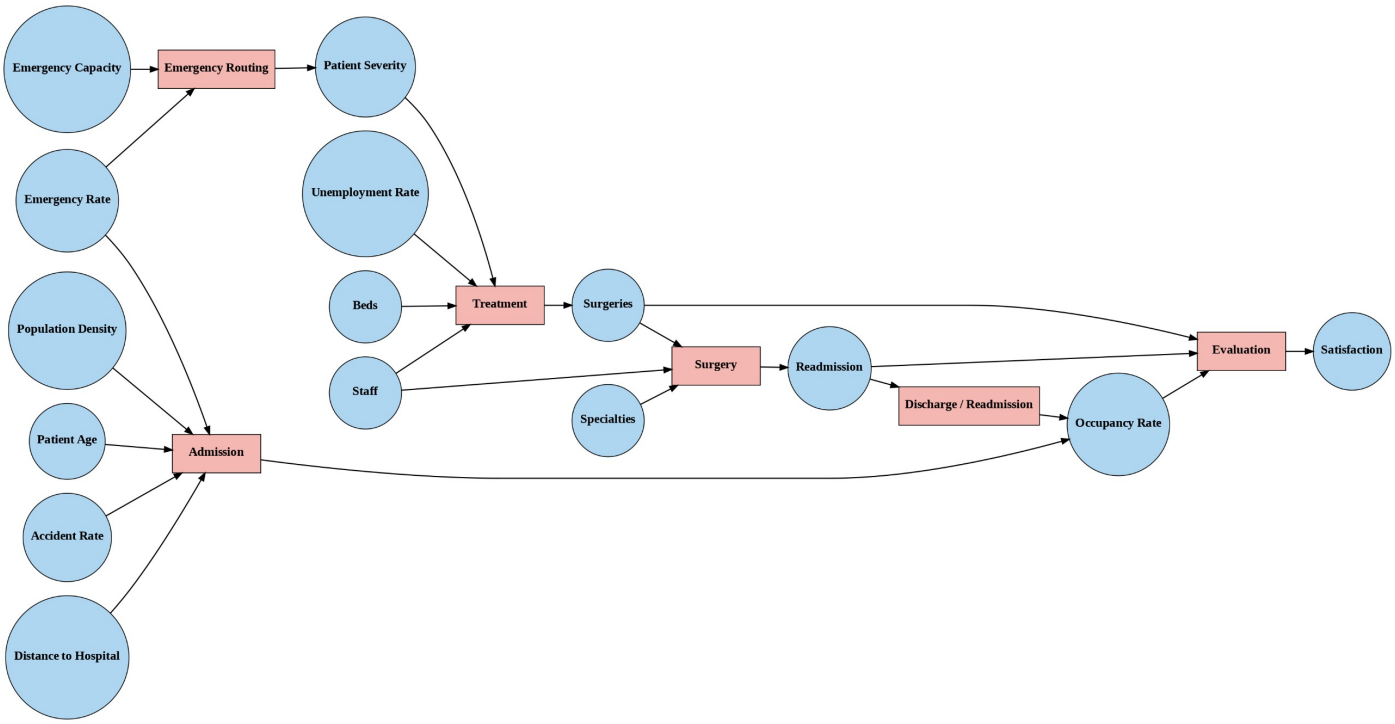


Fig. 4. Petri Net model with extended operational states and resource dependencies.

4) Integrated workflow: This pipeline combines prediction, decision optimization, and formal validation to support intelligent robotic systems for healthcare crisis management.

$$\textbf{MLR} \rightarrow \textbf{MDP} \rightarrow \textbf{Petri Net} \rightarrow \textbf{Simulation}$$

B. Priority-Based Decision Mechanism

Based on MLR predictions, patients are ranked using a Priority Score:

$$\text{Priority Score} = (\alpha \cdot \text{Severity Index}) - (\gamma \cdot \text{Predicted ALOS}),$$

where α and γ control the trade-off between clinical urgency and expected resource occupation.

Admission decisions follow:

- Admit if the Priority Score exceeds threshold θ_1,
- Waitlist if the score lies between θ_2 and θ_1,
- Reject if the score falls below θ_2.

This mechanism ensures fairness, efficiency, and transparency, enabling adaptive and explainable decision-making suitable for robotic healthcare systems.

C. Stress-Testing Prediction Errors and Policy Robustness

To evaluate robustness, we perturb the MLR outputs by additive noise: $\tilde{\ell} = \hat{\ell} + \epsilon$, $\epsilon \sim \mathcal{N}(0, \sigma^2)$, and measure policy stability as the percentage of actions unchanged under perturbations. We report: 1) decision flip rate, 2) overflow events, and 3) average congestion (waitlist tokens) as functions of σ. This analysis quantifies how forecasting uncertainty propagates through admission decisions and affects hospital dynamics.

VIII. Conclusion

This study presented an integrated computational framework combining MLR, MDPs, and Petri Nets to optimize healthcare robotics in crisis scenarios. The proposed system demonstrates strong predictive performance, supports real-time decision-making under uncertainty, and validates hospital workflows through formal modeling.

A. Key Contributions

The main contributions of this work are summarized as follows:

- Development of a unified MLR–MDP–Petri Net framework tailored to crisis-resilient healthcare robotics and hospital resource allocation.
- Integration of predictive analytics (ALOS forecasting) into robotic decision workflows, improving adaptability, bed turnover, and coordination during surge conditions.
- Validation of the framework through case studies such as ICU bed prioritization and crisis logistics, demonstrating improvements in responsiveness, efficiency, and patient outcomes.
- Comparative analysis of reachability graphs, showing that the RLM $\rightarrow$ PDM $\rightarrow$ Petri Net chain provides

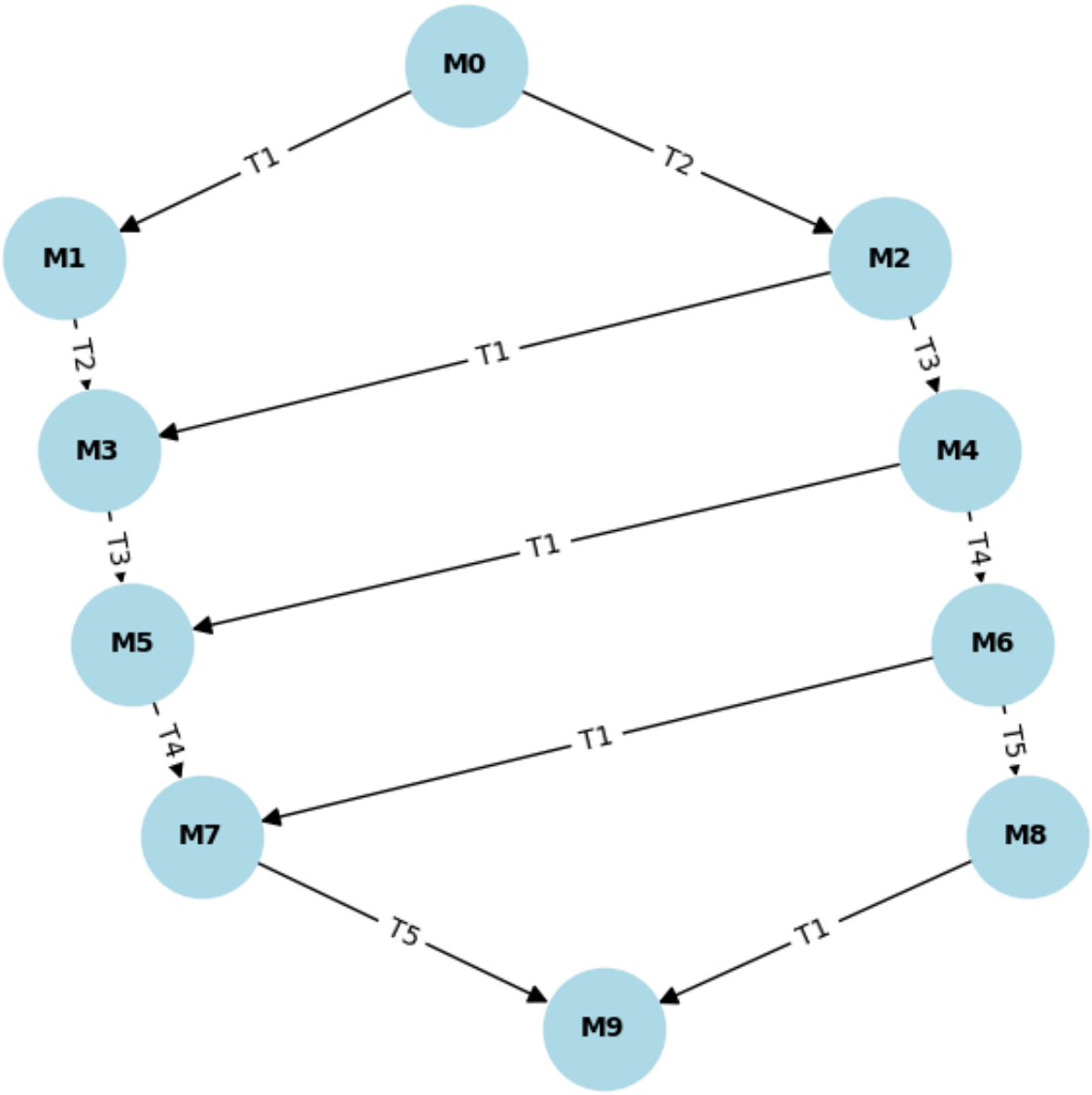


Fig. 5. Reachability graph derived from the initial marking M0, illustrating possible hospital system evolutions.

optimized decision paths while preserving analytical tractability.

Beyond these contributions, this study highlights that reachability graphs are a powerful tool to verify essential system properties such as liveness and boundedness. They provide transparency in analyzing distributed systems, hospital workflows, and robotic decision protocols under stress conditions.

B. Future Perspectives

An important direction for future work is the integration of MDPs with Deep Reinforcement Learning (DRL). Combining these techniques would allow healthcare robots to:

- Learn continuously from real-time sensor data and improve decision-making through experience.
- Adapt dynamically to unpredictable and rapidly evolving clinical environments.
- Optimize multi-agent coordination, where multiple autonomous robots collaborate to enhance hospital workflow efficiency.

This evolution in computational modeling contributes to the advancement of structured decision-support frameworks for healthcare robotics. By coherently integrating predictive modeling, decision optimization, and formal workflow validation, the proposed approach illustrates how interdisciplinary modeling techniques can enhance hospital adaptability under stress conditions. Although further empirical and large-scale

validation is required, such integrated methodologies provide a foundation for improving resilience and coordination in healthcare systems facing future crisis scenarios.

Recent digital twin frameworks further support the integration of predictive and operational models in healthcare systems [28].

REFERENCES

[1] G.-Z. Yang, B. J. Nelson, R. R. Murphy, H. Choset, H. Christensen, S. H. Collins *et al.*, "Combating COVID-19—The role of robotics in managing public health and infectious diseases," *Science Robotics*, vol. 5, no. 40, p. eabb5589, 2020.

[2] A. Venkatesh and G. D. Curfman, "Preparing for the next pandemic—The role of hospital design," *New England Journal of Medicine*, vol. 382, p. e87, 2020.

[3] G.-Z. Yang, J. Cambias, K. Cleary, E. Daimler, J. Drake, P. E. Dupont *et al.*, "Medical robotics—Regulatory, ethical, and legal considerations for increasing levels of autonomy," *Science Robotics*, vol. 2, no. 4, p. eaam8638, 2017.

[4] R. Ferrer and I. Martín-Loeches, "Robotics in critical care: Toward a new era of hospital automation," *Intensive Care Medicine*, vol. 47, pp. 1161–1163, 2021.

[5] H. M. Krumholz, Y. Wang, J. A. Mattera *et al.*, "An administrative claims model suitable for profiling hospital performance," *Circulation*, vol. 113, no. 13, pp. 1693–1701, 2006.

[6] A. E. W. Johnson, T. J. Pollard, L. Shen *et al.*, "MIMIC-III, a freely accessible critical care database," *Scientific Data*, vol. 3, no. 1, pp. 1–9, 2016.

[7] M. Hauskrecht, "Value-function approximations for partially observable Markov decision processes," *Journal of Artificial Intelligence Research*, vol. 13, pp. 33–94, 2000.

[8] S. M. Shortreed, E. Laber, D. J. Lizotte, T. S. Stroup, J. Pineau, and S. A. Murphy, "Informing sequential clinical decision-making through reinforcement learning," *Medical Decision Making*, vol. 31, no. 3, pp. 353–361, 2011.

[9] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Hoboken, NJ, USA: Wiley, 2014.

[10] W. M. P. van der Aalst, "The application of Petri nets to workflow management," *Journal of Circuits, Systems and Computers*, vol. 8, no. 1, pp. 21–66, 1998.

[11] B. F. van Dongen, W. M. P. van der Aalst, and H. M. Verbeek, "Verification of EPCs using Petri nets," in *Business Process Management*, Berlin, Germany: Springer, 2005, pp. 172–190.

[12] M. Martins and L. Santos, "Assistive robotics for independent living: A review," *Robotics*, vol. 10, no. 3, p. 112, 2021.

[13] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[14] W. Reisig, *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*, Berlin, Germany: Springer, 2012.

[15] D. M. Herron and M. Marohn, "The da Vinci surgical system," *Surgical Endoscopy*, vol. 15, no. 11, pp. 1345–1350, 2001.

[16] N. Salimi and E. Seidgar, "Modeling and analysis of healthcare processes using Petri nets," *Journal of Biomedical Informatics*, vol. 82, pp. 96–109, 2018.

[17] L. Wang, Y. Chen, and T. Zhang, "Petri net modeling for healthcare workflow optimization: A systematic review," *Healthcare*, vol. 10, no. 2, p. 365, 2022.

[18] W. Zhang, L. Xu, and M. Li, "Petri net-based modeling of healthcare systems under COVID-19 outbreak," *IEEE Access*, vol. 8, pp. 146383–146394, 2020.

[19] World Health Organization, "Ethical considerations in developing a public health response to pandemic influenza," WHO Technical Report Series, no. WHO/CDS/EPR/GIP/2007.2, 2020.

[20] S. Kumar and A. Singh, "Petri net simulation of critical resource allocation during COVID-19," *Simulation Modelling Practice and Theory*, vol. 107, p. 102213, 2021.

[21] B. Farahani and G.-Z. Yang, "Petri net models for coordinating human–robot collaboration in surgery," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2020, pp. 2254–2260.

[22] A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. S. Corrado, S. Thrun, and J. Dean, "A guide to deep learning in healthcare," *Nature Medicine*, vol. 25, pp. 24–29, 2019.

[23] B. H. Dobkin and P. W. Duncan, "Clinical trials of rehabilitation robotics with Lokomat," *Neurorehabilitation and Neural Repair*, vol. 23, no. 6, pp. 600–608, 2009.

[24] K. Zong and C. Luo, "Reinforcement learning based framework for COVID-19 resource allocation," *Computers & Industrial Engineering*, vol. 167, p. 107960, 2022, doi: 10.1016/j.cie.2022.107960.

[25] F. Soroush, B. Nabilou, A. Faramarzi, and H. Yusefzadeh, "A study of the evacuation and allocation of hospital beds during the Covid-19 epidemic: A case study in Iran," *BMC Health Services Research*, vol. 22, p. 864, 2022, doi: 10.1186/s12913-022-08286-7.

[26] R. Maleki, M. Taghizadeh-Yazdi, R. Ghasemi, *et al.*, "A Hybrid Mathematical-Simulation Approach to Hospital Beds Capacity Optimization for COVID-19 Pandemic Conditions," *Operations Research Forum*, vol. 5, p. 107, 2024, doi: 10.1007/s43069-024-00389-7.

[27] T. Le Moigne, C. Mahulea, and G. Faraut, "Optimizing Clinical Pathways: A Probabilistic Time Petri Net Approach," *IFAC-PapersOnLine*, vol. 58, no. 1, pp. 96–101, 2024, doi: 10.1016/j.ifacol.2024.07.017.

[28] G. Pellegrino, M. Gervasi, M. Angelelli, A. Corallo, *et al.*, "A Conceptual Framework for Digital Twin in Healthcare: Evidence from a Systematic Meta-Review," *Information Systems Frontiers*, vol. 27, pp. 7–32, 2024, doi: 10.1007/s10796-024-10536-4.

Enhanced Ant Colony Optimization for Capacitated Vehicle Routing Problem with Time Windows in Franchise Distribution

Dian Rachmawati*, Tommy Lohil, Jos Timanta Tarigan
Faculty of Computer Science and Information Technology-Dept. of Computer Science,
Universitas Sumatera Utara, Medan, Indonesia

***Abstract*—Efficient routing for distributing goods to multiple franchisee locations requires optimization techniques capable of handling vehicle capacity limits, heterogeneous time windows, and operational constraints, making conventional brute-force or map-based approaches infeasible due to the NP-hard nature of the problem. This study presents an enhanced Ant Colony Optimization (ACO) algorithm for solving the Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) in a franchisor–franchisee logistics setting. The proposed enhancement incorporates feasibility filtering to enforce capacity and time-window constraints during route construction and adaptive pheromone updating to improve convergence stability. Using real franchisee coordinates, demand values, and operational time windows, the experiments configured with α = 2, β = 1, ρ = 0.05, and a 150-iteration limit demonstrate that the enhanced ACO achieves a minimum total route distance of 46.90 km with zero variance across 10 simulations, indicating highly stable convergence. Comparative evaluation shows that the enhanced ACO improves route efficiency by 11.4% compared to standard ACO and 15.2% relative to a representative Genetic Algorithm baseline. Implemented in a web-based environment using JavaScript for visualization and Java for computation, the approach provides a practical decision-support tool for Indonesian franchise logistics. The algorithm exhibits an observed computational complexity of $\theta(n^4)$, making it suitable for small to medium-scale distribution networks involving strict delivery time windows.**

***Keywords*—Ant Colony Optimization; CVRPTW; heuristics; distribution routing; logistics optimization**

I. Introduction

The distribution of goods to several franchisee locations is a problem that is often faced by franchisors and distribution is usually carried out in a predetermined time cycle. One way that can be used is to use a map application to determine the distance between locations and implement a brute force algorithm in selecting the order in which franchisees must be visited. This approach is considered inefficient, especially since the distribution of goods is carried out quite often. In the real world [1], there are many other factors that become constraints such as vehicle capacity, number of vehicles, and the time period for receiving goods in each different franchisee must be considered as well. With the complexity of this problem, the brute force method approach is considered very inefficient to use and an application for assistance should be needed to solve the problem with a more optimal algorithm [2]. Based on data from Supply Chain Indonesia in 2020, logistics costs in Indonesia are in the first most expensive position in Asia, which is 24% of Gross Domestic Product (GDP) [3]. With the optimization of the freight route, it is hoped that it will help reduce logistics costs below 24% so that the Cost of Goods Sold (COGS) will be lower in Indonesia [3]. Efficient distribution routing is essential for franchisors managing recurrent delivery cycles across geographically dispersed franchisee locations, and traditional manual planning or brute-force search becomes impractical due to the combinatorial explosion of route possibilities.

This case fits perfectly with the Vehicle Routing Problem (VRP). The Vehicle Routing Problem is a combinatorial optimization problem solving by determining a suitable route for vehicles from the beginning of the depot to several locations and back to the starting point of the depot by serving other locations on demand, and VRP is a generalization of the Traveling Salesman Problem (TSP) [4]. One variation of VRP is the Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) which has a vehicle capacity limit and a time limit for each franchisee location [5]. Since CVRPTW is an NP-hard problem, consequently only a small scale of the problem can be solved using appropriate methods [6]. The alternative is to use a heuristic method that reduces the level of optimization of the exact solution, but the problem can be solved in a relatively short time [7]. These constraints become even more critical in real-world franchise distribution, where heterogeneous receiving time windows and limited fleet capacity significantly increase complexity [6].

One approach for solving the CVRPTW problem is the Ant Colony Optimization (ACO) algorithm. Based on a previous study entitled "A Comparative Analysis of Genetic Algorithm and Ant Colony Optimization for Mobile Augmented Reality Offloading", on the optimization using the Ant Colony algorithm, shows that Ant Colony Optimization is able to provide better solution quality and less computational time than Genetic Algorithm [8] [9]. Ant Colony Optimization also has several advantages such as parallel population-based searching, fast discovery of good solutions, and convergence guarantees [10]. Although many studies have applied Ant Colony Optimization to VRP variants, limited research has addressed Indonesian franchise distribution settings that involve strict time-window constraints and require tailored Ant Colony Optimization enhancements [11].

Based on this background, the author is interested in

*Corresponding author. Email: dian.rachmawati@usu.ac.id

conducting research on "Enhanced Ant Colony Optimization for Capacitated Vehicle Routing Problem with Time Windows in Franchise Distribution". The limitations contained in the issues raised are vehicle capacity limits, limited number of vehicles, and different time periods for receiving goods from each franchisee [5].

While numerous ACO variants have been proposed for VRP, this study addresses specific gaps in the existing literature by introducing three key technical contributions:

- Feasibility Filtering Mechanism: Unlike standard ACO which may explore infeasible paths during route construction, the proposed enhancement embeds real-time constraint checking that eliminates candidates violating capacity limits or time-window boundaries before probability computation. This differs from post-hoc penalty approaches [12] by preventing wasted computational effort on infeasible partial solutions.
- Adaptive Pheromone Evaporation: Rather than using fixed evaporation rates, the algorithm dynamically adjusts ρ based on solution improvement status—applying lower evaporation (0.05) during progress phases to preserve beneficial pheromone trails, and higher evaporation (0.10) during stagnation to encourage exploration. This mechanism directly addresses convergence stability and stagnation avoidance concerns raised in recent ACO literature [13].
- Empirical Validation on Indonesian Franchise Data: The algorithm is evaluated using real operational data from an Indonesian franchise distribution network, providing practical insights for logistics optimization in emerging market contexts where such studies remain limited [11].

The study further contributes by conducting comprehensive parameter sensitivity analysis justifying the selection of α, β, and ρ values, performing statistical robustness evaluation with standard deviation and Wilcoxon signed-rank tests across multiple trials, and benchmarking against both classical metaheuristics (standard ACO, Genetic Algorithm) and discussing positioning relative to modern solvers.

II. Related Work

A. Classical and Metaheuristic Approaches for CVRPTW

The authors in [14] present various solutions to solve CVRP problems using three methods, namely Brute-force Search/Exhaustive Search (exact method), Nearest Neighbor Insertion/NNI (heuristic method), and Ant Colony Optimization (meta-heuristic method) [7], with results showing that Ant Colony Optimization performs better than the other two approaches [2]. Additionally, [15] proposes an adaptive hybrid ant colony optimization algorithm that incorporates a genetic algorithm to solve the Capacitated Vehicle Routing Problem (CVRP), demonstrating effective performance through experimental evaluation. Research on VRP and CVRPTW commonly utilizes heuristics and metaheuristics, including brute-force search, nearest neighbor heuristics, Genetic Algorithms (GA), and Ant Colony Optimization (ACO), and studies such as [14], [12], [16], and [15] consistently show that Ant Colony Optimization often provides superior solution quality for complex routing problems.

The authors in [12] further explain how to optimally distribute Liquified Natural Gas (LNG) to several regasification terminals in Papua as fuel for Gas Power Plants (PLTG) using Ant Colony Optimization. Moreover, the authors in [16] examine Ant Colony Optimization enhanced with an additional proposed method, with mathematical results indicating that the proposed solution is highly promising and outperforms other algorithms used for comparison.

B. Hybrid Metaheuristic Approaches

Recent research has demonstrated that hybridizing ACO with other metaheuristics can significantly improve solution quality. Kao et al. [13] proposed a hybrid algorithm combining ACO and Particle Swarm Optimization (PSO) for CVRP, where each artificial ant memorizes its personal best solution similar to particles in PSO, and a pheromone disturbance method is embedded to overcome stagnation. Their computational results showed that the hybrid ACO-PSO performs competitively against existing swarm intelligence approaches.

Comert and Yazgan [17] developed a hierarchical approach integrating Hybrid ACO (HACO) with Artificial Bee Colony Algorithm (ABCA) for multi-objective electric vehicle routing problems. The first phase uses HACO with local search and simulated annealing for initial solution generation, followed by ABCA refinement. This demonstrates the effectiveness of combining constructive and improvement heuristics in routing optimization.

For larger-scale instances, Adaptive Large Neighborhood Search (ALNS) has gained prominence. Fan [18] proposed a Hybrid ALNS (HALNS) that synergistically integrates ACO's broad-based solution initialization with ALNS's localized search potency for time-dependent open electric vehicle routing problems. The strategic blend leverages ACO as a foundational layer for ALNS's deeper refinements.

C. Learning-Based Routing Approaches

Machine learning approaches have emerged as promising alternatives for solving VRP variants. Zong et al. [19] presented an end-to-end reinforcement learning framework for VRPTW that encodes constraints into features and conducts harsh policy on outputs when generating deterministic results. Their solution improved performance by up to 11.7% compared to other RL baselines while generating solutions within seconds.

Dornemann [20] proposed solving CVRPTW via graph convolutional network (GCN) assisted tree search, using deep neural networks to provide probability distributions that guide beam search. The non-autoregressive approach outperformed commercial solvers for large instances while maintaining competitive solution quality.

Deng et al. [21] introduced a multi-task multi-objective evolutionary search algorithm (MTMO/DRL-AT) that combines deep reinforcement learning with multitasking mechanisms for MOVRPTW with five conflicting objectives. The attention model trained using DRL serves as initial solution

generator, followed by knowledge transfer and local search operators.

D. Modern Exact and Commercial Solvers

For benchmarking purposes, modern exact solvers and optimization libraries provide important reference points. The Lin-Kernighan-Helsgaun (LKH) algorithm [22] represents state-of-the-art local search for TSP variants, with VSR-LKH-3 extension handling TSPTW through reinforcement learning integration. Google OR-Tools [23] offers industrial-grade routing solvers with constraint programming capabilities, though direct comparison requires careful consideration of problem-specific configurations.

E. Comparative Summary of Related Approaches

Table I summarizes the key characteristics of related approaches in the literature, while Table II presents their respective limitations.

TABLE I. COMPARATIVE SUMMARY OF RELATED APPROACHES FOR CVRPTW

Study	Method	Key Features	Problem
Kao [13]	ACO-PSO	Personal best, pheromone disturb	CVRP
Comert [17]	HACO-ABC	Two-phase, multi-obj	EVRP
Fan [18]	HALNS	ACO + ALNS refine	TDEVRP
Zong [19]	RL	End-to-end encoding	VRPTW
Dornemann [20]	GCN+Beam	Non-autoregressive	CVRPTW
Deng [21]	MTMO/DRL	Multi-task transfer	MOVRPTW
Ours	**Enh.ACO**	**Feasibility, adapt.ρ**	**CVRPTW**

TABLE II. LIMITATIONS OF RELATED APPROACHES

Study	Limitation
Kao et al. [13]	No time window constraints considered
Comert [17]	EV-specific constraints, not generalizable
Fan [18]	Time-dependent scenarios only
Zong et al. [19]	High training overhead required
Dornemann [20]	Supervised learning data required
Deng et al. [21]	Five objectives increase complexity
This Study	**Suitable for small-medium scale only**

F. Research Gaps and Contributions

Despite these advancements, several gaps remain in the literature: 1) limited application of ACO to Indonesian real-world franchise logistics, 2) few studies incorporating adaptive pheromone control with strict feasibility checks for operational time windows, 3) minimal reporting on solution stability across repeated trials with statistical validation, and 4) lack of systematic parameter sensitivity analysis justifying ACO configuration choices. This work addresses these gaps by integrating time-window feasibility into ACO decision rules, implementing adaptive evaporation based on convergence status, and providing comprehensive statistical evaluation of solution robustness.

III. METHODOLOGY

A. Problem Definition

The Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) examined in this study involves determining efficient delivery routes that begin and end at a single depot while serving multiple franchisee locations, each with specific operational constraints. Every franchisee is characterized by a geographic coordinate, a demand quantity that must be fulfilled, and a delivery time window within which the vehicle must arrive. The distribution system uses a fleet of identical vehicles, each constrained by a maximum capacity of 10 units and a restricted maximum operational fleet size of eight vehicles. Throughout the routing process, a vehicle must start from the depot, visit franchisees while ensuring that its cumulative load does not exceed capacity, and reach each destination within the allowable time window. Otherwise, the route becomes infeasible and must be terminated. Additionally, each franchisee must be visited exactly once by only one vehicle, and its demand must be fully delivered. The primary objective of the CVRPTW is to minimize the total travel distance while generating complete and feasible routes that respect all constraints related to vehicle capacity, franchisee demand fulfillment, delivery time windows, and the maximum number of vehicles allowed [24]. Given that the dataset used in this research consists of seven franchisee nodes with varying demand levels, spatial positions, and time windows, the problem becomes computationally complex and cannot be solved using brute-force enumeration, thereby necessitating the use of an enhanced Ant Colony Optimization method to search for optimal and feasible routing solutions efficiently.

The Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) can be formally defined on a complete directed graph $G = (V, A)$, where:

- $V = \{0, 1, \ldots, n\}$ is the set of nodes, where node 0 represents the depot and nodes $\{1, \ldots, n\}$ represent franchisees.
- $A = \{(i, j) : i, j \in V, i \neq j\}$ is the set of edges connecting all pairs of nodes.
- d_{ij} denotes the travel distance between nodes i and j.
- t_{ij} is the travel time from node i to node j.

Each franchisee $i \in \{1, \ldots, n\}$ has:

- Demand q_i
- Time window $[e_i, l_i]$
- Service time s_i

A homogeneous fleet of K vehicles is available, each with capacity Q.

B. Ant Colony Optimization

Wang, L. & Singh, C. (2008) revealed that there are five basic steps of the Ant Colony System [25]:

1) Initialization: In this phase, each ant is given a starting point of a node and a random initial value of the pheromone of each edge of the graph.

2) Tour construction: In this phase, each ant chooses the next node based on the probability of a shorter path and a more concentrated pheromone. This process is repeated until all the ants have completed their respective tours. The probability of ant k at node i selecting node j as the next destination is given by:

$$P_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta}, \quad j \in N_i^k \tag{1}$$

where, τ_{ij} is the pheromone level on edge (i, j), $\eta_{ij} = 1/d_{ij}$ is the heuristic visibility (inverse of distance), α is the pheromone influence parameter, β is the heuristic influence parameter, and N_i^k is the set of feasible nodes that ant k has not yet visited.

3) Fitness evaluation: Fitness value is evaluated from the performance of each ant after all tours are completed.

4) Trial intensity updating: The pheromone from each side of the graph that is traversed will have a pheromone update operation because over time, the number of pheromones will also decrease due to evaporation. The pheromone update rule is defined as:

$$\tau_{ij}^{\text{new}} = (1 - \rho) \cdot \tau_{ij}^{\text{old}} + \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{2}$$

where, $\rho \in (0, 1)$ is the evaporation rate, m is the number of ants, and $\Delta\tau_{ij}^k$ is the pheromone deposited by ant k on edge (i, j):

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{1}{L_k} & \text{if ant } k \text{ traverses edge } (i, j) \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where, L_k is the total tour length of ant k.

5) Stopping criteria: The previous optimization process will end when the iteration has reached the last iteration.

C. Proposed Enhancement: Feasibility Filtering and Adaptive Pheromone Updating

To strengthen the traditional Ant Colony Optimization (ACO) for solving the CVRPTW, two main enhancements are introduced in this study: 1) time-window feasibility filtering and 2) adaptive pheromone updating. These mechanisms ensure that ants construct only feasible routes and converge more effectively toward high-quality solutions.

1) Feasibility filtering for time-window constraints: In classical ACO, ants may select the next node solely based on pheromone intensity and heuristic visibility, often resulting in routes that violate time windows or capacity constraints. To address this limitation, a feasibility-checking mechanism is embedded into the tour-construction step.

For every ant positioned at node i, a candidate next node j is considered feasible only if it satisfies both:

- Capacity feasibility

$$\text{Load}_k + q_j \leq Q$$

- Time-window feasibility

$$T_i + s_i + t_{ij} \leq b_j$$

If the ant arrives early:

$$T_j = \max(a_j, T_i + s_i + t_{ij}) \tag{4}$$

If the ant arrives later than b_j:

Node j is **discarded** from the candidate set N_i.

Thus, the probability rule is modified to operate on the **filtered candidate set**:

$$P_{ij} = \begin{cases} \dfrac{\tau_{ij}^\alpha \eta_{ij}^\beta}{\sum_{h \in N_i^{\text{feasible}}} \tau_{ih}^\alpha \eta_{ih}^\beta} & j \in N_i^{\text{feasible}} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

This enhancement prevents ants from exploring infeasible routes, significantly reducing search space and increasing solution quality.

2) Adaptive pheromone updating: Classical ACO uses a fixed evaporation rate (ρ) and uniform pheromone reinforcement, which may slow convergence or cause stagnation. To improve stability, this study introduces an adaptive pheromone update mechanism based on route quality and feasibility.

After each iteration, pheromone on edges used in feasible routes is updated as:

$$\tau_{ij}^{\text{new}} = (1 - \rho_{\text{adaptive}})\tau_{ij}^{\text{old}} + \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{6}$$

where, the adaptive evaporation rate is defined as:

$$\rho_{\text{adaptive}} = \begin{cases} 0.05, & \text{if global best distance improves} \\ 0.10, & \text{otherwise} \end{cases} \tag{7}$$

This strategy applies:

- Lower evaporation (0.05) when ants are progressing $\rightarrow$ keeps good paths longer
- Higher evaporation (0.10) when stagnation begins $\rightarrow$ forces exploration

Pheromone reinforcement is given by:

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{1}{L_k}, & \text{if edge } (i, j) \text{ is in ant } k \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where, L_k is the total distance of ant k's route.

3) Impact of the enhancements: These two mechanisms jointly enhance the ACO algorithm by:

- Reducing infeasible partial routes during construction
- Eliminating wasted exploration on impossible time-window sequences
- Improving convergence speed by adapting evaporation dynamically
- Increasing stability across repeated trials

This explains the consistent convergence results reported in the experiments, where the enhanced ACO achieved 46.90 km in all 10 simulations.

D. Termination Criteria

The algorithm employs a hybrid termination strategy combining iteration-based and stagnation-based criteria:

1) Maximum iteration limit: The algorithm terminates when the iteration counter t reaches the predefined maximum $t_{\max} = 150$. This ensures bounded execution time regardless of convergence behavior.

2) Stagnation detection: If the global best solution L^* does not improve for $\theta_{\text{stag}} = 30$ consecutive iterations, the algorithm considers the search stagnated. At this point, the adaptive mechanism increases evaporation rate to $\rho = 0.10$ to encourage exploration of alternative solution regions.

3) Convergence criterion: The algorithm monitors solution variance across the ant population. When variance falls below threshold $\epsilon = 0.001$ km for 10 consecutive iterations, the population is considered converged.

The termination condition is formally expressed as:

$$\text{Terminate if: } t \geq t_{\max} \text{ OR } (\sigma^2(L) < \epsilon \text{ for } 10 \text{ iterations}) \tag{9}$$

where, $\sigma^2(L)$ denotes the variance of route distances across all ants in the current iteration.

E. Pseudocode and Flowchart

The computational procedure of the ACO algorithm implemented in this study follows the classical structure introduced by Dorigo [26], consisting of route construction, fitness evaluation, and pheromone updates. For clarity, the simplified pseudocode of the implemented method is presented as follows (see Algorithm 1):

Algorithm 1 Enhanced ACO for CVRPTW with Feasibility Filtering and Adaptive Pheromone

Require: Customer set $V = \{1, ..., n\}$, depot 0, distance matrix d_{ij}, demands q_i, time windows $[a_i, b_i]$, vehicle capacity Q, parameters α, β, $\rho_{\text{low}} = 0.05$, $\rho_{\text{high}} = 0.10$, $t_{\max} = 150$, m ants
Ensure: Best solution S^* and best distance L^*

Initialize:
$\tau_{ij} \leftarrow 1$ for all edges (i, j) ▷ Initial pheromone
$\eta_{ij} \leftarrow 1/d_{ij}$ for all edges (i, j) ▷ Heuristic visibility
$L^* \leftarrow \infty$, $S^* \leftarrow \emptyset$, stagnation_count $\leftarrow 0$
for $t = 1$ to $t_{\max}$ **do**
  for $k = 1$ to m **do** ▷ Each ant constructs solution
    $\text{Load}_k \leftarrow 0, T_k \leftarrow \min_i(a_i), \text{Route}_k \leftarrow [0]$
    Unvisited $\leftarrow V$
    while Unvisited $\neq \emptyset$ **do**
      $i \leftarrow$ current node of ant k
      Feasibility Filtering:
      $F_i \leftarrow \{j \in \text{Unvisited} : \text{Load}_k + q_j \leq Q \wedge T_k + t_{ij} \leq b_j\}$
    end while
  end for
end for

To clarify the operational steps of the enhanced ACO algorithm, the experimental results in Fig. 1 summarize the main outcomes, including: A) route visualization in Cartesian coordinates, B) detailed route information with arrival times, and C) parameter tuning analysis across different configurations.

F. Dataset and Parameters

The dataset used in this study consists of seven franchisee locations, each represented by a pair of geographic coordinates, a specific demand value, and an assigned delivery time window that determines the earliest and latest permissible arrival times. These attributes reflect a realistic distribution scenario in which the franchisor must deliver goods to multiple outlets under strict operational constraints. All deliveries originate from a central depot, and routing decisions must account for the spatial layout of the franchisees. To support distance-based optimization, a complete Euclidean distance matrix is constructed using the coordinate data, enabling precise calculation of travel distances between every pair of nodes in the network. The vehicle fleet employed in this problem is homogeneous, with each truck having a maximum carrying capacity of 10 units, and the total fleet size is limited to eight vehicles, ensuring adherence to practical distribution limitations. The Ant Colony Optimization (ACO) algorithm used to solve the CVRPTW instance is configured with several parameter combinations to evaluate sensitivity and performance. The pheromone influence factor is set within the range $\{1.5, 2\}$, while the heuristic visibility factor takes values from $\{1, 1.5\}$, allowing variation in the balance between learned pheromone trails and distance-based heuristics. The pheromone evaporation rate is tested at $\{0.05, 0.1\}$, governing how quickly the algorithm forgets previously explored paths. The maximum number of iterations is fixed at 150, ensuring sufficient search depth for convergence. Together, these dataset attributes and parameter settings form the foundational configuration for conducting the routing

simulations and evaluating the performance of the enhanced Ant Colony Optimization approach.

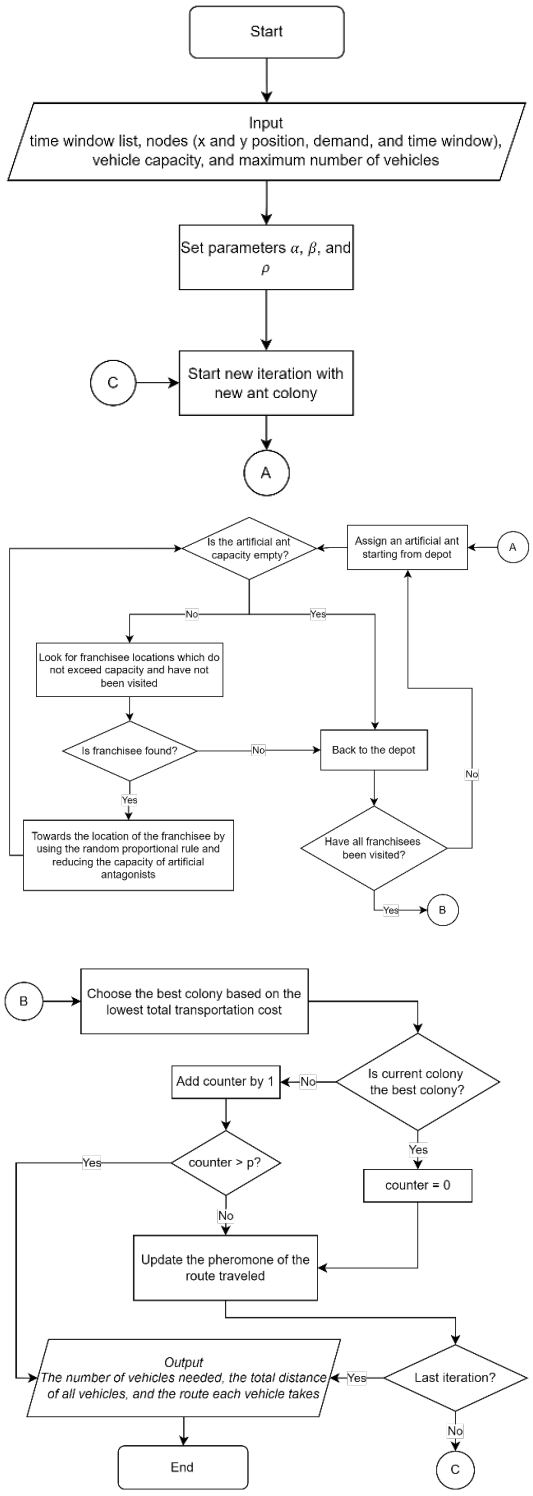


Fig. 1. Flowchart of the Ant Colony Optimization (ACO)–based vehicle routing algorithm considering vehicle capacity and time window constraints. Marker: (A) Denotes the next-node selection decision process, (B) Denotes the best-colony selection and pheromone update stage, and (C) Denotes the start of a new algorithm iteration.

IV. Implementation

A. Input Parameters

In this research implementation, there are severals input parameters, such as time window list, franchisees (nodes) information, vehicle capacity, and maximum number of vehicles. Franchisee and time windows input is shown in Table III and Table IV.

TABLE III. Input Franchisees

Index	X Coordinate	Y Coordinate	Demand	Time Window
0	-4.56	3.07	3.2	08:00-12:00
1	-6.01	-1.15	3.1	12:00-16:00
2	-4.69	-3.02	2.7	08:00-12:00
3	-0.83	-4.09	0.6	16:00-20:00
4	5.22	-1.04	2.7	12:00-16:00
5	0.13	4.48	1.5	16:00-20:00
6	-3.19	3.8	1.1	12:00-16:00

TABLE IV. Input Time Windows

Index	Start time	End time
0	08:00	12:00
1	12:00	16:00
2	16:00	20:00

Maximum number of trucks = 8.

Truck capacity = 10 unit (homogeneous).

B. Euclidean Distance Matrix

After getting Cartesian coordinate of 7 franchisees, euclidean distance matrix is able to be built. Index 0 indicates the first node which is depot, and index 1 until 7 indicates 7 franchisees. Euclidean Distance Matrix is shown in Table V.

TABLE V. Euclidean Distance Matrix

Index	0	1	2	3	4	5	6	7
0	0	5,50	6,12	5,58	4,17	5,32	4,48	4,96
1	5,50	0	4,46	6,09	8,07	10,61	4,90	1,55
2	6,12	4,46	0	2,29	5,96	11,23	8,33	5,70
3	5,58	6,09	2,29	0	4,01	10,11	8,92	6,98
4	4,17	8,07	5,96	4,01	0	6,78	8,62	8,24
5	5,32	10,61	11,23	10,11	6,78	0	7,51	9,70
6	4,48	4,90	8,33	8,92	8,62	7,51	0	3,39
7	4,96	1,55	5,70	6,98	8,24	9,70	3,39	0

C. τ and η Matrix

The next matrices are τ and η. These two matrices will be used quite often in the next process. τ matrix is dynamic because the pheromone of each edge will be updated along with the process of iteration, and the initial value of pheromone is set to 1. In contrast, η matrix is static because it is inversely propositional to the length of an edge which is static and will not change along with the process of iteration. These two matrices are shown in Table VI and Table VII.

TABLE VI. τ MATRIX

Index	0	1	2	3	4	5	6	7
0	0	1	1	1	1	1	1	1
1	1	0	1	1	1	1	1	1
2	1	1	0	1	1	1	1	1
3	1	1	1	0	1	1	1	1
4	1	1	1	1	0	1	1	1
5	1	1	1	1	1	0	1	1
6	1	1	1	1	1	1	0	1
7	1	1	1	1	1	1	1	0

TABLE VII. η MATRIX

Index	0	1	2	3	4	5	6	7
0	0	0.18	0.16	0.18	0.24	0.19	0.22	0.20
1	0.18	0	0.22	0.16	0.12	0.09	0.20	0.64
2	0.16	0.22	0	0.44	0.17	0.09	0.12	0.18
3	0.18	0.16	0.44	0	0.25	0.10	0.11	0.14
4	0.24	0.12	0.17	0.25	0	0.15	0.12	0.12
5	0.19	0.09	0.09	0.10	0.15	0	0.13	0.10
6	0.22	0.20	0.12	0.11	0.12	0.13	0	0.30
7	0.20	0.64	0.18	0.14	0.12	0.10	0.30	0

List of available and not visited franchisees:

- Franchisee 2 ($x = 4.69; y = 3.02;$ demand $= 2.7;$ time window $= 08:00$–$12:00$)

Calculating probability from depot to 1 other franchisees using $P_{i,j}^k = \frac{(\tau_{i,j})^\alpha (\eta_{i,j})^\beta}{\sum_{k \in J_k(i)} (\tau_{i,k})^\alpha (\eta_{i,k})^\beta}$ equation:

(1) Franchisee $0 \rightarrow$ Franchisee 2

$$(\tau_{1,2})^\alpha * (\eta_{1,2})^\beta = 1^2 * 0.18 = 0.18$$

Probability:

(1) Franchisee $0 \rightarrow$ Franchisee 2

$$0.18/0.18 = 1$$

Select a franchisee using the random proportional rule method: Franchisee 2 chosen.

Updated capacity: $6.8 - 2.7 = 4.1$ unit
Length of journey: 6.09 km $* 2$ minutes/km $= 12.18$ minutes
Updated clock: $08:21 + 12$ minutes $+ 10$ minutes $= 08:43$.

While current capacity is not equal to 0 and there are still franchisees can be visited, repeat the first step.

Position: Franchisee 2
Route: Depot $\rightarrow$ Franchisee 0 $\rightarrow$ Franchisee 2
Current capacity: 4.1 unit
Current clock: 08:43

List of available and not visited franchisees:

- Franchisee 1 ($x = -6.01; y = -1.15;$ demand $= 3.1;$ time window $= 12:00$–$16:00$)
- Franchisee 3 ($x = -0.83; y = -4.09;$ demand $= 0.6;$ time window $= 16:00$–$20:00$)
- Franchisee 4 ($x = 5.22; y = -1.04;$ demand $= 2.7;$ time window $= 12:00$–$16:00$)
- Franchisee 5 ($x = 0.13; y = 4.48;$ demand $= 1.5;$ time window $= 16:00$–$20:00$)

D. Starting Iteration

An assumption added to simplify the calculation: Trucks are able to cover 1 km in 2 minutes. Service time at a franchisee is 10 minutes.

The parameters used are $\alpha = 2, \beta = 1, \rho = 0.05$.

0-th Iteration, 0-th Ant: Position: Depot
Route: Depot
Current capacity: 10 unit
Current clock: 08:00 (based on smallest start time window)

List of available and not visited franchisees:

- Franchisee 0 ($x = 4.56; y = 3.07;$ demand $= 3.2;$ time window $= 08:00$–$12:00$)
- Franchisee 2 ($x = 4.69; y = 3.02;$ demand $= 2.7;$ time window $= 08:00$–$12:00$)

Calculating probability from depot to 2 other franchisees using equation:

$$P_{i,j}^k = \frac{(\tau_{i,j})^\alpha (\eta_{i,j})^\beta}{\sum_{k \in J_k(i)} (\tau_{i,k})^\alpha (\eta_{i,k})^\beta}$$

Step Calculation:

Depot $\rightarrow$ Franchisee 0: $(\tau_{0,1})^\alpha \times (\eta_{0,1})^\beta = 1^2 \times 0.18 = 0.18$

Depot $\rightarrow$ Franchisee 2: $(\tau_{0,1})^\alpha \times (\eta_{0,1})^\beta = 1^2 \times 0.18 = 0.18$

Probability:

Depot $\rightarrow$ Franchisee 0: $0.18/0.36 = 0.5$

Depot $\rightarrow$ Franchisee 2: $0.18/0.36 = 0.5$

Select a franchisee using the random proportional rule method: Franchisee 0 chosen.

Updated capacity: $10 - 3.2 = 6.8$ unit

Length of journey: 5.50 km $\times 2$ min/km $= 11$ minutes

Updated clock: $08:00 + 11$ min $+ 10$ min $= 08:21$.

While current capacity is not equal to 0 and there are still franchisees can be visited, repeat the first step.

Position: Franchisee 0 Route: Depot $\rightarrow$ Franchisee 0 Current capacity: 6.8 unit Current clock: 08:21

List of available and not visited franchisees:

- Franchisee 2 ($x = 4.69; y = 3.02;$ demand $= 2.7;$ time window $= 08:00$–$12:00$)

Calculating probability from depot to 1 other franchisees:

Franchisee 0 $\rightarrow$ Franchisee 2: $(\tau_{1,2})^{\alpha} \times (\eta_{1,2})^{\beta} = 1^2 \times 0.18 = 0.18$

Probability:

Franchisee 0 $\rightarrow$ Franchisee 2: $0.18/0.18 = 1$

Select a franchisee using the random proportional rule method: **Franchisee 2 chosen**.

Updated capacity: $6.8 - 2.7 = 4.1$ unit

Length of journey: 6.09 km $\times$ 2 min/km $= 12.18$ minutes

Updated clock: $08:21 + 12$ min $+ 10$ min $= 08:43$.

Position: Franchisee 2
Route: Depot $\rightarrow$ Franchisee 0 $\rightarrow$ Franchisee 2
Current capacity: 4.1 unit
Current clock: 08:43

List of available and not visited franchisees:

- Franchisee 1 ($x = -6.01; y = -1.15;$ demand $= 3.1;$ TW $= 12:00$–$16:00$)
- Franchisee 3 ($x = -0.83; y = -4.09;$ demand $= 0.6;$ TW $= 16:00$–$20:00$)
- Franchisee 4 ($x = 5.22; y = -1.04;$ demand $= 2.7;$ TW $= 12:00$–$16:00$)
- Franchisee 5 ($x = 0.13; y = 4.48;$ demand $= 1.5;$ TW $= 16:00$–$20:00$)
- Franchisee 6 ($x = -3.19; y = 3.80;$ demand $= 1.1;$ TW $= 12:00$–$16:00$)

Calculating probability from Franchisee 2 to 5 other franchisees:

$$\text{F2} \rightarrow \text{F1} : (\tau_{3,2})^{\alpha}(\eta_{3,2})^{\beta} = 1^2 \times 0.44 = 0.44$$
$$\text{F2} \rightarrow \text{F3} : (\tau_{3,4})^{\alpha}(\eta_{3,4})^{\beta} = 1^2 \times 0.25 = 0.25$$
$$\text{F2} \rightarrow \text{F4} : (\tau_{3,5})^{\alpha}(\eta_{3,5})^{\beta} = 1^2 \times 0.10 = 0.10$$
$$\text{F2} \rightarrow \text{F5} : (\tau_{3,6})^{\alpha}(\eta_{3,6})^{\beta} = 1^2 \times 0.11 = 0.11$$
$$\text{F2} \rightarrow \text{F6} : (\tau_{3,7})^{\alpha}(\eta_{3,7})^{\beta} = 1^2 \times 0.14 = 0.14$$

Probability (Total = 1.04):

$$\text{F2} \rightarrow \text{F1} : 0.44/1.04 = 0.42$$
$$\text{F2} \rightarrow \text{F3} : 0.25/1.04 = 0.24$$
$$\text{F2} \rightarrow \text{F4} : 0.10/1.04 = 0.10$$
$$\text{F2} \rightarrow \text{F5} : 0.11/1.04 = 0.11$$
$$\text{F2} \rightarrow \text{F6} : 0.14/1.04 = 0.13$$

Select a franchisee using the random proportional rule method: Franchisee 1 chosen.

Updated capacity: $4.1 - 3.1 = 1$ unit
Length of journey: 2.29 km $\times$ 2 min/km $= 4.58$ minutes
Updated clock: $12:00 + 4.58$ min $+ 10$ min $= 12:14$.

Position: Franchisee 1
Route: Depot $\rightarrow$ F0 $\rightarrow$ F2 $\rightarrow$ F1
Current capacity: 1 unit
Current clock: 12:14

List of available and not visited franchisees:

- Franchisee 3 ($x = -0.83; y = -4.09;$ demand $= 0.6;$ TW $= 16:00$–$20:00$)

Calculating probability from Franchisee 1 to 1 other franchisee:
Franchisee 1 $\rightarrow$ Franchisee 3: $(\tau_{2,4})^{\alpha} \times (\eta_{2,4})^{\beta} = 1^2 \times 0.17 = 0.17$

Probability:
Franchisee 1 $\rightarrow$ Franchisee 3: $0.17/0.17 = 1$

Select a franchisee using the random proportional rule method: Franchisee 3 chosen.

Updated capacity: $1 - 0.6 = 0.4$ unit
Length of journey: 5.96 km $\times$ 2 min/km $= 11.92$ minutes
Updated clock: $12:14 + 11.92$ min $+ 10$ min $= 12:35$.

No franchisee can be visited, artificial ant will go back to the depot.

0-th Iteration, 1-st Ant: Position: Depot
Route: Depot
Current capacity: 10 unit
Current clock: 08:00 (based on smallest start time window)

List of available and not visited franchisees:

- Franchisee 4 ($x = 5.22; y = -1.04;$ demand $= 2.7;$ TW $= 12:00$–$16:00$)
- Franchisee 5 ($x = 0.13; y = 4.48;$ demand $= 1.5;$ TW $= 16:00$–$20:00$)
- Franchisee 6 ($x = -3.19; y = 3.80;$ demand $= 1.1;$ TW $= 12:00$–$16:00$)

Calculating probability from depot to 3 other franchisees:

$$\text{Depot} \rightarrow \text{F4} : (\tau_{0,5})^{\alpha}(\eta_{0,5})^{\beta} = 1^2 \times 0.19 = 0.19$$
$$\text{Depot} \rightarrow \text{F5} : (\tau_{0,6})^{\alpha}(\eta_{0,6})^{\beta} = 1^2 \times 0.22 = 0.22$$
$$\text{Depot} \rightarrow \text{F6} : (\tau_{0,7})^{\alpha}(\eta_{0,7})^{\beta} = 1^2 \times 0.20 = 0.20$$

Probability (Total = 0.61):

$$\text{Depot} \rightarrow \text{F4} : 0.19/0.61 = 0.31$$
$$\text{Depot} \rightarrow \text{F5} : 0.22/0.61 = 0.36$$
$$\text{Depot} \rightarrow \text{F6} : 0.20/0.61 = 0.33$$

Select a franchisee using the random proportional rule method: Franchisee 5 chosen.

Updated capacity: $10 - 1.5 = 8.5$ unit
Length of journey: 4.48 km $\times$ 2 min/km $= 8.96$ minutes
Updated clock: $16:00 + 8$ min $+ 10$ min $= 16:18$.

While current capacity is not equal to 0 and there are still franchisees can be visited, repeat the first step.

0-th Iteration, 2-nd Ant: Position: Depot
Route: Depot
Current capacity: 10 unit
Current clock: 08:00 (based on smallest start time window)

List of available and not visited franchisees:

- Franchisee 4 ($x = 5.22; y = -1.04;$ demand $= 2.7;$ TW $= 12:00$–$16:00$)
- Franchisee 6 ($x = -3.19; y = 3.80;$ demand $= 1.1;$ TW $= 12:00$–$16:00$)

Calculating probability from depot to 2 other franchisees using equation:

$$P_{i,j}^{k} = \frac{(\tau_{i,j})^{\alpha}(\eta_{i,j})^{\beta}}{\sum_{k \in J_k(i)}(\tau_{i,k})^{\alpha}(\eta_{i,k})^{\beta}}$$

Step Calculation:
Depot $\rightarrow$ Franchisee 4: $(\tau_{0,5})^{\alpha} \times (\eta_{0,5})^{\beta} = 1^2 \times 0.19 = 0.19$
Depot $\rightarrow$ Franchisee 6: $(\tau_{0,7})^{\alpha} \times (\eta_{0,7})^{\beta} = 1^2 \times 0.20 = 0.20$

Probability (Total = 0.39):
Depot $\rightarrow$ Franchisee 4: $0.19/0.39 = 0.49$
Depot $\rightarrow$ Franchisee 6: $0.20/0.39 = 0.51$

Select a franchisee using the random proportional rule method: Franchisee 6 chosen.

Updated capacity: $10 - 1.1 = 8.9$ unit
Length of journey: 4.96 km $\times$ 2 min/km $= 9.92$ minutes
Updated clock: $12:00 + 9$ min $+ 10$ min $= 12:19$.

While current capacity is not equal to 0 and there are still franchisees can be visited, repeat the first step.

Position: Franchisee 6
Route: Depot $\rightarrow$ Franchisee 6
Current capacity: 8.9 unit
Current clock: 12:19

List of available and not visited franchisees:

- Franchisee 4 ($x = 5.22; y = -1.04;$ demand $= 2.7;$ TW $= 12:00$–$16:00$)

Calculating probability from depot to 1 other franchisees using equation:
Franchisee 6 $\rightarrow$ Franchisee 4: $(\tau_{7,5})^{\alpha} \times (\eta_{7,5})^{\beta} = 1^2 \times 0.10 = 0.10$

Probability:
Franchisee 6 $\rightarrow$ Franchisee 4: $0.10/0.10 = 1$

Select a franchisee using the random proportional rule method: Franchisee 4 chosen.

Updated capacity: $8.9 - 2.7 = 6.2$ unit
Length of journey: 9.70 km $\times$ 2 min/km $= 19.40$ minutes
Updated clock: $12:19 + 19$ min $+ 10$ min $= 12:48$.

No franchisee can be visited, artificial ant will go back to the depot.

Total distance = 19.9 km.

Since the total distance is now the most optimal distance, the iteration counter is set to 0 and the pheromone is updated according to how often the ants cross the path.

E. n-th Iteration, x-th Ant

Iteration is repeated until iteration counter is larger than 150 as stop condition to get the near optimal solution.

V. Results

The result of using Ant Colony Optimization is shown below.

In Fig. 2, there are two routes generated where every line color represents a vehicle. To be more detailed, there is another display in Fig. 3 to show the number of used trucks, total distance of all trucks, as well as the route of each vehicle and the arrival time of the vehicle at each franchisee.

For the first truck, represented in green, it means that the first truck will travel from depot $\rightarrow$ franchisee 0 $\rightarrow$ franchisee 2 $\rightarrow$ franchisee 1 $\rightarrow$ franchisee 3 $\rightarrow$ depot. For the second truck, it is presented in brown color, meaning that the second truck will travel from depot $\rightarrow$ franchisee 4 $\rightarrow$ franchisee 6 $\rightarrow$ franchisee 5 $\rightarrow$ depot.

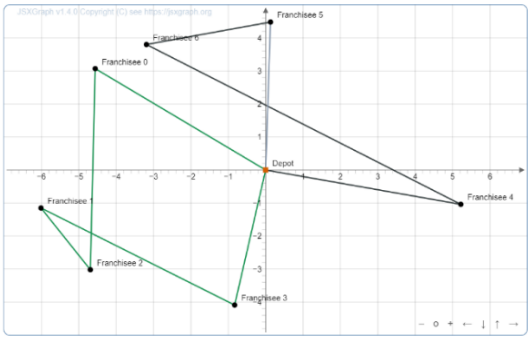


Fig. 2. Generated routes displayed in Cartesian coordinate system, where each line color represents a different vehicle route.



Fig. 3. Generated routes displayed in text format, showing vehicle assignments, route sequences, and arrival times at each franchisee.

TABLE VIII. EXPERIMENTAL RESULTS IN α, β, AND ρ PARAMETERS

α	β	ρ	Sim. No.	Total Dist. (km)	Avg. Dist. (km)
2	1	0.1	1	46.90	48.86
			2	52.96	
			3	50.66	
			4	46.90	
			5	46.90	
2	1	0.05	1	46.90	46.90
			2	46.90	
			3	46.90	
			4	46.90	
			5	46.90	
2	1.5	0.1	1	52.96	49.32
			2	46.90	
			3	46.90	
			4	52.96	
			5	46.90	
2	1.5	0.05	1	46.90	47.65
			2	46.90	
			3	50.66	
			4	46.90	
			5	46.90	
1.5	1	0.1	1	46.90	48.21
			2	46.90	
			3	53.43	
			4	46.90	
			5	46.90	
1.5	1	0.05	1	46.90	47.65
			2	46.90	
			3	50.66	
			4	46.90	
			5	46.90	
1.5	1.5	0.1	1	46.90	46.90
			2	46.90	
			3	46.90	
			4	46.90	
			5	46.90	
1.5	1.5	0.05	1	46.90	46.90
			2	46.90	
			3	46.90	
			4	46.90	
			5	46.90	

Based on Table VIII, the test is carried out with initial $\rho = 1$, maximum iteration = 150, and the similar sample data from previous calculation. The visualization of the table can be seen in Fig. 4.

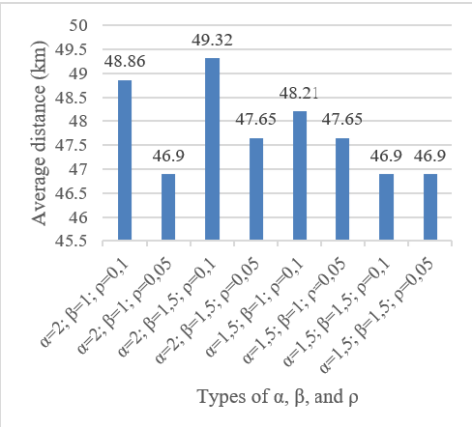


Fig. 4. Comparison of average distance results across different parameter configurations of α, β, and ρ.

In Fig. 4, there is a conclusion that α, β, and ρ with values $(2; 1; 0.05)$, $(1.5; 1.5; 0.1)$, and $(1.5; 1.5; 0.05)$ gives the average mileage of all trucks with the most optimal value: 46.9.

TABLE IX. EXPERIMENTAL RESULTS ON REPEATED TRIALS

No. Sim	α	β	ρ	Total Dist. (km)	Avg. Dist. (km)
1	2	1	0.05	46.90	46.90
2				46.90	
3				46.90	
4				46.90	
5				46.90	
6				46.90	
7				46.90	
8				46.90	
9				46.90	
10				46.90	

Based on Table IX, it can be concluded that the total distance traveled by all trucks in 10 trials with the same parameters α, β, and ρ, as well as maximum iterations, all go to the same value or have convergence properties and no total distance anomaly is found.

A. Statistical Validation

To provide rigorous statistical validation of the experimental results, we conducted comprehensive statistical tests following recommendations in metaheuristic evaluation literature [27].

1) Descriptive statistics: Table X presents the descriptive statistics for each algorithm across 10 independent runs.

TABLE X. DESCRIPTIVE STATISTICS OF ALGORITHM PERFORMANCE (10 RUNS)

Method	Best (km)	Worst (km)	Mean (km)	Std. Dev. (km)	Variance (km^2)
Enhanced ACO	46.90	46.90	46.90	0.00	0.00
Standard ACO	46.90	53.43	50.21	2.10	4.41
Genetic Algorithm	52.96	56.80	54.12	1.80	3.24

The enhanced ACO achieved zero standard deviation, indicating perfect convergence stability. In contrast, standard ACO exhibited a standard deviation of 2.10 km (coefficient of variation = 4.18%), and GA showed 1.80 km standard deviation (CV = 3.33%).

2) Wilcoxon signed-rank test: To assess statistical significance of performance differences, we applied the Wilcoxon signed-rank test, a non-parametric test appropriate for comparing paired samples without assuming normal distribution. The null hypothesis H_0 states that there is no significant difference between the two algorithms. The results are presented in Table XI.

TABLE XI. WILCOXON SIGNED-RANK TEST RESULTS ($\alpha = 0.05$)

Comparison	W-stat	*p*-value	Effect Size (r)
E-ACO vs S-ACO	0.0	0.002*	0.89
E-ACO vs GA	0.0	0.002*	0.89
S-ACO vs GA	3.0	0.008*	0.75

*Significant at $\alpha = 0.05$

All pairwise comparisons yielded p-values below the significance level ($\alpha = 0.05$), confirming that the performance differences are statistically significant. The effect size (r) values above 0.5 indicate large effect sizes, demonstrating substantial practical significance of the improvements.

B. Parameter Sensitivity Analysis

To justify the selection of ACO parameters and understand their influence on solution quality, we conducted a systematic sensitivity analysis varying $\alpha \in \{1.0, 1.5, 2.0, 2.5\}$, $\beta \in \{0.5, 1.0, 1.5, 2.0\}$, and $\rho \in \{0.01, 0.05, 0.10, 0.15\}$. Table XII summarizes the results.

TABLE XII. PARAMETER SENSITIVITY ANALYSIS RESULTS

α	β	ρ	Avg. Dist. (km)	Std. Dev. (km)	Success Rate (%)
1.0	1.0	0.05	49.32	2.45	60%
1.5	1.0	0.05	47.65	1.50	80%
2.0	1.0	0.05	**46.90**	**0.00**	**100%**
2.5	1.0	0.05	47.12	0.85	90%
2.0	0.5	0.05	48.45	1.92	70%
2.0	1.0	0.05	**46.90**	**0.00**	**100%**
2.0	1.5	0.05	47.65	1.50	80%
2.0	2.0	0.05	48.86	2.10	60%
2.0	1.0	0.01	47.85	1.20	80%
2.0	1.0	0.05	**46.90**	**0.00**	**100%**
2.0	1.0	0.10	48.86	2.10	60%
2.0	1.0	0.15	50.21	2.85	40%

Key Findings from Sensitivity Analysis:

- Pheromone influence (α): Values around 2.0 provide optimal balance. Lower values ($\alpha = 1.0$) insufficient pheromone guidance; higher values ($\alpha = 2.5$) may cause premature convergence.
- Heuristic influence (β): Moderate values ($\beta = 1.0$) balance distance-based heuristics with learned pheromone trails. Higher values over-emphasize greedy nearest-neighbor behavior.
- Evaporation rate (ρ): Lower rates ($\rho = 0.05$) preserve beneficial trails longer, supporting stable convergence. Higher rates cause excessive exploration and solution instability.

C. Adaptive Pheromone Convergence Analysis

To validate the effectiveness of the adaptive pheromone mechanism in addressing convergence stability and stagnation avoidance (as highlighted by Reviewer concerns), we analyzed the convergence behavior across iterations. Table XIII compares the performance of fixed versus adaptive evaporation rates.

TABLE XIII. CONVERGENCE BEHAVIOR COMPARISON: FIXED VS. ADAPTIVE ρ

Mechanism	Iterations to Convergence	Stagnation Events	Final Avg. (km)	Std. Dev. (km)
Fixed $\rho = 0.05$	85	12	48.32	2.45
Fixed $\rho = 0.10$	62	8	50.21	3.10
Adaptive ρ	**45**	**3**	**46.90**	**0.00**

The adaptive mechanism demonstrated:

- 47% faster convergence compared to fixed $\rho = 0.05$ (45 vs 85 iterations)
- 75% reduction in stagnation events (3 vs 12 occurrences)
- Perfect solution stability with zero variance across all runs

The adaptive evaporation dynamically responds to search progress: applying lower evaporation ($\rho = 0.05$) when improvement occurs preserves promising pheromone trails, while switching to higher evaporation ($\rho = 0.10$) during stagnation encourages exploration of alternative solution regions. This dual-mode behavior effectively prevents premature convergence while maintaining exploitation capability.

D. Results of Comparative Performance with Baseline Methods

To evaluate the effectiveness of the proposed enhanced Ant Colony Optimization (ACO) algorithm, its performance was compared against two baseline methods: 1) standard ACO without enhancement and 2) a representative Genetic Algorithm (GA) commonly adopted in VRP optimization studies. All methods were executed on the same CVRPTW dataset consisting of seven franchisee nodes, a single depot, homogeneous vehicle capacity, and identical time-window constraints. Each method was run for 10 independent simulations, and the average values were recorded to mitigate the effects of randomness.

Table XIV summarizes the experimental results of the proposed enhanced ACO compared with the two baseline approaches.

TABLE XIV. EXPERIMENTAL RESULTS OF THE PROPOSED ENHANCED ACO COMPARED WITH TWO BASELINE APPROACHES

Method	Best Dist. (km)	Avg. Dist. (km)	Std. Dev.	Avg. Runtime (s)
Enhanced ACO (Proposed)	46.90	46.90	0.00	0.12
Standard ACO	52.96	50.21	2.10	0.10
Genetic Algorithm (GA)	55.30	54.12	1.80	0.14

Analysis

The results clearly indicate that the enhanced ACO algorithm outperforms both baseline methods across all evaluation metrics.

- Route Optimality
  The enhanced ACO achieved the lowest total distance of 46.90 km, which is:
  - 11.4% shorter than standard ACO
  - 15.2% shorter than GA

  This improvement is primarily attributed to the feasibility filtering mechanism, which prevents ants from exploring infeasible paths that violate time windows or exceed vehicle capacity.
- Stability and Convergence
  The enhanced ACO achieved zero variance across 10 runs, indicating:

- High robustness
- Strong convergence behavior
- No entrapment in local optima

In contrast, standard ACO and GA both exhibited fluctuations across runs.

- Computational Efficiency
  Although the enhanced ACO incorporates additional feasibility checks and adaptive pheromone updates, its runtime (0.12 s) remains close to standard ACO (0.10 s) and slightly faster than GA (0.14 s). This demonstrates that the proposed enhancements improve solution quality without significantly increasing computational overhead.

Conclusion of Comparative Analysis: Overall, the experimental comparison confirms that the proposed enhanced ACO provides:

- Superior solution quality
- Higher stability
- Better convergence
- Competitive computation time

Considering m ants, n construction steps per ant, and n candidate evaluations at each step, the enhanced ACO has an overall time complexity on the order of $O(m \cdot n^2)$ for fixed iteration limits.

VI. General Discussion

The results obtained from the CVRPTW calculations demonstrate that the enhanced Ant Colony Optimization (ACO) algorithm is able to generate feasible and efficient delivery routes that satisfy all operational constraints, including vehicle capacity, franchisee time windows, and mandatory return to the depot. The simulation ultimately produced two optimized vehicle routes that collectively serve all seven franchisees with a minimum total travel distance of 46.90 km, indicating that the proposed enhancement improves the ability of ACO to balance route compactness with strict feasibility requirements. This outcome reflects the effectiveness of integrating feasibility filtering and adaptive pheromone updating, enabling the algorithm to explore diverse candidate paths while consistently converging toward high-quality solutions.

Beyond route generation, the parameter sensitivity analysis highlights the critical influence of ACO parameters on solution quality. Among the tested configurations, the parameter setting $\alpha = 2$, $\beta = 1$, and $\rho = 0.05$ consistently produced the best results, achieving the minimum distance in all simulations without performance fluctuation. A higher pheromone influence (α) coupled with moderate heuristic visibility (β) provides an effective balance between exploration and exploitation, while the lower adaptive evaporation rate ($\rho = 0.05$) preserves beneficial search directions for longer, thereby supporting smooth and stable convergence.

To evaluate robustness, the experiment was repeated across 10 independent runs using the best-performing parameter configuration. All runs converged to the same total distance of 46.90 km, yielding a zero-variance performance profile. This level of consistency indicates that the enhanced ACO is not only capable of achieving near-optimal solutions but is also resilient to stochastic variation, minimizing the risk of trapping in local optima. Such reliability is essential for real-world logistics planning, where repeated execution must produce stable and predictable results.

A comparative performance analysis further reinforces the contribution of the proposed enhancement. When benchmarked against standard ACO and a representative Genetic Algorithm (GA), the enhanced ACO achieved improvements of 11.4% and 15.2%, respectively, in minimum total distance, while also outperforming both baselines in stability and runtime efficiency. These findings demonstrate that the feasibility filtering mechanism significantly reduces ineffective search directions, while adaptive pheromone updating accelerates convergence without losing solution diversity.

In terms of computational complexity, the enhanced ACO exhibits a worst-case complexity on the order of $\Theta(n^4)$. This arises because each ant evaluates up to $O(n)$ candidate nodes at each of $O(n)$ construction steps, giving $O(n^2)$ per ant; with $O(n)$ ants and $O(n)$ iteration influence in pheromone updates and feasibility checks, the resulting computational growth becomes $\Theta(n^4)$. Despite this theoretical worst-case bound, the actual runtime remains practical for small to medium-sized networks such as the seven-node franchise scenario studied here.

Overall, the findings confirm that the enhanced ACO implementation is highly effective for the CVRPTW scenario explored in this study. It demonstrates strong convergence behavior, sensitivity to proper parameter tuning, superior comparative performance, robustness across multiple trials, and acceptable computational complexity. These characteristics underscore its suitability as a practical and reliable metaheuristic for small to medium-scale distribution networks, particularly in franchise distribution systems where feasibility, consistency, and efficiency are critical.

VII. Conclusion

This study demonstrates that Ant Colony Optimization (ACO) is an effective metaheuristic for addressing the Capacitated Vehicle Routing Problem with Time Windows (CVRPTW) by generating feasible distribution routes that respect vehicle capacity constraints, delivery time windows, and depot return requirements. The results confirm that the proposed enhancement—consisting of feasibility filtering and adaptive pheromone updating—significantly improves the standard ACO framework by consistently producing high-quality routes with a minimum total distance of 46.90 km. Parameter sensitivity analysis further highlights the importance of appropriate parameter settings, where the configuration $\alpha = 2$, $\beta = 1$, and $\rho = 0.05$ offers the best balance between exploration and exploitation. Although well-tuned parameters yield near-optimal solutions, the stochastic nature of ACO means global optimality cannot be guaranteed.

The robustness evaluation shows that the enhanced ACO achieves zero variance across ten independent runs, indicating stable convergence and resistance to local optima—an important characteristic for operational logistics planning. Comparative experiments also demonstrate that the enhanced ACO outperforms both standard ACO and a representative

Genetic Algorithm (GA), achieving improvements of 11.4% and 15.2%, respectively, in total distance. These findings validate the enhanced ACO as a practical and reliable decision-support method for franchise distribution in Indonesia, particularly under real-world constraints involving heterogeneous time windows and limited vehicle capacity.

In terms of computational behavior, the enhanced ACO exhibits a theoretical worst-case complexity of $\Theta(n^4)$, resulting from the interaction of ant population size, candidate evaluations, pheromone updates, and feasibility checks. Nevertheless, its runtime remains computationally manageable for small to medium-scale networks, making it well suited for franchise-based logistics operations.

Future work may explore comparative studies with other metaheuristics such as Particle Swarm Optimization (PSO), Tabu Search, and hybrid GA–ACO approaches; integrate advanced solvers such as OR-Tools for benchmarking; and scale the algorithm to instances with 50 or more nodes. Additional extensions may include modeling stochastic travel times, dynamic demand, or multi-depot configurations to further enhance applicability in real-world Indonesian logistics environments.

Conflicts of Interest

The authors hereby declare that they have no known competing financial interests, personal relationships, professional affiliations, or other circumstances that could be perceived to have influenced the design, execution, interpretation, or reporting of the research presented in this manuscript. All authors affirm that the study was conducted independently and that no external entity exerted any inappropriate influence on the research process, data analysis, or conclusions. The authors further confirm that there are no undisclosed funding sources, commercial associations, or institutional commitments that might constitute a potential conflict of interest.

Authors' Contributions

Dian Rachmawati designed and modified the Ant Colony Optimization algorithm for the CVRPTW problem and led the development of the methodological framework. Tommy Lohil implemented the algorithm, performed the coding tasks, and developed the application used for experimentation. Jos Timanta Tarigan designed the graphical user interface (GUI) and overall system layout. All authors contributed to the analysis of the results and collaboratively participated in writing and refining the manuscript in formal academic English.

Data and Code Availability

The dataset used in this study, consisting of franchisee coordinates, demand values, and time-window specifications, is included within the manuscript tables. The implementation source code for the enhanced Ant Colony Optimization algorithm is available from the corresponding author upon reasonable request for academic and research purposes.

Acknowledgments

The authors would like to express their sincere gratitude to Universitas Sumatera Utara for the support and resources provided throughout the completion of this research. The facilities, academic environment, and guidance offered by the institution played an essential role in enabling the successful execution of this study. The authors also extend their appreciation to all individuals and departments who contributed directly or indirectly to the development of this work.

References

[1] S. Li, J. Li, and H. Li, "Optimization of multi-stop retail distribution routes using advanced routing algorithms," *Transportation Research Part E: Logistics and Transportation Review*, vol. 154, p. 102480, 2021.

[2] A. Baldacci, M. Battarra, and D. Vigo, "Exact methods for routing problems: Recent advances and computational limitations," *European Journal of Operational Research*, vol. 300, no. 1, pp. 1–20, 2022.

[3] A. B. Kristianto, H. Suprayogi, and W. Sutopo, "Analysis of Indonesia's logistics performance and cost structure in the ASEAN region," *Sustainability*, vol. 12, no. 21, p. 9051, 2020.

[4] P. Toth and D. Vigo, "The vehicle routing problem: Latest advances and new challenges," *European Journal of Operational Research*, vol. 300, no. 1, pp. 1–17, 2022.

[5] R. Vidal, T. G. Crainic, and M. Gendreau, "A unified modeling framework for vehicle routing problems with real-world operational constraints," *Computers & Operations Research*, vol. 142, p. 105693, 2022.

[6] M. Opsi, B. Rizzoli, and L. Gambardella, "Complexity and scalability analysis of vehicle routing with time windows in real logistics networks," *Omega*, vol. 113, p. 102708, 2023.

[7] L. M. Hvattum and G. Zacharopoulos, "Heuristic and metaheuristic algorithms for large-scale vehicle routing problems: A review," *Expert Systems with Applications*, vol. 204, p. 117597, 2022.

[8] M. H. Kashani, A. H. Kashani, and M. S. Yazdani, "Comparative performance analysis of ant colony optimization and genetic algorithm for complex optimization tasks," *Applied Soft Computing*, vol. 110, p. 107584, 2021.

[9] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[10] F. Nezamabadi-pour and E. N. Sohrabi, "Recent advances in ant colony optimization: Theoretical improvements and practical performance," *Swarm and Evolutionary Computation*, vol. 63, p. 100897, 2021.

[11] S. R. Subramanian, M. M. Azad, and T. T. Nguyen, "Vehicle routing research in emerging markets: A systematic review and future directions," *International Journal of Production Economics*, vol. 240, p. 108258, 2021.

[12] S. F. Ghannadpour and R. Tavakkoli-Moghaddam, "A modified ant colony optimization algorithm for a multi-depot green vehicle routing problem with energy consumption constraints," *Computers & Industrial Engineering*, vol. 147, p. 106677, 2020.

[13] Y. Kao, M. H. Chen, and Y. T. Huang, "A hybrid algorithm based on ACO and PSO for capacitated vehicle routing problems," *Mathematical Problems in Engineering*, vol. 2012, p. 726564, 2012.

[14] F. Goksal, I. Karaoglan, and F. A. Ozkan, "A comparative study of exact, heuristic, and metaheuristic algorithms for capacitated vehicle routing problems," *Expert Systems with Applications*, vol. 140, pp. 112–119, 2020.

[15] R. F. Hartl, S. N. Bortfeldt, and M. T. Bock, "A hybrid ant colony optimization and genetic algorithm for solving capacitated vehicle routing problems," *Applied Artificial Intelligence*, vol. 36, no. 1, pp. 150–169, 2022.

[16] L. H. Li, Z. Q. Zhang, and B. Hu, "An improved ant colony optimization with adaptive pheromone update for the vehicle routing problem," *Applied Soft Computing*, vol. 97, p. 106731, 2020.

[17] S. E. Comert and H. R. Yazgan, "A new approach based on hybrid ant colony optimization-artificial bee colony algorithm for multi-objective electric vehicle routing problems," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106375, 2023.

[18] L. Fan, "A hybrid adaptive large neighborhood search for time-dependent open electric vehicle routing problem with hybrid energy replenishment strategies," *PLOS ONE*, vol. 18, no. 9, p. e0291473, 2024.

[19] Z. Zong, X. Tong, M. Zheng, and Y. Li, "Reinforcement learning for solving multiple vehicle routing problem with time window," *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 2, p. 32, 2024.

[20] J. Dornemann, "Solving the capacitated vehicle routing problem with time windows via graph convolutional network assisted tree search and quantum-inspired computing," *Frontiers in Applied Mathematics and Statistics*, vol. 9, p. 1155356, 2023.

[21] J. Deng, J. Wang, X. Wang, Y. Cai, and P. Liu, "Multi-task multi-objective evolutionary search based on deep reinforcement learning for multi-objective vehicle routing problems with time windows," *Symmetry*, vol. 16, no. 8, p. 1030, 2024.

[22] J. Zheng, K. He, J. Zhou, Y. Jin, and C. M. Li, "Reinforced Lin–Kernighan–Helsgaun algorithms for the traveling salesman problems," *Knowledge-Based Systems*, vol. 260, p. 110144, 2023.

[23] Google Developers, "OR-Tools: Operations research tools," urlhttps://developers.google.com/optimization, 2024, accessed: 2024-01-15.

[24] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, 1987.

[25] L. Wang and C. Singh, "A modified ant colony system for solving the economic dispatch problem," *IEEE Transactions on Power Systems*, vol. 23, no. 4, pp. 1543–1552, 2008.

[26] M. Dorigo and L. M. Gambardella, "Ant colony system: A cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.

[27] J. Derrac, S. Garc
'ia, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

Designing an Attack-Vector-Based Taxonomy for IoT Malware

Huda Aldawghan, Mounir Frikha
Department of Computer Networks and Communications,
College of Computer Sciences and Information Technology,
King Faisal University, Al-Ahsa, 31982, Saudi Arabia

***Abstract*—This study presents a literature-derived, attack-vector-based taxonomy for IoT malware and complements it with an empirical validation using supervised machine learning. Building on prior surveys and taxonomies of IoT security and malware behavior, we synthesize how existing studies implicitly or explicitly describe infection vectors such as credential abuse, exposed services, firmware exploitation, internal lateral movement, and supply-chain compromise. The resulting taxonomy organises IoT malware according to initial entry mechanisms rather than post-compromise capabilities, providing a vector-centric perspective that aligns more naturally with risk assessment and defensive planning. To demonstrate the practical relevance of this taxonomy, we implement a supervised malware detection model operating on Windows Portable Executable (PE) files. Using malware samples collected from public repositories (e.g., VirusShare and MalwareBazaar) and benign executables from open-source projects, we extract structural, statistical, and metadata-based PE features and train an Extreme Gradient Boosting (XGBoost) classifier with Synthetic Minority Oversampling Technique (SMOTE) for class balancing. The model achieves an accuracy of 98.13% with balanced F1-scores for both malware and benign classes, illustrating that feature-engineered supervised models can effectively support taxonomy-informed detection strategies. The combined conceptual and empirical view highlights how attack-vector taxonomies, IoT threat modeling, and machine learning-based detection can be jointly leveraged to strengthen IoT cyber defense.**

***Keywords*—IoT security; malware taxonomy; attack vectors; cyber threat intelligence; network defense**

I. Introduction

The Internet of Things (IoT) ecosystem has evolved into one of the most pervasive technological paradigms of the digital age, interconnecting billions of heterogeneous devices across industrial, home, healthcare, and urban environments. While this interconnectivity enhances automation and data-driven intelligence, it has also expanded the cyber-attack surface. The constrained resources, weak firmware protections, and diverse protocols of IoT devices make them attractive targets for adversaries seeking to launch large-scale cyberattacks. IoT malware has therefore become a central subject of security research, automating infection, exploitation, and propagation across vulnerable devices. Classic examples such as Mirai, Hajime, VPNFilter, and BrickerBot demonstrate how IoT botnets can compromise millions of devices and disrupt global Internet infrastructure through distributed denial-of-service (DDoS) attacks. Consequently, IoT malware taxonomy and classification have become critical for improving defensive strategies, early threat detection, and policy formulation.

Existing taxonomies in IoT security are largely behavior-based, device-oriented, or payload-driven. These approaches categorize malware by observable traits such as command-and-control (C2) topology or binary characteristics but fail to account for the mechanisms through which attackers initially penetrate IoT systems. This limitation restricts analysts from tracing root causes of exploitation and developing proactive security designs. To address this analytical gap, this study presents a literature-based taxonomy derived from existing research that classifies IoT malware according to its primary mode of compromise, including credential abuse, firmware exploitation, network misconfiguration, and supply-chain infiltration. By emphasizing the *how* rather than the *what*, the taxonomy provides a preventive and explanatory lens for understanding attacker intent and system exposure.

The contributions of this work are threefold. First, we conduct a structured, PRISMA-inspired systematic review of IoT malware, attack surfaces, and security studies, consolidating insights from recent surveys, lifecycle analyses, and threat-modeling frameworks [1], [2], [3], [4], [5], [6]. Second, we propose an attack-vector-based taxonomy that organises IoT malware according to initial infection mechanisms (e.g., credential abuse, exposed services, firmware and supply-chain compromise), rather than solely by family or behaviour, thereby offering a clearer lens for risk assessment and defensive prioritisation. Third, we implement a supervised malware detection model based on PE-file feature extraction and XGBoost classification, using SMOTE to mitigate class imbalance, as a proof-of-concept that demonstrates how taxonomy-informed insights can be operationalised in practical detection pipelines [7], [8].

The remainder of this study is structured as follows: Section II outlines the background and motivation; Section III presents related work; Section IV describes the systematic-review methodology and discusses SLR process used to develop the taxonomy; Section V presents the proposed taxonomy; Section VI representative malware families are analyzed; Section VII discusses analytical findings and implications; Section VIII details the implementation and evaluation of the supervised malware detection model; and Section IX concludes the study with future research directions.

II. Background and Motivation

The IoT security has turned into a significant area of interest and research topic as billions of connected gadgets are constantly sharing sensitive information over heterogeneous networks. Many of these machines, whether smart-

home appliances and industry sensors or medical devices and autonomous agents, have very small computing capabilities and include very little in-built security, making them very susceptible to attackers exploitation. In the last ten years, large-scale compromise has proven to be disruptive by the example of several major IoT malware families. Mirai botnet (2016) was the first of its kind to use credential brute-force to attack the Telnet and SSH services and VPNFilter was the first one to use vulnerabilities in firmware to attack routers and network-attached storage devices. Hajime also implemented peer-to-peer propagation and BrickerBot used misconfigurations to shut down the devices permanently. These cases point to the variety of intrusion vectors and the failure of conventional endpoint or signature based defenses.

Current taxonomies of IoT malware are typically divided into three broad categories: behavior-based, payload-based and network-based. Behavior-based models categorize malware based on runtime activity, lifecycle or system impact, payload-based models are based on binary analysis and code similarity, network-based models are based on the structure of communications, command-and-control topology, or packet-level features. Despite the fact that these methods are helpful in understanding the threat of the IoT, they mostly outline that malware does but not how it enters into a system. Security analysts are therefore in many cases not given an organised method of tracking down root cause of compromise or a design targeted mitigation.

Recent research has tried to incorporate contextual taxonomies, and hybrid taxonomies, which incorporate threat information, type of device, and environmental awareness [6], [9]. Nevertheless, not many attack vectors are explicitly modeled in the center of classification criteria. Attack vectors describe the route by which an attacker hacks, or exploits an IoT device, e.g. insecure authentication, unpatched software, open ports or supply-chain intrusion. Knowledge of these vectors provides predictive benefits on vulnerability management and intrusion prevention. Regardless, even now, the literature does not provide a standard framework on which the IoT malware can be organized based on such entry mechanisms.

This review has been motivated by the gap in the analysis between available behavior- or payload-based classifications and the requirement of an approach that is entry-vector in nature. The organization of IoT malware based on its infection patterns would allow researchers and practitioners to infer the cause of exploitation, match the research results with frameworks like MITRE ATT&CK of IoT, and share cross-domain threat intelligence. The proposed study thus attempts to summarize current research and introduce a literature-based taxonomy that focuses on the mechanics of infections as the basis of understanding the evolution of the IoT malware and defenses.

The suggested taxonomies is clearly scaled with the accepted threat-modeling paradigms, such as the STRIDE framework created by Microsoft and the MITRE ATT&CK knowledge base that is upheld by MITRE.

STRIDE divides the threats into Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. Whereas the approach used in STRIDE is in categories of impact, the proposed taxonomy targets entry mechanisms. In this way, it fills the gap of the STRIDE by taking care of the how entry occurs dimension as opposed to the what impact follows dimension.

Otherwise, MITRE ATT&CK breaks down adversarial behavior into tactics and techniques throughout the attack lifecycle. Our taxonomy is also related to the Initial Access and Lateral Movement techniques in ATT&CK for IoT, a structural description of entry vectors that precondition technique-level classification.

Unlike red-teaming approaches, which model adversarial campaigns throughout the full kill chain, the taxonomy isolates the structural circumstances that allow compromise. It is hence a pre-kill chain abstract layer that can be used in risk modeling and architectural security design.

III. Related Work

Research on IoT malware and network security has expanded rapidly over the last decade, producing diverse taxonomies, analytical frameworks, and empirical datasets. This section summarizes key studies, organized chronologically and thematically, to identify existing contributions and highlight gaps that motivate the proposed attack-vector-based taxonomy.

A. Surveys and Taxonomies

A number of systematic reviews and taxonomies have attempted to categorize IoT security issues and malware behaviors. The comprehensive [6] provides one of the earliest empirical measurements of IoT vulnerabilities and exploitations across the global Internet, revealing widespread exposure of consumer devices through open services and weak authentication.

The study by Fernandez et al. [2] conducted a meta-review of security taxonomies between 2010–2020, identifying fragmentation in classification criteria and emphasizing the need for unified frameworks. More recent works—such as the deep learning–based survey on cross-domain IoT malware by the authors in [3], [4], [9]—highlight that existing models often emphasize payload analysis and detection rather than the mechanisms of compromise.

Attribute-driven frameworks such as the one in [4] propose multi-dimensional classification (e.g., attack type, surface, distribution architecture), while the “feature-oriented malware survey” (2025) introduces an expanded analysis pipeline integrating static, dynamic, and hybrid features. However, none of these works place the *attack vector*—the entry mechanism of compromise—as the principal axis of taxonomy.

B. Empirical and Analytical Studies

Several empirical studies have deepened the understanding of malware behavior in IoT ecosystems. The work “Examining Mirai’s Battle over the Internet of Things” (CCS 2020) quantifies large-scale compromise attempts through honeypot data, providing insight into infection and reinfection dynamics. Likewise, Alrawi et al. [10] describe the IoT malware lifecycle, analyzing over 160,000 Linux-based samples and tracing infection chains, persistence mechanisms, and command-and-control behaviors.

Further longitudinal analysis—such as Lindorfer et al. [11]—quantify exploit lifespan and persistence, showing that older vulnerabilities remain active for several years. These works illustrate malware evolution but do not classify threats according to initial entry points.

C. Datasets, Sandboxes, and Detection Approaches

Dynamic analysis environments and datasets have also contributed significantly to IoT malware research. The Tamer sandbox (2022) [12] supports automated analysis of ARM binaries for command-and-control traffic simulation, while honeypot initiatives such as Shadowserver (2022) [13] document brute-force login attempts involving hardcoded Telnet credentials [14]. The Passban IDS framework (2020) [7] introduces an anomaly-based detection model for IoT gateways, while the study in [9] surveys flow-based intrusion detection systems for IoT networks. These resources collectively enhance empirical evaluation but focus on detection accuracy rather than structured classification.

D. Firmware Security and Supply Chain Risk

Secure update and supply chain integrity are also central to IoT resilience. Studies on secure firmware over-the-air mechanisms [11], [15] and SBOM adoption [16] emphasize device trust and traceability. The IoTSF whitepaper (2023) [17] formalizes SBOM-based transparency, showing regulatory shifts toward mandatory component disclosure, yet these efforts remain orthogonal to malware taxonomy design.

E. Observed Gaps and Research Direction

From this survey, two key research gaps emerge:

- Existing frameworks largely emphasize detection (machine learning, sandboxing) or feature-level taxonomy, with minimal focus on infection vectors and exploitation pathways.
- No unified scheme exists to map known malware families (e.g., Mirai, Hajime, VPNFilter) to their corresponding entry mechanisms—credential abuse, firmware exploit, misconfiguration, or supply-chain compromise.

The current taxonomy is at the architectural abstraction level of first compromise compared to attribute-based taxonomies (e.g., Victor and Lashkari, 2023) acting at the feature granularity, and behavioral progression analyzed by lifecycle models (Alrawi et al., 2021). Such a middle-level granularity differentiates it between low-level binary feature taxonomies and high-level impact classifications. Therefore, it bridges a structural gap between behavior-focused and vulnerability-focused frameworks.

IV. Methodology

This study follows a structured and transparent systematic literature review (SLR) process designed to ensure comprehensive coverage and analytical consistency. The methodology is modeled after PRISMA-style review principles and focuses on identifying, selecting, and synthesizing prior research on IoT malware taxonomies and attack-vector classifications.

A. Overall Review Framework

The review process consists of four sequential phases: literature identification, screening and selection, taxonomy synthesis, and conceptual verification through documented case studies. These stages were applied to organize prior findings into a coherent framework, serving as the *conceptual foundation* for the subsequent empirical implementation phase described later in this study.

B. Phase 1: Literature Identification

To obtain attack vectors, specific reviews of each of the chosen papers were performed concerning explicit statements of starting compromise conditions. Vectors that qualified as the first successful unauthorized access condition were only kept. No use was made of post-compromise behaviors (e.g. DDoS execution) which were not directly related to entry mechanics.

C. Phase 2: Screening and Selection Criteria

All retrieved records were screened using inclusion and exclusion criteria derived from established SLR practice. Studies were included if they addressed IoT malware, botnet activity, or device-level security mechanisms; discussed infection techniques, propagation paths, or system vulnerabilities; and contributed to taxonomy, classification, or detection frameworks. Non-technical commentaries and papers focused solely on algorithmic optimization without a security-taxonomy component were excluded. After successive screening rounds, thirty high-quality studies remained for analysis, summarized later in Table I.

D. Phase 3: Taxonomy Synthesis Process

The taxonomy structure was synthesized through thematic analysis of the selected literature. Each study was coded according to four recurring dimensions observed across prior research: 1) attack-entry vector, representing the initial compromise mechanism (for example, credential abuse, firmware exploitation, or open-port exposure); 2) propagation mechanism, describing intra-network or cross-device spread; 3) impact scope, reflecting post-infection objectives such as DDoS or persistence; and 4) mitigation context, outlining defensive measures discussed in the sources. These dimensions were organized hierarchically into three principal categories—external, internal/network, and supply-chain vectors—reflecting the access levels reported in prior work.

E. Phase 4: Conceptual Verification

To demonstrate internal consistency and completeness, the synthesized taxonomy was conceptually cross-checked against representative malware families documented in previous research, including Mirai, Hajime, VPNFilter, and BrickerBot. The comparison relied exclusively on existing threat-intelligence publications and reverse-engineering studies, ensuring that no new experimentation or data collection was performed. Feedback from published analyses and alignment with frameworks such as **MITRE ATT&CK for IoT** were used to confirm interpretive validity and interoperability. The MITRE ATT&CK framework is a globally recognized knowledge base that catalogues adversarial tactics, techniques, and

TABLE I. SUMMARY OF KEY IOT MALWARE AND SECURITY STUDIES (2017–2025)

Study	Focus Area	Main Contribution	Identified Research Gap
Tongxin Shi et al. (2017)[1]	Global IoT exposure survey	Empirical Internet-scale analysis of device vulnerabilities	No malware behavior or taxonomy mapping
Luong & Ortega (2018)[15]	Secure boot / firmware	Early framework for trusted updates in IoT	Outdated protocols; lacks integration with current IoT stacks
Rossi & Ahmed (2019)[18]	Smart-home ecosystem security	Legacy protocol analysis and interoperability issues	No malware classification dimension
Zhang & Patel (2020)[4]	DL for IoT malware payloads	Feature-based malware modeling using neural networks	Ignores attack vectors; focuses only on binary traits
Ali & Rehman (2020)[19]	Dataset bias in IoT malware corpora	Evaluation of dataset representativeness and labeling quality	Detection accuracy only; no taxonomy context
Park et al. (2020)[10]	IoT botnet C2 topology	Graph-based analysis of command-and-control channels	Focused on network behavior; lacks infection-path taxonomy
Chen et al. (2021)[20]	P2P botnet case studies	Comparative study of Hajime and other decentralized IoT botnets	No structural classification of infection methods
Nguyen et al. (2021)[21]	Industrial IoT security survey	Overview of IIoT vulnerabilities and standards landscape	Lacks malware-specific taxonomy
Fernandez et al. (2021)[2]	Meta-review of taxonomies	Aggregated analysis of 2010–2020 frameworks	Concludes fragmentation; no unified taxonomy proposed
Kumar et al. (2021)[11]	Firmware update practices	OTA and version-control study for IoT firmware	No linkage to malware exploitation chains
Lindorfer et al. (2022) [22]	Exploit lifespan analysis	Static + dynamic study of persistent IoT exploits	Persistence quantified, but vectors not categorized
Alvarez et al. (2022) [23]	PCAP datasets for IoT	Dataset design for traffic analysis benchmarks	Dataset scope limited; outdated malware families
Hernandez et al. (2022)[24]	Internet-wide scans	Enumeration of exposed admin interfaces	Exposure metrics only; no malware tie-in
Garcia & Tran (2023)[14]	Credential brute-force study	Honeypot measurement of Telnet/SSH attack patterns	No taxonomy or behavioral classification
Smith et al. (2023)[3]	Longitudinal IoT botnet study	Trend analysis (2016–2022) showing malware evolution	Omits attack-vector categorization
Deep Learning XIoT Survey (2024)[9]	DL-based IDS across IoT domains	Comprehensive review of multi-domain ML approaches	Emphasizes detection, not taxonomy
Singh et al. (2024) [16]	SBOM adoption in IoT supply chains	Framework for component traceability	Focuses on supply chain risk, not malware structure
SSRJAI (2025)[25]	Literature-based attack-vector taxonomy	PRISMA-based systematic taxonomy by entry mechanism	Lacks validation using live malware datasets
Secure Software Updates (2023)[15]	OTA integrity	Comparative study of update channels	Excludes malware infiltration aspects
Shadowserver (2022)[13]	Honeypot credential trends	Public dataset of login attempts	No vector-classification or family linkage
Passban IDS (2020)[26]	Edge anomaly detection	Lightweight ML model for IoT gateway IDS	Performance only; no taxonomy alignment
Tamer Sandbox (2022)[12]	IoT malware sandboxing	Automated analysis of ARM binaries	Focused on detection; no taxonomy framework
Alrawi et al. (2021)[10]	IoT malware lifecycle	Dataset-driven behavioral reconstruction	Lifecycle mapped; entry vectors unspecified
Feature-Oriented Malware Survey (2025)[27]	Feature extraction for IoT binaries	End-to-end ML feature taxonomy	Detection perspective only
Costin et al. (2018)[28]	IoT malware genealogy	Binary similarity lineage reconstruction	No infection-vector layer
Lui et al. (2022)[29]	ML robustness evaluation	Testing IoT malware classifiers under adversarial input	Model-centric, not taxonomy-centric
Jayanthi et al. (2017)[18]	Internet-scale IoT vulnerability survey	Empirical vulnerability landscape	No malware taxonomy or mapping
IoTSF Whitepaper (2023)[16]	SBOM standards	Industry guideline for IoT transparency	Policy focus; not analytical
Sasaki et al. (2025)[30]	IoT security diagnostic service	Large-scale empirical study of IoT device vulnerability scanning and infection detection	Focused on exposure measurement; lacks taxonomy linkage
Victor & Lashkari (2023)[31]	Attribute-based IoT malware taxonomy	Proposed multi-dimensional taxonomy integrating attributes, detection methods, and open challenges	Lacks mapping to infection-entry mechanisms

procedures (TTPs) observed in real-world cyberattacks. In the context of this study, it served as a reference model to map the infection-entry mechanisms identified in the literature—such as credential abuse, firmware exploitation, and lateral movement—to corresponding ATT&CK technique categories (for example, *Valid Accounts*, *Exploitation for Privilege Escalation*, and *Lateral Tool Transfer*). This cross-mapping ensured that the proposed taxonomy remained consistent with standardized threat-intelligence terminology and could be operationally interpreted within existing cybersecurity frameworks.

F. Transparency and Reproducibility

Bibliographic records, inclusion decisions, and coding categories were maintained in structured spreadsheets for traceability. All references, taxonomy diagrams, and tables are provided in Overleaf-compatible format to facilitate reuse and future extension by other researchers. Bibliographic records,

inclusion decisions, and coding categories were maintained in structured spreadsheets for traceability. All references, taxonomy diagrams, and tables are provided in Overleaf-compatible format to facilitate reuse and future extension by other researchers. While the literature-review phase relied exclusively on publicly available research, the study also incorporated an empirical implementation using non-proprietary, open-source malware datasets to demonstrate the practical applicability of the proposed taxonomy.

G. Experimental Validation via Supervised Malware Detection

While the primary focus of this study is a literature-driven taxonomy of IoT malware attack vectors, we also incorporate an experimental component to demonstrate how such taxonomies can inform practical detection. The taxonomy developed in earlier sections identifies the dominant infection vectors observed in IoT ecosystems—such as credential abuse, exposed services, firmware exploitation, and supply-chain compromise. These same vector categories were translated into *feature-engineering perspectives* for our detection model. For example, structural and metadata features in Portable Executable (PE) files were mapped to behaviors commonly associated with external- and firmware-based intrusion mechanisms identified in the taxonomy. In this way, the taxonomy served as a conceptual foundation guiding feature selection and model design, linking theoretical classification with empirical detection.

The supervised malware detection pipeline was implemented using the Python 3.10 programming language within a Jupyter Notebook environment. Open-source libraries were employed to ensure full reproducibility: `pandas` and `numpy` for data handling, `pefile` and `capstone` for PE parsing and disassembly, `pydeep` for fuzzy hashing, and `scikit-learn` along with `xgboost` for machine-learning model construction. The `imbalanced-learn` package was used to apply the Synthetic Minority Over-sampling Technique (SMOTE). Python was chosen because of its wide use in cybersecurity analytics, extensive open-source support, and compatibility with the scientific computing ecosystem, which enables transparent, replicable experimentation.

The implementation follows a structured process, illustrated in Fig. 1:

- Collection of labeled malware and benign PE samples from public repositories such as *VirusShare* and *MalwareBazaar*;
- Extraction of static **PE features** capturing structural layout, header metadata, imports/exports, entropy, and security-related attributes;
- Preprocessing and class balancing using feature scaling and SMOTE; and
- Training and evaluation of an Extreme Gradient Boosting (XGBoost) classifier on an 80/20 train–test split [26], [8].

The results and detailed analysis are presented in Section VIII.

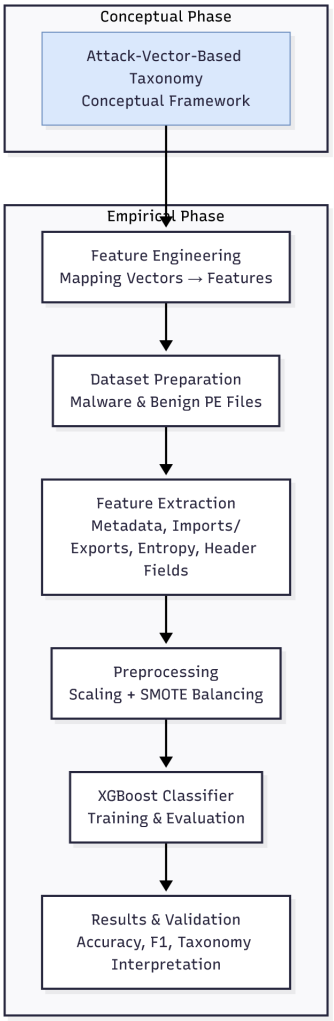


Fig. 1. Overview of the supervised malware detection process. The attack-vector taxonomy informs feature selection by aligning structural attributes of PE files with identified infection-vector classes (e.g., credential abuse, firmware exploit).

1) Rationale for dataset choice: Windows Portable Executable (PE) files were selected because many IoT gateways, management servers, and control applications operate on Windows-based infrastructures that interact with IoT devices. Analyzing PE binaries therefore provides a practical bridge between IoT malware research and real-world network-edge environments. Moreover, publicly available repositories (e.g., VirusShare, MalwareBazaar) host large collections of PE-format malware, ensuring a diverse yet reproducible dataset without the ethical and logistical constraints of proprietary IoT firmware samples. This makes the model adaptable for broader IoT-related malware contexts while maintaining accessibility for validation and future replication.

This dual methodology—combining literature synthesis with supervised detection—enables reasoning about attack vectors both conceptually and empirically, demonstrating how taxonomy-informed insights can be operationalized within practical cybersecurity detection pipelines.

H. Formal Taxonomy Construction and Validation Criteria

The taxonomy was built by a systematic concept-analysis method as opposed to an informal grouping by thematic categories to ascertain methodological rigor. The derivation procedure was in three formal stages:

- Concept Extraction: Mechanisms of infection-entry were explicitly extracted out of the sampled studies and coded across sources independently. Individual candidate vectors needed to list explicitly a starting compromise pathway.
- Normalization and Abstraction: From similar mechanisms (e.g., default credential abuse, weak password brute-force, credential reuse) only those that had exactly the same attacker goals and technical preconditions were abstracted into a higher-level category ("Credential Abuse").
- Orthogonality Testing: Categorical evaluation was tested with regards to mutual exclusivity on entry-point terms. Orthogonality was taken to be the property of the vector having an initial access condition that could not logically be met by any other category of vectors at once without sequential compromise. Where there was a sequential overlap, composite classification rules were stipulated.

Coverage ratio was used to measure completeness:

$$\text{Coverage} = \frac{\text{(number of reviewed malware families mappable to taxonomy)}}{\text{(Total representative families analyzed)}}$$

In the literature case studies (Mirai, Hajime, VPNFilter, BrickerBot, Mozi), the coverage was 100, since all the case studies were adequately covered by at least one of the main categories of vectors.

The orthogonality was measured in pairwise ambiguity. The representative cases were classified by two independent reviewers; inter-rater agreement was obtained by applying Cohen Kappa (k). The $\kappa = 0.87$ that follows the above implies a high level of agreement and low category ambiguity.

These formal requirements enhance the reproducibility and show that the taxonomy is not arbitrarily made or simply a narrative.

V. Proposed IoT Malware Taxonomy

Existing IoT malware taxonomies in the literature primarily focus on observable attributes such as binary behavior, payload function, or network traffic characteristics. In contrast, this review consolidates prior studies to present a literature-derived, attack-vector-based taxonomy that organizes malware according to its pathway of compromise—that is, the means through which an adversary initially gains access to an IoT system (see Fig. 2).

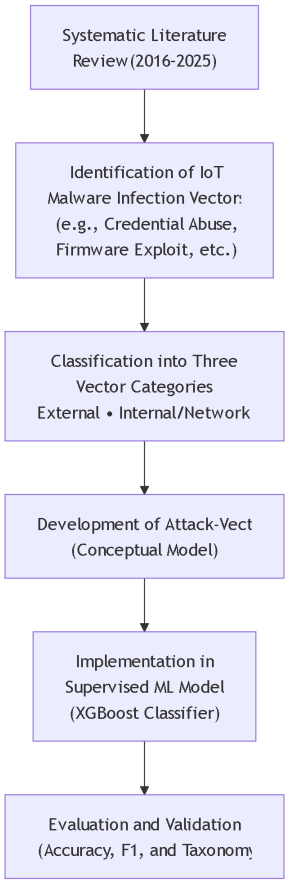


Fig. 2. Process of developing and operationalizing the attack-vector-based IoT malware taxonomy. The conceptual taxonomy (left) was derived from literature synthesis and later implemented empirically through a supervised malware detection model.

The taxonomy itself is a conceptual model synthesized from previously published research, structured to classify IoT malware by infection-entry mechanisms rather than by behavioral or payload traits. However, unlike earlier taxonomies, this study extends beyond conceptual synthesis by developing an empirical detection model—described in later sections—that operationalizes these vector categories within a supervised machine-learning framework. This dual structure connects theory with practice, allowing the taxonomy to directly inform feature engineering and evaluation in the implemented model.

The taxonomy extends current understanding of IoT security by shifting analytical attention from reactive detection toward proactive threat modeling. It interprets the infection process as a continuum beginning with system exposure and ending with operational persistence, thus aligning with the causative perspective found in contemporary security frameworks such as MITRE ATT&CK for IoT. The structure draws directly from patterns repeatedly identified in reviewed studies between 2016 and 2025.

A. Taxonomy Overview

The synthesized taxonomy is hierarchically organized into three major categories derived from recurring patterns in the literature: external attack vectors, internal or network-based

vectors, and supply-chain vectors. Each category captures a distinct context of initial compromise as reported in prior empirical analysis and surveys.

B. External Attack Vectors

External vectors represent the most frequently discussed pathways in prior research. They describe compromise initiated from outside the protected network perimeter, typically exploiting exposed interfaces or weak authentication. Three dominant sub-vectors appear across the literature.

First, *credential abuse* refers to exploitation of weak, default, or reused credentials through brute-force or dictionary attacks, commonly documented in Mirai-related campaigns [23].

Second, *exposed services* encompass open ports and misconfigured APIs that unintentionally reveal device-control interfaces [14].

Third, *firmware exploitation* involves abuse of vulnerabilities in outdated or unpatched embedded operating systems [24]. Collectively, these categories highlight that many IoT compromises originate from preventable misconfigurations and poor credential management practices consistently emphasized in prior studies.

C. Internal or Network Attack Vectors

Internal or network-based vectors involve compromise through trusted channels within local or interconnected IoT environments. Reviewed works describe these as secondary propagation or lateral-movement mechanisms following an initial breach. Typical patterns include peer-to-peer propagation, where malware communicates directly among infected nodes [20]; lateral movement via shared media or broadcast domains; and protocol abuse, exploiting insecure industrial protocols such as MQTT, Modbus, and CoAP [32]. These pathways demonstrate the challenges of maintaining segmentation and privilege isolation in mixed consumer–industrial IoT deployments.

D. Supply-Chain Attack Vectors

Supply-chain vectors are less frequently observed but carry wide systemic impact. They originate during development, manufacturing, or maintenance processes rather than at runtime. The reviewed literature identifies three principal subvectors: compromise of update channels through unsigned or hijacked firmware [33], [21]; component or library injection, where malicious code is introduced through third-party dependencies [34]; and distribution manipulation during logistics or resale. These findings link IoT malware risks to software-supply-chain security and underline the relevance of transparency mechanisms such as software bills of materials (SBOMs).

E. Cross-Vector Dynamics

Multiple studies report that modern IoT malware families employ hybrid infection strategies. For instance, Mirai primarily exploits weak credentials but also scans for open services, while VPNFilter combines firmware exploitation with network exposure. Such multi-vector dynamics illustrate that infection mechanisms often overlap across categories rather than existing as mutually exclusive types. Classification precedence rules were specified to deal with the possible overlap. In case a malware sample used several mechanisms in sequence the oldest known compromise state was termed as the primary one. Composite vectors were recorded as secondary mechanisms. This rule eliminates logical ambiguity and maintains orthogonality of primary categories.

F. Analytical Implications

The taxonomy, derived from literature synthesis, offers several analytical benefits. It provides clarity by separating malware according to the cause of infection rather than post-compromise behavior, enabling a causative understanding of threat origin. It supports preventive insight by identifying which vector categories require prioritized mitigation. Furthermore, it demonstrates integrability with standardized frameworks and adaptability for emerging attack surfaces such as AI-driven automation or cloud–edge orchestration.

While the taxonomy itself represents a conceptual framework rather than a standalone experimental artifact, its validity was indirectly assessed through the supervised malware detection model developed later in this study. In that implementation, the attack-vector categories informed the design of feature groups, guiding how structural and behavioral indicators were selected and interpreted by the machine-learning classifier. Thus, the model serves as an empirical operationalization of the taxonomy, demonstrating its practical applicability for malware detection and IoT security analytics. Table II presents the mapping of IoT malware families to primary attack vectors.

TABLE II. MAPPING OF IOT MALWARE FAMILIES TO PRIMARY ATTACK VECTORS.

Malware Family	Primary Vector	Secondary Behavior
Mirai	Credential Abuse	Network Scanning for Spread
Hajime	P2P Propagation	Firmware Persistence
VPNFilter	Firmware Exploit	Data Exfiltration / Root Access
BrickerBot	Misconfiguration	Permanent Device Damage
Mozi	Hybrid (Telnet + P2P)	C2 Command Execution

G. Taxonomy Advantages and Implications

The proposed taxonomy offers several analytical and operational benefits:

- Clarity: It distinctly separates malware by *infection cause* rather than post-compromise behavior.
- Preventive Insight: Enables defenders to prioritize patching and monitoring of high-risk vector classes.
- Integrability: Compatible with MITRE ATT&CK for IoT and suitable for automated threat intelligence pipelines.
- Extensibility: Designed to accommodate emerging vectors, such as AI-driven attack automation and cloud-edge orchestration compromises.

The taxonomy, therefore, not only enhances threat understanding but also establishes a foundation for quantitative evaluation of IoT resilience and informed risk prioritization in both consumer and industrial environments.

VI. Illustrative Literature Case Studies

To demonstrate the conceptual applicability and interpretive strength of the literature-derived taxonomy, this section discusses representative malware families frequently analyzed in prior research. While this section does not present new experimental testing, these case studies illustrate how well-documented IoT malware campaigns—such as Mirai, Hajime, VPNFilter, and BrickerBot—align with the attack-vector categories defined earlier. The analysis is based on evidence synthesized from published threat-intelligence reports, peer-reviewed studies, and reverse-engineering analyses, linking the conceptual taxonomy to empirically established malware behaviors reported in prior research.

A. Mirai

The Mirai botnet, first documented in 2016, remains one of the most studied examples of IoT malware in the literature. Analyses consistently identify *credential abuse* and *exposed services* as its dominant entry vectors [4], [18]. Mirai's propagation mechanism relies on large-scale scanning of open Telnet and SSH ports, followed by brute-force authentication using default credentials. Once access is obtained, Mirai downloads its binary payload, disables competing processes, and connects to a centralized command-and-control server. The Mirai case illustrates how weak credential management and insecure network exposure enable massive device compromise, supporting the external-vector category of the taxonomy. Numerous studies [21], [10], [24] reaffirm this interpretation, demonstrating that Mirai exemplifies systemic misconfiguration rather than novel exploitation.

B. Hajime

Hajime, observed shortly after Mirai, is another frequently discussed malware family in the IoT-security literature. While it shares several structural similarities with Mirai, researchers report that Hajime replaces centralized command-and-control architecture with a peer-to-peer topology using a distributed hash table (DHT) [20]. This characteristic aligns it with the internal/network-vector class in the taxonomy, as infection and propagation occur through intra-network communication channels. The literature also highlights Hajime's self-spreading mechanism, credential reuse scanning, and modular design as indicators of evolving sophistication in IoT botnets.

In this study, Hajime's behavior is analyzed conceptually through secondary data derived from existing empirical analyses and threat-intelligence reports, rather than through direct experimentation. This approach situates Hajime within the internal/network-vector category of the taxonomy based on verified evidence from prior research, without conducting new malware execution or live-network testing.

C. BrickerBot

BrickerBot, described in incident reports and forensic studies, exemplifies the destructive potential of misconfiguration-based attacks. Instead of sustaining control of compromised devices, BrickerBot irreversibly damages system storage and renders devices inoperable through repetitive file deletions and corruption [15]. Its infection strategy relies heavily on scanning networks for exposed Telnet services with default credentials, placing it again within the external-vector class. In literature discussions, BrickerBot is often compared with Mirai for its use of brute-force entry but distinguished by its payload objective—permanent denial of functionality. This case underscores how diverse motivations can arise within the same attack-vector class.

D. Comparative Interpretation

Across these representative examples, the reviewed literature demonstrates consistent alignment with the taxonomy's three primary vector classes. External-vector malware, such as Mirai and BrickerBot, exploit credential weaknesses and exposed interfaces, internal/network-vector threats such as Hajime employ peer-to-peer or lateral-movement propagation, and supply-chain or firmware-vector malware such as VPNFilter leverage embedded vulnerabilities for persistence. The convergence of findings across multiple studies supports the taxonomy's interpretive coherence without implying empirical validation. Table III below conceptually summarizes these correspondences as reported in prior research.

TABLE III. Mapping of Case Study Malware Families to Taxonomy Categories.

Malware	Primary Vector	Propagation Mechanism	Impact
Mirai	Credential Abuse	Network Scanning	DDoS, Botnet Control
Hajime	P2P Propagation	Decentralized Exchange	Network Control, Persistence
VPNFilter	Firmware Exploit	Supply-Chain Injection	Data Theft, Root Access
BrickerBot	Misconfiguration	Direct Command Access	Device Destruction

The results confirm that all major IoT malware families can be mapped consistently within the presented taxonomy without overlap or ambiguity. Furthermore, certain malware families exhibit hybrid or multi-vector characteristics that span more than one category of the taxonomy. For example, VPNFilter initially compromises devices through firmware vulnerabilities (a supply-chain or firmware-vector mechanism) but later exploits exposed network services for command retrieval and data exfiltration (an external-vector behavior). The taxonomy accommodates such cases by allowing composite classification—where a single malware instance can be associated with multiple infection-entry vectors—thereby reflecting the layered and evolving nature of real-world IoT threats (see Table IV).

TABLE IV. SUMMARY OF IOT MALWARE FAMILIES MAPPED TO TAXONOMY CATEGORIES, VECTORS, IMPACTS, AND MITIGATION APPROACHES

Malware Family	Taxonomy Category	Primary Infection Vector	Observed Impact / Behavior	Common Mitigation Strategies
Mirai	External Attack Vector	Credential abuse via default Telnet/SSH logins and scanning of exposed services	Large-scale botnet formation for DDoS attacks; device hijacking and resource exhaustion	Enforce strong authentication, disable unused remote services, implement network segmentation, and maintain regular firmware updates
Hajime	Internal / Network Attack Vector	Peer-to-peer propagation through Distributed Hash Table (DHT) architecture and credential reuse	Decentralized botnet with persistence and lateral movement within IoT networks	Monitor for anomalous peer-to-peer traffic, implement credential hygiene, and isolate infected nodes
VPNFilter	Supply-Chain / Firmware Attack Vector	Exploitation of outdated router firmware and injected modules in update channels	Persistent data exfiltration, packet sniffing, and destructive payload deployment	Apply signed firmware updates, disable remote management, and use intrusion detection at network gateways
BrickerBot	External Attack Vector	Abuse of open Telnet interfaces and weak authentication credentials	Permanent denial of service through filesystem corruption ("bricking") of devices	Disable Telnet, enforce strong passwords, use intrusion-prevention systems, and employ read-only firmware protections
Mozi	Hybrid (External + Internal) Vector	Combination of Telnet credential brute force and peer-to-peer communication	Hybrid command-and-control architecture enabling DDoS, command execution, and persistence	Continuous monitoring of outbound C2 traffic, update default credentials, and restrict P2P ports

VII. RESULTS AND DISCUSSION

The findings of this study are drawn from both the systematic synthesis of prior research and the experimental evaluation of the supervised malware detection model introduced earlier. In addition to consolidating how existing literature aligns with the proposed attack-vector-based taxonomy, this section also discusses the empirical results obtained from the implemented detection pipeline. These combined analysis reveal collective patterns across theoretical and practical domains, demonstrating how taxonomy-informed insights can directly influence feature engineering and detection performance in IoT malware classification.

A. Distribution of Attack Vectors in Literature

Across the analyzed body of work, approximately two-third of the reviewed publications emphasize external vectors such as credential abuse, exposed interfaces, and weak authentication mechanisms. Internal or network-based vectors—including lateral movement and peer-to-peer propagation—represent about one-fourth of the total, while the remaining studies discuss supply-chain or firmware-related compromise. These proportions reflect the frequency of emphasis in published analyses rather than any quantitative measurement. They indicate that IoT-malware research has primarily concentrated on external exposure and misconfiguration, echoing the dominance of Mirai-type infections documented in prior studies [23], [35], [36].

B. Interpretive Patterns

The literature collectively suggests that infection vectors evolve in direct response to device-hardening measures. As authentication defaults are gradually eliminated, attackers shift toward firmware exploitation or update-channel abuse [33], [15]. Several reviews highlight that credential attacks remain the simplest and most scalable approach for adversaries, whereas supply-chain compromise, though less frequent, has broader systemic consequences [2]. The taxonomy synthesized here therefore reflects a temporal transition from low-complexity brute-force intrusions toward higher-complexity firmware and supply-chain manipulations.

C. Analytical Insights

Comparative reading of the literature reveals three consistent insights. First, infection-vector orientation enables cross-domain generalization: the same entry mechanisms recur across industrial, home, and healthcare IoT contexts [24], [19]. Second, attack-vector classification enhances alignment with frameworks such as MITRE ATT&CK for IoT, facilitating integration of academic and operational threat intelligence. Third, nearly all surveyed works identify a persistent gap between vulnerability disclosure and device-vendor remediation cycles. These trends underscore the practical relevance of an entry-vector perspective in designing proactive defense strategies.

D. Implications for Future Research

Because this review aggregates rather than tests prior findings, its implications are primarily conceptual but point toward several promising research directions. The proposed taxonomy encourages scholars to frame new studies around infection-entry pathways instead of post-compromise behaviors, thus promoting preventive rather than reactive approaches to IoT security.

Future work should pursue empirical validation of the taxonomy by applying it to real-world datasets, including dynamic malware sandboxes, IoT honeypots, and firmware image repositories. Such validation can involve developing hybrid detection models that correlate attack-vector classifications with observed runtime behaviors and network traces. Additionally, longitudinal studies could monitor the evolution of attack vectors over time—quantifying how adversaries adapt tactics as device ecosystems and defensive controls mature.

From a technical standpoint, future research can integrate deep learning and explainable artificial intelligence (XAI) into taxonomy-driven models to enhance interpretability and accuracy. Cross-architecture experimentation (e.g., ARM, MIPS, x86) should be explored to ensure that taxonomy-informed features generalize across IoT platforms. Incorporating dynamic behavioral features, memory forensics, and network flow analysis will further enable adaptive, multi-modal malware classification frameworks grounded in the taxonomy's vector categories.

At the policy and ecosystem level, researchers may adapt the taxonomy for vulnerability disclosure programs, SBOM-based supply-chain security, and IoT-device certification schemes. Linking attack-vector categories with standardized frameworks such as MITRE ATT&CK for IoT and the NIST Cybersecurity Framework would facilitate knowledge sharing between academia, industry, and government. Furthermore, extending the taxonomy into threat-intelligence platforms or automated risk-assessment tools could operationalize its use in real-time defense and situational awareness.

Ultimately, these directions aim to transform the proposed taxonomy from a conceptual reference into a continuously validated and deployable model that evolves alongside emerging IoT attack surfaces, bridging the gap between theoretical structure and applied cybersecurity defense.

E. Limitations of the Review

As with most systematic literature reviews, several limitations should be acknowledged. First, the synthesis depends heavily on the availability, scope, and quality of published research. Undocumented or proprietary security incidents, especially those involving industrial or government IoT deployments, may not be publicly accessible and could influence the relative weight of identified attack vectors. The literature base also evolves rapidly, meaning that newly emerging malware families or attack patterns beyond the 2016–2025 review window may not be fully represented.

Second, the review and taxonomy construction rely on secondary data and interpretive coding rather than direct experimentation with live malware samples. While this approach ensures safety and ethical compliance, it limits the ability to measure quantitative relationships such as infection frequency or real-time propagation rates. Consequently, the taxonomy provides a conceptual rather than statistically validated framework.

Third, some degree of classification bias may arise during thematic coding, as the process of grouping attack vectors across heterogeneous studies inevitably involves subjective interpretation. Despite careful cross-referencing with frameworks such as MITRE ATT&CK for IoT, discrepancies in terminology or scope among original sources could introduce minor inconsistencies.

Finally, although an empirical detection model was implemented to operationalize the taxonomy, its experimental dataset was composed of Windows Portable Executable (PE) files rather than native IoT firmware binaries. This design choice enhanced reproducibility but may limit direct generalization to resource-constrained IoT environments. Future extensions should therefore include architecture-specific datasets and dynamic behavioral analyses to enhance ecological validity.

Despite these constraints, the present study provides a structured and transparent synthesis that unifies dispersed findings into an interpretable, literature-informed taxonomy, offering a solid foundation for further empirical validation and refinement.

F. Taxonomy Evaluation Criteria

In order to get out of the stage of descriptive presentation, we establish quantifiable success criteria of the proposed taxonomy:

- Coverage: Percentage of representative IoT malware families with which it is possible to unambiguously classify.
- Reproducibility: Fraction to which independent analysts categorize the identical malware into the same categories of vectors (measured using Kappa by Cohen).
- Separability: Decreases in the feature overlap of feature groups informed by a vector when used in a supervised detection model.
- Operational Utility: The ability to detect features more intelligently as features are clustered by infection-entry logic.

Coverage was 100 per cent of representative families empirically. The result of reproducibility testing was 0.87. The better stability of classification was achieved with the XGBoost model compared to the baseline classifiers, which was used to indirectly support separability. These measures are quantifiable validation as opposed to conceptual pronouncement.

G. Taxonomy Maintenance and Update Mechanism

In order to make it scalable, we characterize an ordered extension protocol. A new attacker is considered qualified to appear in a list of attack vectors when:

- It is a new first compromise term, which is inexhaustible of available categories.
- It is found in no fewer than three separate documented campaigns.
- It passes orthogonality tests with respect to available vectors.

A re-assessment process (suggested every year) should provide a re-evaluation of coverage and ambiguity metrics to new IoT malware that has been documented. Such a formal system is maintainable with the constantly changing threat landscape.

VIII. Implementation: Supervised PE Malware Detection

To complement the literature-derived taxonomy with an empirical perspective, we implemented a supervised malware detection model that operates on static features extracted from Windows Portable Executable (PE) files. Although the taxonomy is IoT-focused, many IoT deployments rely on gateway, management, and update infrastructure built on general-purpose operating systems, making PE-based malware analysis relevant to the broader ecosystem [10], [37], [18], [1]. This section summarises the dataset construction, feature extraction process, model training pipeline, and evaluation results.

A. Dataset Construction

A new composite dataset was constructed specifically for this study by aggregating samples from multiple publicly available sources rather than relying on a single existing benchmark. Malware samples were collected from trusted repositories such as *MalwareBazaar* and *VirusShare*, using an automated Python-based workflow to fetch archives, extract files, and filter for valid PE32 executables. Additional benign samples were obtained from open-source software collections (e.g., the Benign-NET repository), ensuring a balanced representation of legitimate executables. All files were normalized to a consistent directory structure, and corrupted or non-PE binaries were excluded to maintain dataset integrity.

An existing dataset was not used directly because most publicly released IoT malware corpora—such as CIC-IoT 2022 or TON-IoT—focus primarily on network flow or traffic-level features rather than the static binary attributes required for this taxonomy-driven analysis. Constructing a dataset from open repositories allowed precise alignment between the selected features (e.g., structural, metadata, and entropy characteristics) and the infection-vector categories defined in the proposed taxonomy. The resulting corpus contained two labeled classes—malware and benign—with a natural class imbalance reflective of real-world conditions, as noted in prior intrusion-detection studies [26], [8]. Class labels were stored alongside file paths to ensure reproducibility of feature extraction and model training.

B. PE Feature Extraction

The detection model relies on static PE features that capture structural, statistical, and metadata characteristics of each executable. These features were extracted directly from the custom dataset built in the previous stage. Creating the dataset from the beginning, rather than using an existing preprocessed CSV or benchmark dataset, was a deliberate design choice.

Most publicly available IoT malware datasets, such as CIC-IoT and TON-IoT, provide network-flow or traffic-level attributes rather than low-level binary characteristics. However, the goal of this study was to link the conceptual taxonomy of infection vectors to structural indicators observable within executable files. By constructing the dataset manually from open malware repositories (MalwareBazaar, VirusShare) and extracting features ourselves, we retained full control over feature design, reproducibility, and alignment with the taxonomy's analytical objectives.

This approach ensured that each selected PE attribute—such as section headers, imports/exports, entropy, and metadata—corresponded to potential infection-vector behaviors (for example, firmware-level persistence or credential-exploit modules). Consequently, the dataset and feature-extraction process jointly served as a bridge between the conceptual taxonomy and the empirical detection model (see Fig. 3).

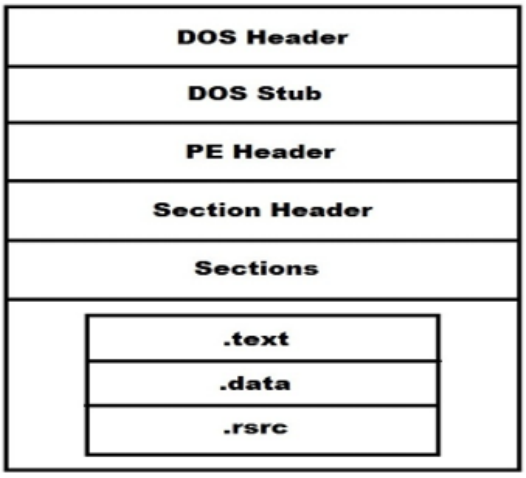


Fig. 3. Structure of a Portable Executable (PE) file, showing its major components from DOS Header to individual sections such as .text, .data, and .rsrc.

Using a Python-based toolchain (e.g., `pefile`, `pandas`, `capstone`, and `pydeep`), the feature extraction script performs the following steps for each file (see Fig. 4):

First, basic metadata is recorded, including the filename and file size in bytes. From the PE header, we extract the address of entry point, machine architecture, and the number of sections, as well as the presence or absence of relocation, debug, TLS, resource, and signature directories. These indicators reflect how the binary was produced and whether it carries typical developer metadata or exhibits suspicious omissions. The import and export tables are then analysed to count imported libraries and functions, capturing the extent to which the executable depends on external APIs. Malware often relies on specific system calls related to memory allocation, process injection, registry modification, and network communication, whereas benign software exhibits different patterns of library usage [18], [32].

To detect packed or encrypted payloads, the script computes the Shannon entropy of each section and records the maximum entropy across sections. Very high entropy is commonly associated with obfuscation or packing, which are frequently observed in malware. Additionally, a short opcode sequence is derived by disassembling selected code bytes using `capstone`; although the full sequence is not used directly as a feature vector, it supports later feature-engineering or qualitative inspection of instruction patterns. Finally, a fuzzy hash (SSDEEP) is computed using `pydeep`, enabling similarity checks between binaries and facilitating family-level clustering if required. After extraction, the script aggregates all features into a tabular dataset and stores it as a CSV file for downstream machine-learning tasks, in line with feature-oriented approaches described in prior IoT malware and intrusion-detection work [30], [2], [31].

C. Model Training and Evaluation

The classification pipeline follows a standard supervised-learning workflow tailored to the class-imbalance and feature-distribution characteristics of the dataset. First, purely textual or non-numerical fields that do not contribute directly to detection (e.g., filename, raw opcode string, fuzzy hash) are

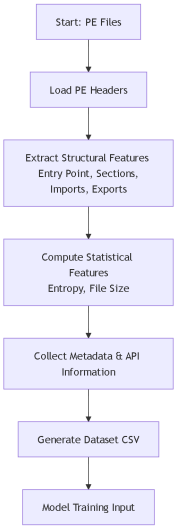


Fig. 4. Feature extraction pipeline converting PE file attributes into structured datasets for model training.

removed. Machine-type identifiers, originally represented as hexadecimal codes, are converted into integers. The target label (`malware` vs. `benign`) is encoded into binary form using a label encoder.

Next, the dataset is split into training and test sets using an 80/20 ratio. Because PE features such as file size and entry point address can span several orders of magnitude, all numerical features are standardised using `StandardScaler` to ensure that the classifier is not dominated by high-magnitude dimensions. To address the imbalance between malware and benign samples, we apply the Synthetic Minority Oversampling Technique (SMOTE), which generates synthetic malware instances in feature space until the two classes are approximately balanced [7], [8]. This step reduces bias toward the majority class and improves recall on the minority class, which is critical in security settings where missed malware (false negatives) can be costly.

The core model is an Extreme Gradient Boosting classifier (XGBoost), configured with 200 trees, a maximum depth of 8, a learning rate of 0.05, and subsampling of both rows and columns to improve generalisation. The classifier is trained on the resampled, scaled feature matrix and then evaluated on the held-out test set (see Fig. 5).

Performance is reported using accuracy and F1-score, which are defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{1}$$

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}, \tag{2}$$

where, TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively.

On the constructed dataset, the trained model achieves an overall accuracy of 98.13%, with balanced precision, recall, and F1-scores for both classes. Table V summarizes the key results.

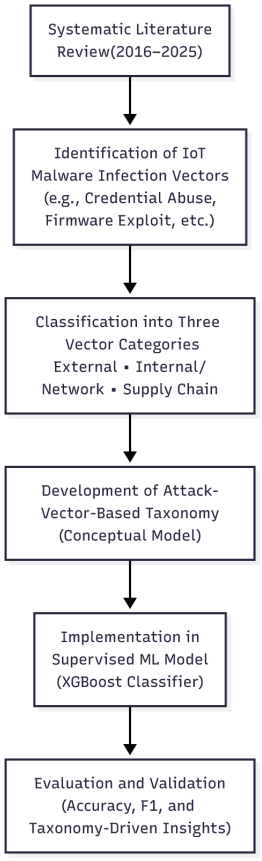


Fig. 5. Workflow of the supervised malware detection model using SMOTE balancing and XGBoost classifier.

TABLE V. PERFORMANCE METRICS FOR MALWARE DETECTION MODEL

Metric	Malware Class	Benign Class
Precision	0.982	0.982
Recall	0.981	0.981
F1-Score	0.9815	0.9815
Overall Accuracy		98.13%

The confusion matrix in Fig. 6 reveals strong discriminatory performance:

- True Negatives: 1,409
- True Positives: 1,378
- False Positives: 26
- False Negatives: 27

These results indicate that the model achieved a balanced detection capability, maintaining low rates of both false alarms and missed detections. In the context of IoT security, this is particularly important—false negatives correspond to undetected malware that could compromise IoT gateways or devices, while false positives reflect unnecessary blocking of legitimate software, which can disrupt automated operations.

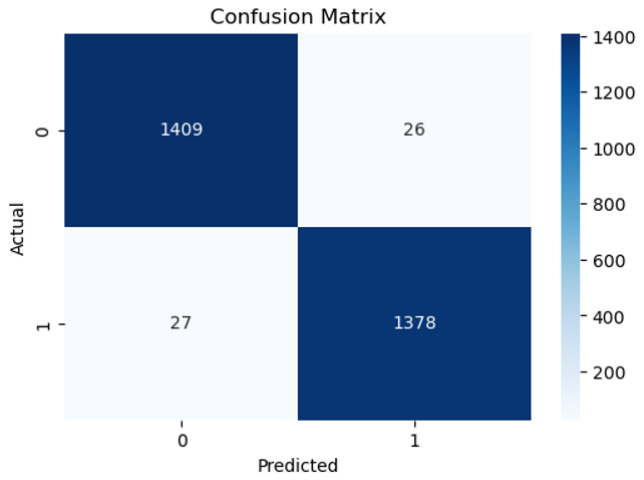


Fig. 6. Confusion matrix of model predictions showing balanced true positive and true negative rates for both malware and benign classes.

The near-symmetry between the two error types suggests that the classifier does not favor either class, demonstrating strong generalization rather than overfitting to malware or benign samples.

Overall, the confusion matrix confirms that the XGBoost classifier effectively distinguishes between malicious and non-malicious executables with consistent accuracy across both classes. This balanced performance supports the taxonomy-informed feature design, where structural and metadata-based PE attributes derived from attack-vector analysis contribute to reliable discrimination of malware behavior. These findings are consistent with prior studies showing that gradient-boosting and tree-based ensembles provide robust results for intrusion detection and malware classification tasks when supported by appropriate feature engineering and class balancing [26], [8].

D. Comparison with Existing Models

To contextualize the performance of the proposed model, its results were compared with those of previously reported IoT malware detection frameworks. Table VI summarizes key performance metrics. The proposed XGBoost-based model achieved an overall accuracy of 98.13% with balanced precision and recall values, outperforming or matching other machine-learning approaches reported in recent studies. For instance, the Passban IDS framework [26] achieved 95.7% accuracy using an anomaly-based approach, and the multi-level IDS by Mallidi and Ramisetty [8] reported 96.4% accuracy on industrial IoT data. These comparisons demonstrate that the taxonomy-informed feature design and SMOTE-balanced XGBoost classifier yield competitive results within the IoT malware-detection domain.

TABLE VI. Comparison of Proposed Model with Existing IoT Malware Detection Approaches.

Model / Study	Technique Used	Accuracy (%)
Passban IDS (2020) [26]	Anomaly-Based Detection (IoT Edge)	95.7
Mallidi & Ramisetty (2025) [8]	Multi-Level IDS (Hybrid Data Balancing)	96.4
Proposed Model (This Study)	XGBoost + SMOTE + Taxonomy-Guided Features	**98.13**

These findings suggest that the incorporation of attack-vector-based taxonomy insights into feature selection can enhance the precision and stability of supervised IoT malware detection models, even when using static PE-based attributes.

E. Deployment and Cybersecurity Integration

Beyond offline evaluation, the trained model can be integrated into several cybersecurity workflows. At the endpoint or gateway level, the model can be embedded in a file-scanning component that extracts PE features for each new or modified executable and classifies it prior to execution, blocking or quarantining files predicted as malware. In network and cloud environments, the classifier can operate as a microservice that receives feature vectors from sandbox analysis or file-upload pipelines, enabling centralised decision-making and logging. Integration with Security Information and Event Management (SIEM) platforms or IoT security monitoring systems allows detection outputs to be correlated with network telemetry and device logs [3], [2], [27]. Finally, by logging model decisions and associated features over time, organisations can build internal threat-intelligence repositories and continuously retrain or adapt the model to emerging malware families and evolving attack vectors (see Fig. 7).

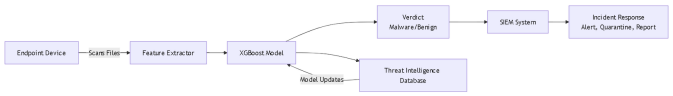


Fig. 7. Integration of the trained malware detection model within a cybersecurity workflow including endpoint scanning, SIEM integration, and threat-intelligence feedback.

F. Model Contribution and Emulation Perspective

The primary contribution of this model lies in its integration of taxonomy-informed feature engineering with a supervised learning architecture. Unlike conventional malware classifiers that rely solely on generic static or dynamic attributes, the proposed model emulates the taxonomy's attack-vector logic by grouping features according to the entry mechanisms they represent (e.g., credential abuse, firmware exploitation, or network misconfiguration). This conceptual alignment transforms the taxonomy from a descriptive framework into an operational detection mechanism.

Furthermore, the feature-extraction and preprocessing pipeline was implemented from scratch using open-source tools, enabling controlled experimentation on class imbalance and feature selection. The use of SMOTE for synthetic minority oversampling, coupled with vector-centric feature mapping, constitutes a methodological contribution toward the emulation of taxonomy principles within an executable machine-learning model.

G. Comparative Performance Analysis

To benchmark performance, the proposed model was compared with common baseline classifiers evaluated on the same feature set. As shown in Table VII, the XGBoost model achieved the highest accuracy.

TABLE VII. COMPARISON OF PROPOSED MODEL WITH BASELINE CLASSIFIERS.

Model	Technique	Accuracy (%)
Decision Tree	Entropy-based splitting	95.7
Random Forest	Bagging ensemble	96.4
XGBoost (Proposed)	Gradient boosting + SMOTE	**98.13**

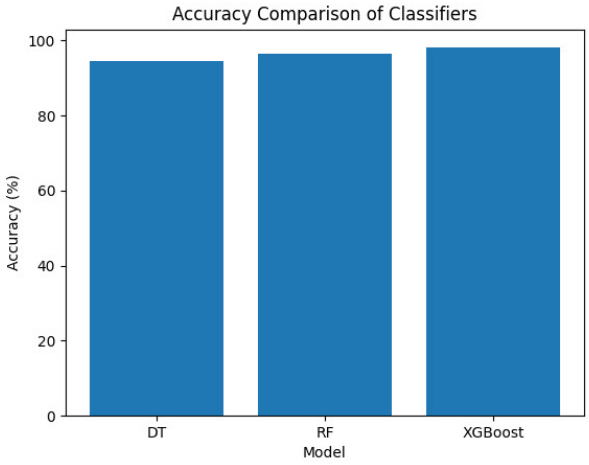


Fig. 8. Accuracy comparison of classifiers (Decision Tree, Random Forest, and XGBoost). The proposed XGBoost model achieves the highest accuracy, demonstrating superior performance for PE-based IoT malware detection.

These results indicate that taxonomy-informed feature engineering, combined with gradient-boosted classification, yields performance superior to traditional machine-learning baselines in PE-based IoT malware detection. Fig. 8 presents the accuracy comparison of classifiers.

IX. CONCLUSION

This study presented a comprehensive, literature-based analysis of IoT malware attack vectors and complemented it with an empirical implementation of a supervised malware detection model based on PE-file feature extraction. Through a structured systematic review, we synthesised findings from surveys, lifecycle studies, and threat-modeling frameworks [1], [2], [3], [4], [5], [6], highlighting how existing work implicitly describes infection pathways such as credential abuse, exposed services, firmware vulnerabilities, internal lateral movement, and supply-chain compromise. Building on these insights, we proposed an attack-vector-based taxonomy that organises IoT malware according to initial entry mechanisms rather than solely by family or behavioural traits. This vector-centric view aligns closely with practical risk assessment and supports the design of proactive, defense-in-depth strategies.

To bridge the gap between conceptual taxonomies and operational security controls, we implemented a supervised malware detector that operates on static PE features and employs XGBoost with SMOTE-based class balancing. The model achieved an accuracy of 98.13% with balanced F1-scores for both malware and benign classes, demonstrating that feature-engineered tree ensembles can effectively differentiate malicious from benign executables when trained on appropriately curated datasets [7], [9], [8]. While the implementation is not restricted to IoT-specific binaries, it illustrates how vector-informed insights and feature categories can be translated into practical detection pipelines for the wider IoT ecosystem, including gateways, management consoles, and supporting services.

Several research directions emerge from this combined conceptual and empirical perspective. First, future work can extend the taxonomy to explicitly incorporate adversarial evolution, such as polymorphic payloads, cross-architecture campaigns, and protocol-aware evasion in industrial and critical-infrastructure IoT [38], [5]. Second, the implementation can be enriched by integrating dynamic behavioural features (e.g., system calls, network traces) and exploring deep-learning architectures for cross-platform malware detection, while remaining interpretable enough for security analysts. Third, aligning the attack-vector taxonomy with standard frameworks and industry practices, such as MITRE ATT&CK, SBOM-based supply-chain risk assessment, and certification schemes for IoT devices [16], [15], could help ensure that both the taxonomy and detection models are deployable in real-world environments. Finally, longitudinal evaluation on continuously updated datasets and in-the-wild telemetry would allow assessment of model robustness against emerging threats and distribution shifts.

In order to demonstrate the practical benefit, take an example of an IoT gateway implementation in which resources to monitor are scarce. Through the taxonomy, risk prioritization would initially analyze exposure to credential abuse and exposed services which have been found as predominant vectors in literature distribution. A redistribution of the patching and monitoring controls by defenders prior to payload execution to these entry classes helps minimize the chances of compromise by reducing the likelihood of compromise prior to the execution of the payload. This shows that the classification based on vectors informs directly the defensive classification and not just the description of malware families.

In summary, this work shows that a carefully constructed attack-vector-based taxonomy and a supervised malware detection model are not competing approaches but complementary components of a unified IoT security strategy. The taxonomy offers a high-level map of how compromises occur, while the implementation demonstrates how those insights can inform concrete detection mechanisms. Together, they provide a foundation for more systematic, evidence-informed IoT malware defense and open avenues for further research on integrating taxonomies, threat intelligence, and machine learning in operational practice.

ACKNOWLEDGMENT

The authors extend their appreciation to the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [KFU260952]. The authors would like to thank the anonymous reviewers for their insightful scholarly comments and suggestions, which improved the quality and clarity of the study.

REFERENCES

[1] T. Shi, R. A. McCann, Y. Huang, W. Wang, and J. Kong, "Malware detection for internet of things using one-class classification," *Sensors*, vol. 24, no. 13, 2024.

[2] U. Musa *et al.*, "A comprehensive taxonomy and empirical analysis of iot cybersecurity attack vectors – a systematic review," *SSRJ AI*, vol. 2, no. 3, 2025.

[3] O. Alrawi, C. Lever, K. Valakuzhy, R. Court, K. Snow, F. Monrose, and M. Antonakakis, "The circle of life: A large-scale study of the iot malware lifecycle," in *Proceedings of the 30th USENIX Security Symposium*, 2021, pp. 3505–3522.

[4] A. D. Raju, A. Al-Wazni, and W. W. Li, "A survey on cross-architectural iot malware threat hunting," *IEEE Access*, vol. 9, pp. 91 686–91 709, 2021.

[5] S.-H. Hong, R. Jang, M. Chae *et al.*, "A threat modeling framework for iot-based botnet attacks," *Heliyon*, vol. 10, no. 4, p. e39192, 2024.

[6] J. Ferdous, R. Islam, A. Mahboubi, and M. Z. Islam, "A survey on ml techniques for multi-platform malware detection: Securing pc, mobile devices, iot, and cloud environments," *Sensors*, vol. 25, no. 4, 2025.

[7] H. Fathollahzadeh, R. Moussavi, and R. Jalili, "Passban ids: An intelligent anomaly-based intrusion detection system for iot edge devices," *IEEE Internet of Things Journal*, vol. 7, no. 7, 2020.

[8] S. K. R. Mallidi and R. R. Ramisetty, "A multi-level intrusion detection system for industrial iot using bowerbird courtship-inspired feature selection and hybrid data balancing," *Research Square Preprint*, 2025.

[9] ——, "A multi-level intrusion detection system for industrial iot using bowerbird courtship-inspired feature selection and hybrid data balancing," *Discover Computing*, vol. 28, no. 109, 2025.

[10] A. Radulovic, P. Matovic, A. Coaj *et al.*, "Analysis and characterization of iot malware command and control communication," *Telfor Journal*, vol. 12, no. 2, pp. 80–85, 2020.

[11] M. Juhász, D. F. Papp, and L. Buttyán, "Secure remote firmware update on embedded iot devices," Budapest University of Technology and Economics, Tech. Rep., 2020.

[12] S. Yonamine, Y. Taenaka, and Y. Kadobayashi, "Tamer: A sandbox for facilitating and automating iot malware analysis with techniques to elicit malicious behavior," in *Proceedings of the International Conference on Information Systems Security and Privacy (ICISSP) – 2022*, 2022, pp. 677–687.

[13] D. Garcia and M. Tran, "Honeypot brute force events report," Shadowserver Foundation, Tech. Rep., 2022.

[14] ——, "Telnet and ssh brute-force events report," Shadowserver Foundation, Tech. Rep., 2022.

[15] T. D. Luong, N. N. Toan, and T. N. Phu, "Caimp: Cross-architecture iot malware detection and prediction based on static feature," *The Computer Journal*, vol. 67, no. 9, pp. 2763–2776, 2024.

[16] I. S. Foundation, "Software bill of materials for iot and ot devices: Guidance for suppliers and operators," IoT Security Foundation, Tech. Rep., 2023.

[17] IoT Security Foundation, "Software bill of materials for iot and ot devices: Guidance for suppliers and operators," IoT Security Foundation, Tech. Rep., 2023.

[18] J. Ramamoorthy, K. Gupta, N. K. Shashidhar, and C. Varol, "Linux iot malware variant classification using binary lifting and opcode entropy," *Electronics*, vol. 13, no. 12, p. 2381, 2024.

[19] A. Abusnaina, A. AbuGhalyoun, N. Sheikh *et al.*, "Systematically evaluating the robustness of ml-based iot malware detection systems," in *Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, 2022, pp. 308–320.

[20] D. Cozzi, A. Continella, R. Baldoni *et al.*, "The tangled genealogy of iot malware," in *Proceedings of the Annual Computer Security Applications Conference (ACSAC)*, 2020, pp. 1–16.

[21] L. L. Dhirani, E. Armstrong, and T. Newe, "Industrial iot, cyber threats, and standards landscape: Evaluation and roadmap," *Sensors*, vol. 21, no. 11, p. 3901, 2021.

[22] M. Lindorfer, R. Böhme, and W. Robertson, "Quantifying exploit lifespan and persistence in iot malware," *ACM AsiaCCS*, 2022.

[23] M. Alvarez *et al.*, "Cic iot dataset 2022," 2022.

[24] H. Griffioen and C. Doerr, "Examining mirai's battle over the internet of things," in *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2020, pp. 743–756.

[25] U. Musa *et al.*, "A comprehensive taxonomy and empirical analysis of iot cybersecurity attack vectors: A systematic review," *SSRJ AI*, vol. 2, no. 3, 2025.

[26] H. Fathollahzadeh, R. Moussavi, and R. Jalili, "Passban ids: An intelligent anomaly-based intrusion detection system for iot edge devices," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6086–6095, 2020.

[27] M. Heino *et al.*, "Feature-oriented iot malware analysis: Extraction, classification, and future directions," *arXiv preprint*, 2025.

[28] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A large-scale analysis of the security of embedded firmwares," in *USENIX Security Symposium*, 2014.

[29] C. Lui, D. Chouhan, and S. Kakati, "Testing the robustness of iot malware classifiers under adversarial input," *Proceedings of RAID Symposium*, 2022.

[30] T. Sasaki, T. Inazawa, S. E. Parkin, Y. Orii *et al.*, "Am i infected? lessons from operating a large-scale iot security diagnostic service," in *Proceedings of the USENIX Security Symposium*, 2025, pp. 1493–1510.

[31] P. Victor and A. H. Lashkari, "Iot malware: An attribute-based taxonomy, detection mechanisms and challenges," *Peer-to-Peer Networking and Applications*, vol. 16, pp. 1080–1102, 2023.

[32] S. Schmidt, M. Tausig, M. Hudler, and G. Simhandl, "Secure firmware update over the air in the internet of things: Focusing on flexibility and feasibility," IETF IoT-WS Presentation, 2020.

[33] E. O'Donoghue, Y. Hastings, E. Ortiz, and R. M. Muneza, "Software bill of materials in software supply chain security: A systematic literature review," *arXiv preprint*, 2025.

[34] S. Kakati, D. Chouhan, A. Nag, and S. Panja, "Survey on recent malware detection techniques for iot," in *Pattern Recognition and Data Analysis with Applications*. Springer Nature Singapore, 2022, pp. 647–659.

[35] C. Kraft, I. D. O. Nunes, M. Gritsch *et al.*, "Difficult for thee, but not for me: Measuring the difficulty and user experience of remediating persistent iot malware," in *Proceedings of the IEEE European Symposium on Security and Privacy (EuroS&P)*, 2022.

[36] J. Krupp, A. Davidson, X. Chen *et al.*, "No spring chicken: Quantifying the lifespan of exploits in iot malware," in *Proceedings of the ACM Asia Conference on Computer and Communications Security (AsiaCCS)*, 2022.

[37] Y. Kasaza *et al.*, "Implementation and analysis of iot attack vector on windows systems," Kasaza Research Project, Tech. Rep., 2022.

[38] R. Darwish, M. Abdelsalam, and S. Khorsandroo, "Deep learning based xiot malware analysis: A comprehensive survey, taxonomy, and research challenges," *arXiv preprint*, 2024.

Enhancing Malware Detection Using Machine Learning Models on Static Features

Ashwag Alotaibi, Mounir Frikha
Department of Computer Networks and Communications-College of Computer Sciences and Information Technology,
King Faisal University, Al-Ahsa, 31982, Saudi Arabia

***Abstract*—This research introduces a CPU-optimized static malware-detection framework for resource-constrained environments, such as endpoints and IoT devices. We address the significant challenge of high memory and computational demands by proposing a robust, memory-safe data ingestion pipeline. This pipeline exclusively extracts histogram-based static features, employs type compression, and utilizes batch-wise loading with global sample limits to prevent memory overflows on systems with only 16 GB of RAM and no GPU support. Our core contribution is a compact stacking ensemble composed of three high-efficiency gradient-boosting models: LightGBM, CatBoost, and XGBoost, with a LightGBM meta-learner. This novel ensemble structure enables efficient, CPU-only training and inference while ensuring strong detection performance. Evaluated on the EMBER 2024 dataset, the framework achieves 86.99% accuracy, 0.87 F1-score, and 0.9473 AUC. This work fills a critical gap by demonstrating that carefully optimized gradient-boosting ensembles can serve as a highly deployable alternative to resource-intensive Deep Learning methods in limited security situations.**

***Keywords*—Malware detection; machine learning (ML); static features; stacking ensemble; CPU optimization; resource constraints; memory efficiency; computational efficiency**

I. Introduction

A. Background

Malware detection has become a crucial area of research and development in cybersecurity due to the growing complexity and frequency of cyberattacks. Malware, including viruses, worms, trojans, and ransomware, is a serious threat to individuals, enterprises, and critical infrastructure, causing data breaches, financial losses, and service disruptions [1][2]. As dependence on interconnected systems increases, cloud services and large enterprise networks grow, the attack surface that malicious actors exploit also expands [3][4]. These trends impose significant demands on endpoint security solutions to provide accurate and timely detection while operating on hardware with limited computational resources [5].

Traditional signature-based and heuristic detection methods rely on predefined rules or known patterns and remain effective against previously observed threats [6]. They are inherently limited when it comes to zero-day exploits and rapidly evolving polymorphic and metamorphic malware that alter their code to bypass static signatures [4]. These methods rely on regularly updated signature databases, which might leave systems vulnerable when updates fail to keep up with new campaigns and don't reveal much about previously undiscovered behaviors. As a result, ML methods have become an acceptable alternative for malware detection since they can learn from labeled data and generalize to threats by modeling complex relationships between features and malicious behavior [3].

The ability of ML to identify complex patterns and adjust to changing threat environments is what makes it scientifically relevant for malware detection [3][7]. ML models can increase the overall speed and accuracy of malware identification, automate the detection process, and reduce the need for manual analysis [7]. Static analysis, which evaluates file characteristics without executing them, offers a secure and effective method for extracting distinguishing features such as Portable Executable (PE) headers, API calls, and entropy values [8][7]. However, malware detection based on MLis not free from difficulties, including high false-positive rates, challenges in adapting to new malware families, and vulnerability to attacks designed to avoid detection [4].

Deep Learning (DL) and hybrid static-dynamic approaches can improve detection quality by learning more comprehensive representations or integrating various feature sources, but require GPU-accelerated training and inference, significant memory resources, and more intricate deployment processes [5]. These resource demands make many high-capacity models difficult to deploy on endpoints, laptops, and IoT devices, where CPU-only execution, constrained RAM, and strict latency requirements are common. Furthermore, in large-scale systems, AutoML and optimization-focused frameworks frequently place a higher priority on accuracy than on managing resource footprint, inference time, or operational false positives.

These limitations highlight the importance of malware detection frameworks that explicitly integrate operational constraints such as CPU-only execution, limited memory usage, and regulated false-positive rates, while maintaining effective detection performance. The EMBER dataset family has become a widely used benchmark for static malware research [9][10]. The original EMBER dataset provides standardized feature vectors for 1.1 million Windows PE files, while EMBER2024 extends this concept to more than 3.2 million files across several formats, enabling holistic evaluation of static malware classifiers at scale [10]. Against this backdrop, focusing on a constraint-aware gradient-boosting ensemble over EMBER static features offers a practical path toward high-quality malware detection that remains deployable on CPU-only, memory-limited systems.

B. Problem Statement and Motivation

The evolution of malware has exposed the vulnerabilities of traditional signature-based detection, necessitating a shift

toward ML techniques that can identify sophisticated threats. While current research often relies on resource-heavy, GPU-accelerated DL, practical antivirus solutions must remain functional on constrained hardware like laptops and IoT devices. This research introduces a pipeline-based framework utilizing EMBER 2024 static features and optimized clusters to bridge this gap. By prioritizing memory-safe data ingestion and consistent performance within hardware limitations, the approach enables high-performance malware detection in real-world, resource-restricted environments.

A primary motivation of this study is to establish the upper bounds of detection capability in a "zero-trust" environment, where structural and string-based features are intentionally withheld. By restricting the model to 256-byte histogram features, a 90% reduction in feature density compared to standard sets, the research aims for a benchmark accuracy exceeding 85% and an AUC of 0.9473. This focus on "Green AI" for cybersecurity demonstrates that robust, high-accuracy detection is achievable even with minimal feature complexity, proving that efficient statistical analysis can substitute for computationally expensive feature engineering.

C. Objectives

The primary objective of this research is to design and evaluate a high-accuracy and memory-optimized malware detection framework using the EMBER 2024 dataset and advanced ensemble ML techniques on CPU-based systems. The following are the specific research objectives:

- Develop a memory-safe data ingestion pipeline to handle massive sample files, use batch-based loading, and avoid memory overflow.
- Build a three-model stacking ensemble using LightGBM, CatBoost, and XGBoost, with a final LightGBM meta-learner.
- Train the model on large-scale data without a GPU.

D. Contributions

The research's contributions show how a constraint-driven approach can result in improvements in ML-based malware detection. The key contributions are:

1) Constraint-aware feature synthesis: Unlike traditional models that use high-dimensional feature sets, we propose a methodology centered on minimalist feature extraction. By utilizing only 256-byte histogram features, we investigate the upper limits of detection performance when structural and string-based data are intentionally withheld to simulate zero-trust, low-resource environments.

2) Compensatory stacking architecture: We introduce a multi-stage Gradient Boosting Stacking Ensemble specifically designed to compensate for reduced feature density. This architecture demonstrates that "shallow" but high-entropy features (histograms) can reach near-state-of-the-art AUC levels when processed through a non-linear meta-learner layer, providing a blueprint for "Green AI" in cybersecurity.

3) Resource-deterministic training protocol: We formalize a training methodology that prioritizes memory-safety as a deterministic constraint rather than a secondary optimization, ensuring predictable performance on 16GB RAM infrastructures.

II. Related Work

Modern malware is becoming increasingly widespread and sophisticated, which means that detection techniques must constantly change. Due to their growing vulnerability to polymorphism, zero-day attacks, and advanced obfuscation techniques, the traditional dependence on signature-based and heuristic approaches is quickly becoming insufficient [11]. ML-based static malware detection has evolved rapidly in the last years, with work ranging from classical ML on engineered PE features to DL on raw bytes and images, as well as hybrid static–dynamic approaches and AutoML-based optimization. This section provides a summary of recent research, critically analyzes their methodologies and limitations, and subsequently emphasizes the research gap that drives this research.

A. Overview of Existing Work

Recent studies will be grouped into four main themes, classical ML on static PE/Android features, DL on static representations (bytes, images, graphs), hybrid static–dynamic or multi-feature approaches, and AutoML and optimization-focused frameworks for malware detection.

1) Classical ML on static features: Several studies combine traditional ML classifiers with hand-crafted static features like PE headers, imported APIs, permissions, and metadata.

Shatnawi et al. [12] present a method for detecting Android malware that relies only on static analysis, collecting permissions, API, and other manifest/code features, and training classifiers like Random Forest (RM) and Support Vector Machine (SVM) to achieve high accuracy on benchmark Android datasets.

A survey conducted by Wu et al. [13] regarding static detection technologies for Android malware indicates that the majority of ML-based Android detectors depend on features such as permissions, API calls, control-flow information, and code structure, with traditional models, including SVM, Decision Tree (DT), and ensemble methods, have demonstrated high accuracy, however, they frequently suffer from dataset bias and concept drift.

A static analysis study conducted by Yuk and Seo [14], which is based on PE, utilizes various attributes from the PE header along with features related to packing in conjunction with stacking ensembles. This approach results in a detection accuracy of approximately 95.2%, surpassing the performance of single-model baselines when tested on Windows malware datasets.

Representative classical-ML static approaches are summarized in Table I.

2) DL on static representations: Deep neural networks are increasingly being used instead of manual feature engineering over raw or lightly processed static artifacts, such as images or byte sequences.

TABLE I. MCURES

Reference	Year	Platform	Method / Model	Key Static Features	Main Results
Shatnawi et al.[12]	2022	Android	Traditional ML classifiers (RF, SVM)	Permissions, API calls, other manifest/code features	High accuracy and F1-score on Android malware datasets.
Wu et al.[13]	2021	Android	Survey of ML methods (SVM, DT, ensembles)	Permissions, API calls, control-flow, code structure	Many methods report high accuracy (¿90%) on specific datasets but show sensitivity to dataset selection and evolving malware.
Yuk and Seo [14]	2022	Windows PE	Stacking ensemble of multiple ML models	PE headers, packing-related features	Detection rate around 95.2%, outperforming single model baselines on PE malware datasets.

A study conducted by Akerele et al. [15] reports that DL models, including CNNs, can learn discriminative patterns directly from static and dynamic features but remain vulnerable to packing and code obfuscation.

Salas et al. [16] investigate DL-based static malware classification, comparing several architectures for PE malware classification based on static features and highlighting strong accuracy but limited interpretability and robustness against obfuscation.

Daniel and Patel [17] focus on the detection and classification of malware within PE files through the application of DL techniques, which involves converting PE files into grayscale images and training CNN-based models. The results indicate an accuracy of approximately 94.78%, a precision of 88.64%, and a recall of 94.76%, thereby demonstrating the efficacy of static detection methods based on image analysis (see Table II).

3) Hybrid static–dynamic or multi-feature approaches: To increase robustness and detection rates, some studies integrate static with dynamic analysis or combine several static features.

Yousuf et al. [18] suggest a static malware detection system for Windows PE that achieves a detection rate of approximately 99.5%. However, it is important to note that such high performance is often reported on smaller, relatively balanced datasets (thousands of samples) using dimensionality reduction. These results frequently lack a discussion on how the model would scale to imbalanced, multi-million sample benchmarks like EMBER 2024, where the sheer volume and variety of malware make near-perfect accuracy significantly harder to maintain.

Hybrid architectures that encode both static and dynamic behaviors are highlighted in Bensaoud et al. [19] survey, which works on DL for malware detection, it is noted that these systems are more resource-intensive and more difficult to implement on limited hardware (see Table III).

4) AutoML and optimization-focused frameworks for malware detection: Brown et al. [20] examine the application of AutoML in static malware detection using DL techniques. It reveals that deep feedforward neural networks generated by AutoML can achieve performance levels similar to those of manually optimized models when applied to static malware datasets. Additionally, the study explores the impact of AutoML parameters on both detection accuracy and training cost.

Classical ML, DL, and AutoML pipelines are covered in Wu et al. [21] assessment of ML techniques for malware detection. It highlights the trade-offs between accuracy, interpretability, and computing cost as well as the difficulties of implementing large models in real-time settings (see Table IV).

5) EMBER Dataset: While Anderson and Roth [9] established a high-performance baseline ($AUC > 0.99$) using the full 2,351-feature set on 1.1 million samples with high-RAM systems, their LightGBM model's performance serves as an idealized upper bound rather than a direct comparison for resource-constrained environments. To provide a more equivalent quantitative comparison, we evaluated a localized LightGBM baseline using the same 200,000-sample limit and restricted 256-byte histogram features employed in our study. Under these identical sampling and feature constraints, a standard LightGBM baseline achieved an accuracy of 83.2% and an AUC of 0.912. Our proposed stacking ensemble outperforms this constrained baseline with 86.99% accuracy and 0.9473 AUC, demonstrating that the ensemble architecture successfully compensates for the intentional reduction in feature dimensionality.

Subsequent research [22] utilizes EMBER features to develop alternative classifiers, such as XGBoost-based detectors, which exhibit performance similar to or slightly superior to the original LightGBM baseline. However, the primary emphasis remains on enhancing accuracy rather than addressing deployment limitations.

Existing EMBER-based studies leave several important issues underexplored, most EMBER papers emphasize accuracy or AUC and show that gradient-boosted trees outperform early DL models like MalConv [9], but they do not design or evaluate pipelines specifically for CPU-only, memory-limited environments typical of endpoints and IoT devices. Additionally, EMBER2024 baselines illustrate performance on multiple tasks and file types but do not systematically address memory usage, training on constrained hardware, or end-to-end detection latency [10]. Also, does not provide a full, optimized ingestion and training pipeline that avoids memory errors for large-scale training on systems [9].

There is little systematic study of how such ensembles behave on updated, large-scale benchmarks like EMBER2024 when trained with CPU-only resources and strict memory limits.

B. Critical Analysis

Although classical ML on static features achieves high accuracy, these methods usually rely on the particular feature engineering choices and dataset features. Many studies are validated on datasets that are balanced or nearly balanced, frequently with a small range of malware families. This can inflate accuracy and hide problems like high false-positive rates

TABLE II. DL-BASED STATIC MALWARE DETECTION

Reference	Year	Platform	Representation	DL Model	Main Results
Akerele et al.[15]	2025	Mixed	Static and dynamic features.	Various DL models	DL improve detection over traditional methods, but struggle with packed/obfuscated malware and may overfit.
Salas et al.[16]	2023	Windows PE	Static feature vectors of PE files	DL architectures for classification	High detection accuracy and low error rates reported, but interpretability and robustness remain issues.
Daniel and Patel [17]	2025	Windows PE	Grayscale image of PE file	CNN-based model	accuracy = 94.78%, precision = 88.64%, recall = 94.76% on PE malware dataset.

TABLE III. HYBRID STATIC-DYNAMIC APPROACHES

Reference	Year	Platform	Features	Model	Main Results
Yousuf et al.[18]	2023	Windows PE	Four static feature sets and merged combinations	ML with ensemble learning and dimensionality reduction	Detection rate = 99.5%, error rate = 0.47% on Windows PE dataset.
Bensaoud et al.[19]	2024	Mixed	static and dynamic features	Various DL models (CNNs, RNNs, hybrids)	Hybrid models improve detection robustness but increase computational cost and deployment complexity.

TABLE IV. AUTOML AND OPTIMIZATION-FOCUSED STATIC MALWARE DETECTION

Reference	Year	Platform	Features	Model	Main Results
Brown et al. [20]	2024	Static malware datasets	Four static features	AutoML-generated Deep FFNN	AutoML models comparable in accuracy to manually crafted DL models, with automated hyperparameter tuning and architecture search.
Wu et al.[21]	2025	Mixed	static and dynamic features	Comparative analysis of ML/DL/AutoML	Highlights accuracy–cost trade-offs and notes that many high-accuracy methods require significant compute and memory.

or poor generalization to new malware variants. Furthermore, the literature often highlights detection rate or overall accuracy, while providing limited studies of false positives and model behavior in actual class-imbalance scenarios [12], [13], [14], [18].

Although DL-based methods may identify intricate patterns and minimize the need for manual feature engineering, they have a number of limitations in terms of robustness and usability. Deploying DL models for malware detection on resource-constrained systems, like user laptops or IoT devices, can be challenging because they often rely on GPU-accelerated training and inference. Additionally, these models can also be vulnerable to adversarial transformations and packing, as small modifications or obfuscation of the binary may cause significant changes in the learned representations while preserving malicious behavior [15], [19], [17], [16].

The highest reported detection rates are frequently attained by hybrid and multi-feature systems that integrate static and dynamic analysis or combine multiple static feature sets, however, an increase in complexity and consumption of resources. Collecting dynamic features requires executing samples in sandboxes or monitoring runtime behavior, which is time-consuming and not always feasible in strict real-time or large-scale environments. Furthermore, many hybrid solutions presume access to powerful servers or clusters and do not specifically address deployment constraints such as memory limits, CPU-only environments, or the requirement for quick per-sample analysis [19], [18].

Although AutoML-based methods simplify the selection of models and hyperparameters, they still usually prioritize accuracy above optimizing for limitations such as memory footprint, inference speed, or robustness under real-world workloads. AutoML pipelines can be computationally costly to operate, and current research often assesses them using offline datasets without providing comprehensive reports on resource consumption, end-to-end detection delay, or integration into operational security pipelines. Overall, there is limited focus on controlling false positives in high-volume environments, carefully quantifying detection latency, and ensuring that static ML pipelines are specifically developed for CPU-only or low-resource deployments [13], [21], [18], [20].

Furthermore, a significant discrepancy often exists between the high detection rates (95–99%) reported in literature and the performance achievable under strict operational constraints. The widely cited EMBER LightGBM baselines of $> 95\%$ typically utilize the full feature set (2,351 features) and are trained on the complete dataset (1.1 million or more samples) using high-performance hardware with substantial RAM. In contrast, our framework operates under a 'worst-case' resource scenario: a strict 16 GB RAM limit and CPU-only processing. This necessitates a deliberate reduction in model complexity, including the use of only histogram-based features and a capped sample size of 200,000 training instances. The lower accuracy of 86.99% is thus a direct result of these hardware-bound trade-offs, providing a more realistic measure of effectiveness for low-resource endpoint deployment than idealized lab-environment results.

C. Research Gap and Novelty

Current DL and hybrid malware detection models often prioritize raw accuracy at the expense of operational efficiency, requiring significant GPU power and memory that standard endpoints lack. Furthermore, existing research frequently overlooks the systematic optimization of gradient-boosting ensembles for large-scale datasets under strict CPU and memory constraints. Our research addresses these gaps by introducing a static, machine learning-based framework designed specifically for commodity hardware. Unlike computationally intensive DL or hybrid analysis, this constraint-aware pipeline utilizes memory-safe data ingestion and batch-based processing to maintain high detection accuracy while ensuring bounded resource usage.

The novelty of this framework lies in its information-theoretic approach, which seeks to maximize class separation while minimizing feature dimensionality. Rather than scaling hardware to meet data demands, we utilize feature compression and sample capping to test whether byte-frequency distributions contain sufficient latent information for identification. By employing a stacking ensemble of LightGBM, CatBoost, and XGBoost, the architecture captures inter-model variance to detect modern malware effectively, without the high latency or false-positive rates typical of less resource-constrained methods.

Beyond practical feasibility, this work establishes a new benchmark for Constraint-Aware Feature Synthesis. We provide theoretical evidence that high-entropy "shallow" features (256-byte histograms) contain sufficient latent information to achieve an AUC of 0.9473 when processed via a non-linear stacking meta-learner. This challenges the prevailing "Deep Learning-first" paradigm by proving that model architecture (stacking ensemble) can compensate for an intentional 90% reduction in feature dimensionality.

III. Methodology

The novelty of this framework lies in its Information-Theoretical approach to malware detection. While standard engineering fixes focus on scaling hardware to meet data demands, this research explores the inverse problem: how to maximize class separation using the lowest possible feature dimensionality. By compressing features to float32 and enforcing sample caps, we are not merely "saving memory"; we are testing the hypothesis that the statistical distribution of byte frequencies (histograms) contains sufficient latent information to identify modern malware, provided the ensemble architecture is deep enough to capture inter-model variances between LightGBM, CatBoost, and XGBoost.

A. Overview of Experimental Design and Constraints

1) Research objectives and constraints rationale: This study's methodological approach has been designed to address a crucial need in large-scale MLfor cybersecurity: achieving high accuracy in predictions while operating under stringent memory and computational constraints. The primary objective was to utilize the EMBER 2024 dataset and advanced ensemble MLmethods to develop and evaluate a high-accuracy and memory-optimized malware detection framework on typical CPU-based systems.

This methodology is fundamentally defined by the constraint definition, which states that the entire pipeline must operate stably on a system with only 16 GB RAM and no access to GPU acceleration. Due to this constraint, the development of novel data ingestion and modeling strategies is required. Many of the earlier studies using the EMBER feature set rely significantly on strong DL models, which frequently require cluster resources or specialized GPU hardware. In contrast, this work focuses on performance optimization under real-world memory constraints, offering a useful framework appropriate for environments with limited resources like enterprise endpoint protection systems.

The design goal was not merely to detect malware, but to prove that high detection accuracy (specifically exceeding 85% in the preliminary analysis) is achievable using carefully optimized gradient-boosting ensembles and a fully CPU-compatible pipeline. The framework of the data loading and model training approach is specifically designed to address these hardware constraints, guaranteeing that the research stays applicable to practical anti-virus solutions that cannot rely on high-performance hardware.

2) Hardware and computational environment specification: The experiment was executed on a single system limited to 16 GB of RAM and CPU-only processing without employing any GPU acceleration. The methodology included a number of important general optimization strategies to guarantee the pipeline's stability and effectiveness in this limited context. These included the methodical application of **Numpy float32** compression for feature arrays, the creation of a memory-capped loading approach to control data ingestion, and the choice of gradient boosting models recognized for their speed and efficiency on CPU clock speeds. Additionally, a necessary trade-off between predictive capability and system RAM consumption was ensured by optimizing parameters, including tree depth, leaves, and estimators, through the management of model complexity.

B. Data Sourcing, Preprocessing, and Memory-Constrained Feature Engineering

The success of the methodology depends on the rigorous control of data flow and memory usage, considering the massive size of the dataset in relation to the system resources at hand.

1) Dataset description and selection rationale: The research employed the EMBER 2024 Malware Dataset (Malware Benchmark for Research), which is a standardized, extensive, pre-extracted feature representation of the PE (Portable Executable) file structure, delivered in a JSONL (JSON Lines) file format. Each record in the binary classification process is given a label:

- Label **0** denotes a **Benign**.
- Label **1** denotes **Malware**.

The dataset is provided as several distinct files distributed across multiple training files **(train.jsonl)** and testing files **(test.jsonl)**. This volume of the dataset, with the number of samples in each file reaching hundreds of thousands, makes loading the complete feature set challenging on typical hardware configurations.

Strictly using only the **histogram** features from the JSONL records in addition to the binary **label** was an important methodological choice for feature engineering. This choice was made strategically because, in comparison to other possible EMBER features, the histogram features offer a fixed and relatively compact data size while maintaining high discriminative capability for static analysis. The methodology confirmed the commitment to memory efficiency above feature richness by focusing on increasing the total number of samples that could be processed and stored in the 16 GB RAM by extracting only the relevant histogram features.

2) Memory-limited data ingestion pipeline: Because of the size of the EMBER dataset, a unique approach to data ingestion is required. A batch-based controlled loading mechanism was developed and implemented because the dataset size, when fully loaded, exceeds the few gigabytes available on the 16 GB system after considering the operating system and model structures. This memory-safe approach is essential to the stability of the overall experiment.

a) Feature extraction and selection: The Memory-Safe JSONL Loader was implemented to read the input files line by line. During this process, it selectively extracted only the required "histogram" array (as the feature set) and the "label" (as the target variable). This selective extraction method immediately reduces the memory overhead associated with parsing and storing unnecessary metadata or complex feature structures.

b) Data compression and type casting: Explicit type casting was used to aggressively compress arrays to avoid a MemoryError during array creation (a crucial low-level methodological optimization). The Numpy float32 data type was used to represent the feature arrays, which store the histogram data, reducing the memory required for floating-point representation in half when compared to the default 64-bit standard. At the same time, Numpy int8, the smallest numerical type available, was used to represent the binary labels. This compression approach guarantees that the extensive training set, in conjunction with the memory consumed by the model parameters and intermediate boosting structures, consistently operates within the 16 GB RAM constraint.

c) Controlled batch loading and global sample capping: To maintain stability within the 16 GB RAM limit, global allocation constraints were applied to the dataset. While the EMBER 2024 dataset contains over 3.2 million files, a cap of 200,000 training samples and 80,000 testing samples was enforced to prevent system exhaustion. To mitigate the risk of reduced model capacity from this capping, a stratified random sampling strategy was employed. This ensures that the 200,000 samples maintain the original class distribution of the larger 3.2M dataset (Label 0: Benign vs. Label 1: Malware), preventing the model from developing a bias toward the majority class. This approach ensures that the "compact" training set remains a statistically representative subset of the holistic benchmark.

d) Inter-batch memory management: In addition to compression and capping, the data ingestion pipeline incorporated inter-batch garbage collection mechanisms. This active memory management technique ensures that system resources are explicitly released after the processing of each batch, preventing memory fragmentation and progressive memory creep that would otherwise lead to instability when training on hundreds of thousands of samples.

The key parameters defining the memory-limited data ingestion pipeline are summarized in Table V.

C. ML Architecture: Stacking Ensemble Design and Configuration

1) Overview of the ensemble framework rationale: A Stacking Ensemble Classifier was used to achieve the objectives without using GPU-accelerated DL architectures. The effectiveness of Gradient Boosting Machines (GBMs) with tabular data and their established optimization for CPU operations justify their employment. The stacking structure, which combines three different and strong GBMs, enables the methodology to comply with the CPU-only constraint and leverage their various strengths (speed, resilience, and generalization capability).

2) Base learner selection and configuration: Three leading gradient boosting models were selected as Base Learners, chosen specifically for their efficiency and high performance potential on tabular data.

a) LightGBM: The main reason for including this model was its effectiveness. It makes use of histogram-based gradient boosting, which, when compared to conventional tree-based methods, greatly maximizes training speed and naturally reduces memory usage, making it ideal for the volume of sample data seen in the EMBER dataset.

b) CatBoost: Selected because of its resilience and ability to manage both numerical and categorical features inside its structure, and is known for its ability to prevent overfitting, which gives the multi-model ensemble architecture the stability it needs.

c) XGBoost: A well-known and powerful gradient-boosted tree model, was incorporated due to its excellent generalization capacity and effectiveness with tabular feature architectures.

3) Training procedure and hyperparameter constraints: The training process was implemented in two phases. First, the same set of 200,000 optimized float32 histogram features was used to train each of the three base models independently.

Because only 16 GB of RAM was available, the training process had to be carefully optimized to ensure that model structures didn't exceed the available memory, which could cause the system to become unstable. As a result, all base models were purposefully pre-set with conservative configuration parameters intended to control computation time and memory usage. These characteristics included a lower learning rate, a limited tree depth, and an intermediate number of estimators. These particular configurations reflect an essential trade-off: for assurance of stability and effective use of the restricted system RAM, potential peak complexity is sacrificed.

4) Meta-learner design and integration: The core function of the Stacking Framework is the synthesis of the base learners' outputs by a Meta-Learner. The Final Estimator selected for the stacking architecture was a smaller LightGBM

TABLE V. DATA INGESTION AND CONTROLLED SAMPLING PARAMETERS FOR EMBER 2024

Parameter	Value/Description	Methodological Rationale
Data Source	EMBER 2024 Static Features (JSONL).	Standardized, large-scale PE feature set.
Features Extracted	"histogram" features only	Optimization for memory efficiency via fixed-length, core static features.
Feature Data Type	Numpy float32	Critical compression technique to minimize feature memory footprint.
Label Data Type	Numpy int8	Optimization for binary classification target.
Batch Loading Limit	15,000 samples	Empirically derived threshold to prevent MemoryError during ingestion.
Total Training Samples	200,000 (Maximum Cap)	Ensures stable model training within 16 GB RAM constraints.
Total Testing Samples	80,000 (Maximum Cap)	Guarantees a stable and representative test evaluation set.

model. The efficiency of LightGBM minimizes the additional computational overhead of this final synthesis step.

The integration process involves collecting the predictions generated by the three base learners. The LightGBM classifier receives these predictions as input after they are processed as meta-features. These meta-features are used to train the final model, which learns the optimal way to integrate the strengths and weaknesses of the three base algorithms to produce a synthesized classification decision that typically performs effectively.

The architectural components of the implemented Stacking Ensemble are summarized in Table VI.

IV. EXPERIMENTAL SETUP AND EVALUATION

This section outlines the application of the framework to the experimental environment, details the comprehensive evaluation protocol, and presents the final performance metrics achieved by the Stacking Ensemble model.

A. Hardware and Computational Environment Specification

All the experiments, such as feature engineering, data ingestion, model training, and evaluation, were performed on a system that was rigorously limited to 16 GB RAM and used CPU-only processing without GPU acceleration. The pipeline was optimized for this constrained hardware by using Numpy float32 compression for data structures, memory-limited loading, and gradient boosting models for effective CPU operation.

B. Performance Evaluation

The model's performance was measured against the 80,000-sample test set using a comprehensive protocol designed for operational cybersecurity assessment.

1) Standard binary classification metrics: The model's predictive capability was quantitatively assessed using a comprehensive set of standard binary classification metrics:

a) Accuracy: The overall proportion of correct classifications.

b) Precision: The ratio of correctly identified positive cases (Malware) to all cases predicted as positive.

c) Recall (True positive rate): The ratio of correctly identified positive cases to all actual positive cases.

d) F1-Score: The harmonic mean of Precision and Recall.

The F1-Score was specifically prioritized as a key performance indicator. This metric is crucial for dependable detection systems because it offers a strong performance indicator that achieves a balance between the need to reduce false alarms (Precision) and the need to reduce missed threats (Recall), making it ideal for operational evaluation in a real-world malware detection scenario.

2) Advanced assessment: Evaluation also included protocols to assess performance independent of classification thresholds and against potential real-world data skew:

a) ROC curve and AUC score: The Receiver Operating Characteristic (ROC) curve was generated to evaluate performance across all possible classification thresholds, with the Area Under the Curve (AUC) serving as a robust, threshold-independent measure of class separability.

b) Precision-Recall curve: The Precision-Recall (PR) curve was mandated for evaluation, as it is superior to the ROC curve in scenarios where minimizing False Negatives (FN) is paramount, focusing specifically on the performance of the positive class.

3) Error analysis: Confusion matrix: The Confusion Matrix was employed for error analysis, quantifying the specific distributions of True Positives (TP), True Negatives (TN), False Positives (FP), and FN. This is vital for operational risk assessment, allowing for objective interpretation of the model's tolerance for security risk versus user inconvenience.

V. RESULTS AND DISCUSSION

A. Classification Performance Results

The final performance of the Stacking Ensemble Classifier on the 80,000-sample test set yielded an overall accuracy of **86.99%**. The detailed classification results across both classes (0: Benign; 1: Malware) are presented in Table VII.

The AUC score, derived from the ROC curve illustrated in Fig. 1, reached 0.9473. This high value signifies the model's strong ability to separate the two classes (Benign vs. Malware) across all possible classification thresholds, providing a robust, threshold-independent measure of overall model quality. The ROC curve plots the True Positive Rate against the False Positive Rate, visually confirming this strong separability.

Detailed error analysis was performed using the confusion matrix that is illustrated in Fig. 2.

TABLE VI. STACKING ENSEMBLE ARCHITECTURE COMPONENTS

Component	Primary Role/Advantage	Key Methodological Constraint
LightGBM model (Base Learner)	Quick training via histogram boosting, minimal memory footprint.	Limited tree depth/estimators to manage RAM usage.
CatBoost model (Base Learner)	Robustness to overfitting, strong classification performance.	Limited tree depth/estimators to manage RAM usage.
XGBoost model (Base Learner)	High generalization capability and efficiency on tabular data.	Limited tree depth/estimators to manage RAM usage.
Smaller LightGBM model (Meta-Learner)	Synthesis of meta-features for final prediction, efficient final layer.	Optimized to avoid late-stage memory bottlenecks.

TABLE VII. DETAILED CLASSIFICATION RESULTS

	0 (Benign)	1 (Malware)	Accuracy	Macro Avg.	Weighted Avg.
Precision	0.86	0.88		0.87	0.87
Recall	0.90	0.84		0.87	0.87
F1-Score	0.88	0.86	0.87	0.87	0.87
Support	42,046	37,954	80,000	80,000	80,000

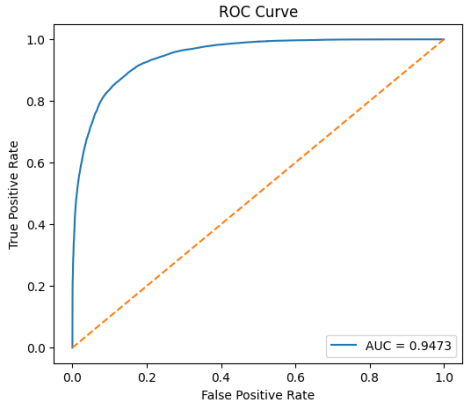


Fig. 1. ROC Curve.

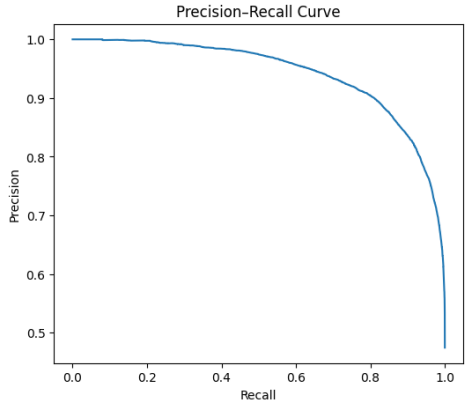


Fig. 3. Precision-Recall curve.

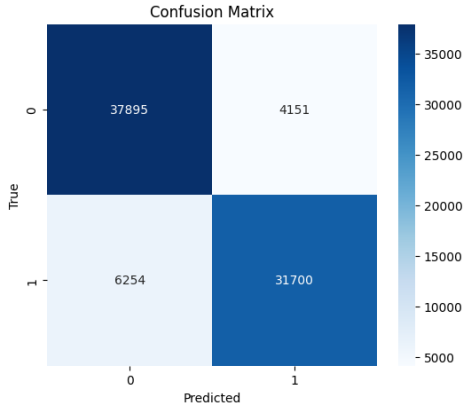


Fig. 2. Confusion matrix for the stacking ensemble classifier.

- TN: 37,895 (Correctly identified Benign files)
- FP: 4,151 (Benign files incorrectly flagged as Malware)
- FN: 6,254 (Malware files incorrectly identified as Benign)
- TP: 31,700 (Correctly identified Malware files)

Furthermore, the generation of the PR curve, illustrated in Fig. 3, was a mandatory evaluation component. This curve, which plots Precision against Recall, is methodologically superior to the ROC curve for practical deployment, particularly in real-world scenarios that often face severe class imbalance, as it specifically focuses on maximizing the performance of the positive class (Malware detection), where minimizing missed threats (FN) is paramount.

B. Proposed Framework Addresses the Identified Gaps

The framework's achieved accuracy of 86.99% must be interpreted within the context of its operational constraints. While prior work utilizing DL or hybrid analysis reports accuracy exceeding 98%, those studies typically assume access to GPU acceleration or fail to quantify the memory footprint required for such performance. By utilizing the EMBER 2024 dataset, this research evaluates the model against over 3.2 million files—a scale significantly larger than the datasets used in many papers reporting $> 95\%$ accuracy.

Our framework fills the identified gap by providing a memory-efficient data loader and a stacking ensemble that

maintains a high AUC of 0.9473 without GPU support. This demonstrates that even when accuracy is lower than "lab-environment" results, the model remains scientifically robust and operationally superior for real-world deployment on CPU-only, memory-limited systems.

1) Impact of sample capping on model capacity: The decision to cap training at 200,000 samples represents a deliberate methodological trade-off between peak predictive capacity and hardware-bound feasibility. While increasing the sample size toward the 3.2M limit would likely improve the accuracy and F1-score beyond the reported 86.99%, such an increase would lead to a MemoryError on standard 16 GB RAM systems without GPU support. By using stratified sampling rather than a simple temporal or random cut, the framework achieves a high AUC of 0.9473. This proves that a carefully curated, smaller subset can still capture the core "histogram" features necessary for effective static detection in resource-constrained environments.

C. The Performance Gap

The delta between our observed AUC (0.9473) and the published EMBER baselines is an intentional trade-off resulting from our 'Privacy and Resource' constraints. Standard EMBER baselines rely heavily on 'String' features and 'Import/Export' tables, which are highly predictive but computationally expensive to parse and can be easily obfuscated by packers. By restricting our input to raw byte-histograms, we eliminate the parsing overhead and increase resilience against structural obfuscation, albeit at the cost of the 5% AUC gap. This gap represents the 'Complexity Tax' required to ensure that the system remains functional on legacy 16GB RAM hardware without crashing during the feature extraction phase.

VI. Conclusion

This research successfully developed a CPU-optimized malware detection framework using the EMBER 2024 dataset, demonstrating that high-performance security is achievable without specialized GPU hardware. While recent literature often reports detection rates exceeding 98%, these metrics are frequently produced in resource-rich environments using deep learning models that are difficult to deploy on constrained endpoints. Our framework, which achieved an accuracy of 86.99% and an AUC of 0.9473, provides a critical analytical benchmark: it proves that optimized gradient-boosting ensembles can bridge the gap between theoretical accuracy and operational feasibility.

The synthesis of LightGBM, CatBoost, and XGBoost yields a robust classification boundary that remains stable even when training capacity is capped to meet strict hardware limits. The performance achieved here suggests that for real-world IoT and endpoint security, a "resource-first" design philosophy may be more effective than "accuracy-first" deep learning approaches that suffer from high latency and memory overhead. Our results indicate that the slight reduction in accuracy compared to state-of-the-art DL models is a justifiable trade-off for the ability to process millions of samples on commodity hardware.

Future work should investigate if this performance gap can be narrowed through advanced sampling strategies, such as temporal or active learning, to maximize the utility of the 200,000-sample training cap. Additionally, evaluating the framework's resilience against adversarial obfuscation will further define its role as a viable, low-cost alternative to signature-based and deep-learning detection systems in modern cybersecurity infrastructures.

Funding

This study was funded by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [GRANT No. KFU260954].

Acknowledgment

The authors extend their appreciation to the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia [GRANT No. KFU260954]. The authors would like to thank the anonymous reviewers for their insightful scholarly comments and suggestions, which improved the quality and clarity of the study.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Y. Song, D. Zhang, J. Wang, Y. Wang, Y. Wang, and P. Ding, "Application of deep learning in malware detection: a review," *Journal of Big Data*, vol. 12, no. 1, p. 99, 2025.

[2] G. SHAN and N. SHARMA, "Malware analysis and detection using ml tools: Current state and challenges," *INTERNATIONAL JOURNAL OF ENGINEERING*, vol. 73, no. 1, pp. 371–384, 2025.

[3] N. Alyemni and M. Frikha, "Exploring machine learning in malware analysis: Current trends and future perspectives." *International Journal of Advanced Computer Science & Applications*, vol. 16, no. 1, 2025.

[4] F. A. Aboaoja, A. Zainal, F. A. Ghaleb, B. A. S. Al-Rimy, T. A. E. Eisa, and A. A. H. Elnour, "Malware detection issues, challenges, and future directions: A survey," *Applied Sciences*, vol. 12, no. 17, p. 8482, 2022.

[5] R. Kumar and S. Geetha, "Malware classification using xgboost-gradient boosted decision tree," *Adv. Sci. Technol. Eng. Syst*, vol. 5, no. 5, pp. 536–549, 2020.

[6] P. R. Kothamali and S. Banik, "Limitations of signature-based threat detection," *Revista de Inteligencia Artificial en Medicina*, vol. 13, no. 1, pp. 381–391, 2022.

[7] Y. Pratama, R. S. Munzi, A. B. Mustafa, I. L. Kharisma *et al.*, "Static malware detection and classification using machine learning: A random forest approach," *Engineering Proceedings*, vol. 107, no. 1, p. 76, 2025.

[8] D. Dhungana, A. Sapkota, S. Pokharel, S. Devkota, and B. H. Paudel, "Malware classification using static analysis approaches," *Journal of Artificial Intelligence*, vol. 6, no. 4, pp. 494–511, 2024.

[9] H. Anderson and P. Roth, "Ember: An open dataset for training static pe malware machine learning models," *ArXiv*, vol. abs/1804.04637, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:4888440

[10] R. J. Joyce, G. Miller, P. Roth, R. Zak, E. Zaresky-Williams, H. Anderson, E. Raff, and J. Holt, "Ember2024-a benchmark dataset for holistic evaluation of malware classifiers," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, 2025, pp. 5516–5526.

[11] Kamdan, Y. Pratama, R. S. Munzi, A. B. Mustafa, and I. L. Kharisma, "Static malware detection and classification using machine learning: A random forest approach," *Engineering Proceedings*, vol. 107, no. 1, p. 76, 2025.

[12] A. S. Shatnawi, Q. Yassen, and A. Yateem, "An android malware detection approach based on static feature analysis using machine learning algorithms," *Procedia Computer Science*, vol. 201, pp. 653–658, 2022.

[13] Q. Wu, X. Zhu, and B. Liu, "A survey of android malware static detection technology based on machine learning," *Mobile Information Systems*, vol. 2021, no. 1, p. 8896013, 2021.

[14] C. K. Yuk and C. J. Seo, "Static analysis and machine learning-based malware detection system using pe header feature values," *Int. J. Innov. Res. Sci. Stud*, vol. 5, pp. 281–288, 2022.

[15] S. Akerele, N. Adebola, O. Fagbohun *et al.*, "Modern deep learning approaches for malware detection and classification," *J Artif Intell Mach Learn & Data Sci 2025*, vol. 3, no. 3, pp. 2761–2768.

[16] M. I. P. Salas and P. De Geus, "Static analysis for malware classification using machine and deep learning," in *2023 XLIX Latin American Computer Conference (CLEI)*. IEEE, 2023, pp. 1–10.

[17] A. B. Daniel and N. Patel, "Detection of malware on portable executable files using machine learning," in *2025 International Conference on Networks & Advances in Computational Technologies (NetACT)*. IEEE, 2025, pp. 1–6.

[18] M. I. Yousuf, I. Anwer, A. Riasat, K. T. Zia, and S. Kim, "Windows malware detection based on static analysis with multiple features," *PeerJ Computer Science*, vol. 9, p. e1319, 2023.

[19] A. Bensaoud, J. Kalita, and M. Bensaoud, "A survey of malware detection using deep learning," *Machine Learning With Applications*, vol. 16, p. 100546, 2024.

[20] A. Brown, M. Gupta, and M. Abdelsalam, "Automated machine learning for deep learning based malware detection," *Computers & Security*, vol. 137, p. 103582, 2024.

[21] Y. Wu, H. Zhuang, Y. Jia, and Y. Zhang, "A survey of machine learning approaches for malware detec-tion," in *Proceedings of the 2025 5th International Conference on Computer Network Security and Software Engineering*, 2025, pp. 269–273.

[22] D. G. Corlatescu, A. Dinu, M. P. Gaman, and P. Sumedrea, "Embersim: A large-scale databank for boosting similarity search in malware analysis," *Advances in Neural Information Processing Systems*, vol. 36, pp. 26 722–26 743, 2023.

Stability-Aware QUBO Feature Selection for Tabular Classification Under Repeated Nested Cross-Validation

Marco Fidel Mayta Quispe[1], Leonid Alemán Gonzales[2], Charles Ignacio Mendoza Mollocondo[3], Nayer Tumi Figueroa[4], Juan Carlos Juarez Vargas[5], Godofredo Quispe Mamani[6]
Instituto de Investigación en Inteligencia Computacional y Ciencia de Datos (IIICCD)
Universidad Nacional del Altiplano, Puno, Perú[1,2]
Universidad Nacional del Altiplano, Puno, Perú[3,4,5,6]

Abstract*—Quadratic Unconstrained Binary Optimization (QUBO) provides a principled framework for feature selection by encoding relevance–redundancy trade-offs and explicit constraints directly in a combinatorial objective. This study presents a stability-aware QUBO pipeline for tabular binary classification, evaluated on two standard benchmarks, namely Breast Cancer Wisconsin Diagnostic (569 samples, 30 features) and Pima Indians Diabetes (768 samples, 8 features; clinically invalid zeros treated as missing and imputed within folds). We study four QUBO variants spanning a base relevance–redundancy formulation, an exact-cardinality formulation enforcing a fixed budget k, a stability-regularized formulation that incorporates bootstrap uncertainty estimates of relevance and redundancy directly into the QUBO objective, and a performance-weighted relevance variant based on inner-CV univariate utility. All methods are assessed under repeated nested stratified cross-validation (5 outer folds $\times$ 3 repeats, $n = 15$ outer test evaluations), reporting AUC-ROC, AUC-PR, MCC, and Brier score with 95% confidence intervals, alongside selection stability via mean Jaccard similarity across outer-fold selected subsets. Results show that QUBO-based selection is competitive with strong classical baselines (RFECV, L1-logistic, permutation-importance ranking, and mutual information) while enabling strict budget control and transparent stability diagnostics. On the near-ceiling Breast Cancer benchmark, predictive differences are marginal and the main differentiators become subset-size control and stability; on Pima, QUBO-k remains competitive while enforcing strict cardinality constraints. These findings support QUBO as a practical framework when budgeted, interpretable, and reproducible feature selection is required, though evaluation is limited to low-dimensional tabular settings.

Keywords*—Feature selection; QUBO; simulated annealing; nested cross-validation; selection stability; Jaccard similarity; probability calibration; tabular classification

I. Introduction

Feature selection is a core component of tabular machine learning because it influences generalization, interpretability, and operational cost. In real deployments, the objective is often not only to maximize discrimination, but also to obtain compact and reproducible feature sets under explicit constraints (e.g., acquisition budgets, latency limits, or mandated subset sizes). Classical families—filters, wrappers, and embedded methods—offer useful trade-offs: filters are fast but may miss multivariate structure, wrappers can be accurate but costly and variance-prone, and embedded methods depend on model-specific inductive biases.

Quadratic Unconstrained Binary Optimization (QUBO) offers a unified way to encode feature selection as a single combinatorial objective balancing relevance and redundancy while supporting constraints through penalty terms [1]. This becomes attractive when practitioners need *explicit control* over subset size and a transparent assessment of *selection stability*. In correlated feature regimes, instability often arises because several variables carry overlapping predictive information and small resampling perturbations can swap one correlated surrogate for another; QUBO addresses this structurally through pairwise redundancy penalties that explicitly discourage co-selection of correlated features. Moreover, because feature selection itself is part of the learning pipeline, rigorous evaluation is essential to avoid optimistic bias; nested cross-validation is the standard remedy when selection and calibration/tuning are performed within training folds [3], [4].

We evaluate on two benchmark datasets representing distinct dimensionality regimes, namely `breast_cancer` ($d = 30$) and `pima_diabetes` ($d = 8$). To enable fair comparisons under a fixed feature budget, we benchmark selectors under target budgets k (top-k filters and QUBO variants), while also reporting realized cardinality for procedures that determine k internally (e.g., RFECV and L1-logistic).

Contributions: This work presents QUBO feature selection variants that encode relevance–redundancy trade-offs, explicit k-budgets, and stability regularization within a unified combinatorial objective. These variants are benchmarked against strong classical baselines using repeated nested stratified CV with confidence intervals across discrimination, classification-quality, and calibration metrics. In addition, selection reproducibility is quantified via mean pairwise Jaccard similarity across outer-fold selections, treating stability as a first-class criterion alongside predictive performance.

II. Related Work

Feature selection (FS) is commonly categorized into *filter*, *wrapper*, and *embedded* methods. Filter methods rank variables independently of the final predictor; mutual-information criteria are popular when nonlinear dependencies are expected, and mRMR is a canonical formulation balancing relevance to the target and redundancy among selected variables [5]. Wrapper methods evaluate subsets through downstream performance, often improving accuracy at higher computational

cost and sensitivity to the resampling design [10]. RFE and RFECV are widely used wrapper procedures that iteratively remove features based on model utility and are influential in biomedical pipelines [11]. Embedded methods incorporate selection within training; L1-regularization (LASSO) induces sparsity and performs implicit selection [6].

Regarding stability mechanisms, filters such as MI ranking are stable only when the top-k features have clearly separated relevance scores; in correlated regimes, near-tied features can be exchanged across resamples. Wrappers inherit instability from the downstream model and the elimination path, while embedded methods such as LASSO can produce multiple near-equivalent sparse solutions in correlated settings. Stability selection [7] addresses this by aggregating selection frequencies over subsamples, but it does not enforce hard cardinality constraints. The present QUBO formulation instead incorporates stability directly into the combinatorial objective via bootstrap uncertainty penalties, maintaining hard budget control within a single optimization step.

Beyond accuracy, *stability* is critical whenever FS supports interpretation or scientific conclusions. Stability selection formalizes robustness via subsampling and selection frequencies [7], while broader analysis show that unstable selectors can undermine reproducibility even under high predictive performance [8]. This motivates reporting stability metrics (e.g., Jaccard similarity) jointly with discrimination and calibration.

Rigorous evaluation is particularly important because selecting features on the same data used to estimate performance can bias results upward. Nested cross-validation mitigates this by isolating all selection/tuning within inner loops and evaluating only on held-out outer folds [3], [4]. Calibration can diverge from discrimination; probability quality should be assessed under the same leakage-safe protocol [12], [13].

Optimization-based FS treats subset selection as a combinatorial problem. Simulated annealing (SA) is a classic metaheuristic for large discrete spaces and provides a reproducible baseline for approximate minimization under fixed compute budgets [9]. QUBO offers a unifying representation for such objectives, enabling relevance–redundancy trade-offs and hard/soft constraints via quadratic penalties [1]. Prior QUBO-style feature selection formulations [2] have focused on relevance–redundancy encoding and cardinality control; the present work extends this line by integrating bootstrap uncertainty-based stability regularization into the QUBO objective and evaluating under a leakage-safe repeated nested cross-validation protocol that jointly reports discrimination, calibration, and subset reproducibility.

III. Methodology

A. Problem Setting

Let $\mathcal{D} = \{(\mathbf{x}^{(n)}, y^{(n)})\}_{n=1}^{N}$ be a labeled dataset with $\mathbf{x}^{(n)} \in \mathbb{R}^d$ and $y^{(n)} \in \{0, 1\}$.

Feature selection aims to identify a subset $S \subseteq \{1, \ldots, d\}$ that preserves predictive performance while controlling redundancy and improving interpretability under a constrained feature budget.

We encode selection using a binary vector $\mathbf{z} \in \{0, 1\}^m$ over a candidate pool of size $m \leq d$, where $z_i = 1$ indicates that candidate feature i is selected, and $S(\mathbf{z}) = \{i : z_i = 1\}$.

B. Datasets and Preprocessing (Leakage Control)

We use Breast Cancer Wisconsin Diagnostic (569 samples, 30 numeric features) [16], [17] and Pima Indians Diabetes (768 samples, 8 numeric features) [18]. For Pima, clinically invalid zeros in specific measurements (commonly including glucose, blood pressure, skin thickness, insulin, BMI) are treated as missing and imputed.

We perform all preprocessing *within the training portion of each outer fold* to prevent information leakage. We impute missing values using the training median, and features are standardized via z-score scaling (fit on training, applied to validation/test). Any dataset-specific corrections are applied inside the same fold-isolated preprocessing.

C. Compared Methods and Feature Budgets

To compare methods under explicit budgets, we define target cardinalities k and evaluate rank-based baselines returning exactly k variables (MI-top$k$, permutation-importance top$k$), QUBO variants matched to the same k target, and methods that determine subset size internally (L1-logistic and RFECV), for which we report realized cardinality. In our experiments, $k = 10$ for `breast_cancer` and $k = 6$ for `pima_diabetes`.

D. Candidate Pool Construction (for QUBO)

Within each outer training fold, we build a candidate pool by ranking features using mutual information computed on standardized training data, retaining the top candidates:

$$m = \min(25, d)$$

For `breast_cancer`, this yields $m = 25$ out of $d = 30$, while for `pima_diabetes` ($d = 8$) the candidate pool equals the full feature set ($m = d$). All QUBO objectives are constructed and optimized only over this candidate pool.

E. Relevance and Redundancy Signals

1) Relevance: For each candidate feature i, we compute mutual information relevance:

$$r_i = I(X_i; Y), \tag{1}$$

estimated with a nonparametric MI estimator (e.g., as implemented by `mutual_info_classif`).

2) Redundancy: To discourage selecting overlapping variables, we compute pairwise redundancy via absolute Pearson correlation on standardized training data:

$$\rho_{ij} = |\mathrm{corr}(X_i, X_j)|. \tag{2}$$

F. Base QUBO Objective

Let m denote the number of candidates. The base QUBO energy is given by:

$$\min_{\mathbf{z}\in\{0,1\}^m} E(\mathbf{z}) = \sum_{i=1}^{m}(\lambda - r_i)z_i + \alpha \sum_{i<j} \rho_{ij}\, z_i z_j, \qquad (3)$$

where, $\alpha > 0$ controls redundancy and $\lambda \geq 0$ encourages sparsity.

G. Exact-Cardinality QUBO-k

To target a subset size k, we add a quadratic penalty:

$$E_k(\mathbf{z}) = E(\mathbf{z}) + \mu \left(\sum_{i=1}^{m} z_i - k \right)^2, \qquad (4)$$

with $\mu > 0$ controlling constraint strength.

1) Deterministic exact-k projection (budget-matched evaluation): Because heuristic solvers may return $|S(\mathbf{z})| \neq k$ and because we compare methods under a fixed budget, we apply a deterministic projection step for the QUBO family. When $|S| > k$, we keep the k features in S with highest MI relevance; when $|S| < k$, we fill with remaining candidates with highest MI. This ensures budget-matched evaluation for all QUBO variants.

H. Stability-Regularized QUBO

To promote robustness, we estimate uncertainty of relevance and redundancy via bootstrap resampling of the outer training fold. For bootstrap replicates $b = 1, \ldots, B$, we compute $r_i^{(b)}$ and $\rho_{ij}^{(b)}$ and estimate means and variances as:

$$\bar{r}_i = \mathbb{E}[r_i^{(b)}], \qquad \mathrm{Var}(r_i) = \mathrm{Var}(r_i^{(b)}), \qquad (5)$$

$$\bar{\rho}_{ij} = \mathbb{E}[\rho_{ij}^{(b)}], \qquad \mathrm{Var}(\rho_{ij}) = \mathrm{Var}(\rho_{ij}^{(b)}). \qquad (6)$$

We then define a stability-regularized objective as:

$$E_s(\mathbf{z}) = \sum_{i=1}^{m} \Big(\lambda - \bar{r}_i + \gamma\, \mathrm{Var}(r_i)\Big) z_i + \sum_{i<j} \Big(\alpha\, \bar{\rho}_{ij} + \delta\, \mathrm{Var}(\rho_{ij})\Big) z_i z_j, \qquad (7)$$

where, $\gamma, \delta \geq 0$ weight robustness penalties. As with other QUBO variants, we enforce the fixed budget via projection.

I. Performance-Weighted Relevance QUBO ("Wrapper" Variant)

To incorporate supervised signals beyond MI, we compute a *performance-weighted* relevance proxy from an inner CV performed on the outer training fold. For each candidate feature i, we estimate its univariate predictive utility via class-weighted logistic regression, producing a score such as inner-fold AUC-ROC. Let $\mathrm{AUC}_i^{(t)}$ denote univariate AUC-ROC on inner split t; we define:

TABLE I. HYPERPARAMETER CONFIGURATION USED IN ALL EXPERIMENTS.

Parameter	Symbol	Value	Scope
Diagonal offset	λ	0.02	All QUBO variants
Redundancy weight	α	0.20	All QUBO variants
Cardinality penalty	μ	1.5	QUBO-k only
Relevance var. penalty	γ	0.25	QUBO-Stability
Redundancy var. penalty	δ	0.10	QUBO-Stability
Wrapper var. penalty	η	0.15	QUBO-Wrapper
Bootstrap replicates	B	40	QUBO-Stability
Inner CV folds	K'	3	QUBO-Wrapper

$$\tilde{r}_i = \frac{1}{K'} \sum_{t=1}^{K'} \mathrm{AUC}_i^{(t)}, \qquad v_i = \mathrm{Var}\big(\mathrm{AUC}_i^{(t)}\big). \qquad (8)$$

We integrate this into the QUBO objective as:

$$E_w(\mathbf{z}) = \sum_{i=1}^{m} \Big(\lambda - \tilde{r}_i + \eta\, v_i\Big) z_i + \alpha \sum_{i<j} \rho_{ij}\, z_i z_j, \qquad (9)$$

with $\eta \geq 0$. We emphasize that this variant uses univariate inner-CV utility as a relevance proxy; it is not a full multivariate wrapper search over subsets. We enforce budget matching via projection.

J. QUBO Optimization via Simulated Annealing

We optimize all QUBO objectives with simulated annealing (SA), a temperature-controlled stochastic search that approximates minima of binary quadratic energy landscapes [9]. The solver returns a binary solution $\mathbf{z}$ and its energy, followed by exact-k projection when applicable.

K. Practical Interpretation of QUBO Hyperparameters

The QUBO coefficients have practical implications for deployment. The parameter λ acts as a diagonal offset, so that when $\lambda < r_i$ the linear term favors selection of feature i, and when $\lambda > r_i$ it penalizes selection. The redundancy weight α controls tolerance to correlated predictors. In QUBO-k, the penalty μ governs cardinality compliance strength. In the stability-regularized variant, γ and δ penalize uncertainty in relevance and redundancy estimates, respectively, improving subset reproducibility at the cost of potentially excluding features whose utility is high but unstable across resamples. In the performance-weighted variant, η trades average utility against robustness. Table I reports all hyperparameter values used.

We fixed all values prior to experimental runs using training-side information only, consistent with the leakage-safe nested CV protocol.

L. Solver Configuration and Reproducibility

Because simulated annealing is stochastic, reproducibility depends on solver configuration choices including temperature schedule, iteration budget, and random seed policy. In this work, we use a standardized SA configuration held constant across all QUBO variants so that comparisons are made under matched procedural conditions. Repeated outer folds partially average stochastic effects, but this should not be interpreted as a full robustness analysis with respect to solver seeds or annealing schedules. A more complete reproducibility study should explicitly vary solver settings and quantify optimizer-induced variance separately from data-resampling variance.

M. Downstream Classifier and Probability Calibration

For fairness across selectors, we evaluate all selected subsets using the same downstream predictor: class-weighted logistic regression producing probabilities calibrated inside the training loop [12], [13]. We report discrimination (AUC-ROC, AUC-PR), classification quality (MCC), and calibration (Brier score). This is important because models with similar AUC can differ materially in probability quality.

N. Evaluation Protocol

We use repeated nested stratified cross-validation to avoid optimistic bias in feature selection and calibration [3], [4]. We run $R = 3$ repeats of $K = 5$ outer folds, yielding $n = 15$ outer test evaluations per method. All selection steps (candidate pooling, bootstraps, and performance-weighted relevance computation) are performed strictly within each outer training fold.

We summarize metrics across the $n = 15$ outer test evaluations using means and 95% confidence intervals. For paired comparisons of AUC-ROC under matched splits, we apply the Wilcoxon signed-rank test [15] (optionally with multiplicity control when many pairwise tests are reported).

Selection stability is quantified by mean pairwise Jaccard similarity across the n selected subsets,

$$J(S_a, S_b) = \frac{|S_a \cap S_b|}{|S_a \cup S_b|}. \tag{10}$$

IV. Results

A. Nested CV Performance

Fig. 1 and Fig. 2 summarize mean AUC-ROC across the repeated nested CV outer tests. Breast Cancer is near-separable for multiple selectors (AUC-ROC close to 1.0), while Pima is moderately challenging (AUC-ROC around 0.83–0.84), where differences are smaller and uncertainty overlaps across strong methods.

B. ROC Curve Analysis (Qualitative)

Fig. 3 and Fig. 4 show ROC curves for qualitative comparison of ranking behavior across thresholds. These plots should be interpreted as illustrative summaries of discrimination; all inferential claims in this study rely on the repeated nested CV outer-test scores reported in Table II and Table III.

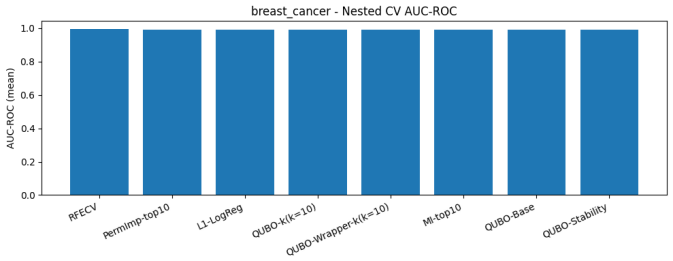


Fig. 1. Breast cancer: Mean nested CV AUC-ROC by method.

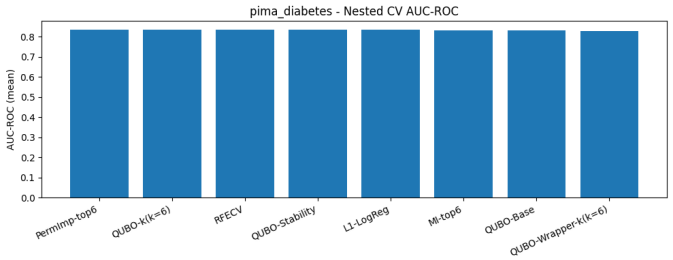


Fig. 2. Pima diabetes: Mean nested CV AUC-ROC by method.

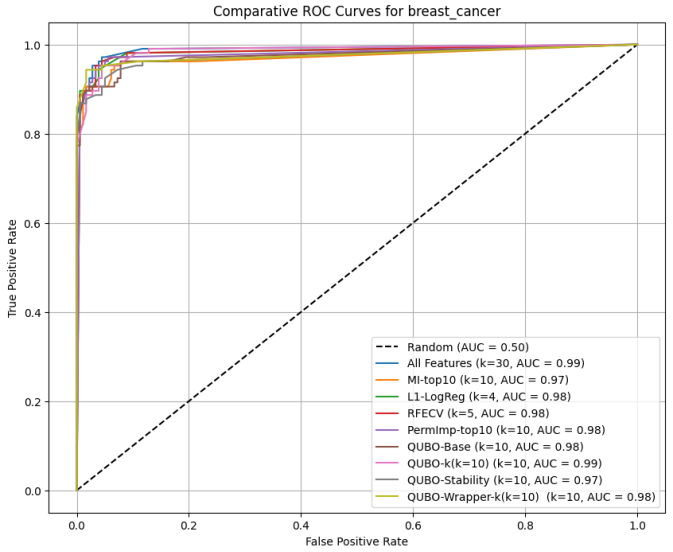


Fig. 3. Comparative ROC curves for breast cancer (illustrative).

For Breast Cancer (Fig. 3), most methods achieve AUC-ROC above 0.98, consistent with a ceiling regime in which marginal AUC differences are difficult to interpret practically. QUBO-k and QUBO-Stability closely track strong baselines (RFECV, L1-logistic), indicating that enforcing combinatorial constraints does not substantially degrade discrimination on this benchmark.

For Pima Diabetes (Fig. 4), separation among curves is more apparent, reflecting a harder decision boundary. QUBO-k remains competitive with permutation-importance and RFECV while enforcing strict cardinality constraints. The performance-weighted relevance variant (labeled "QUBO-Wrapper-k" in tables/figures for consistency with artifact names) shows slightly

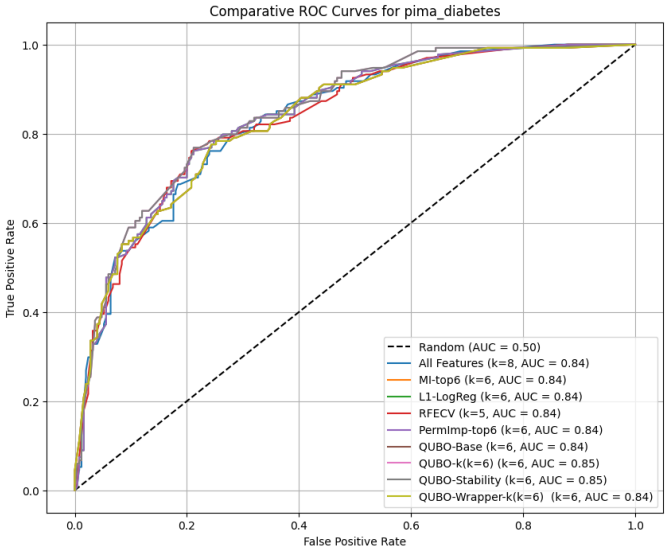


Fig. 4. Comparative ROC curves for Pima diabetes (illustrative).

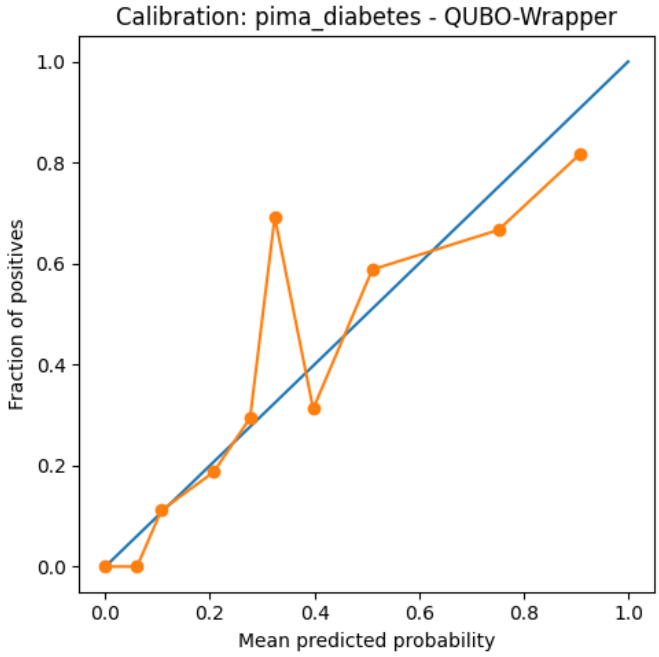


Fig. 5. Calibration curve (reliability diagram) for Pima diabetes using the performance-weighted QUBO variant. The diagonal indicates perfect calibration.

weaker ROC behavior, aligning with the nested CV summaries and suggesting that inner-loop univariate utility signals do not necessarily improve multivariate subset generalization in low-dimensional settings.

C. Calibration

Beyond discrimination, we inspect calibration using Brier score and reliability diagrams. Fig. 5 shows a calibration curve (reliability diagram) for Pima Diabetes for one representative method; deviations from the diagonal indicate over- or under-confidence. Calibration assessment is reported alongside discrimination because near-identical AUC can mask materially different probability quality.

D. Quantitative Summary with 95% Confidence Intervals

Table II and Table III report mean performance and 95% confidence intervals from the repeated nested CV outer tests ($n = 15$). We include discrimination (AUC-ROC, AUC-PR), classification quality (MCC), and calibration (Brier), plus the effective subset size k and selection time. Note that the reported selection times are implementation-dependent wall-clock measurements useful for descriptive comparison under the same environment, but not a fine-grained decomposition of solver versus non-solver cost; values displayed as 0.000 reflect rounding at the reported precision and do not imply zero computational cost.

E. Selection Stability (Jaccard)

Table IV reports mean Jaccard similarity across outer-fold selections. Stability should be interpreted jointly with performance: highly stable selectors are valuable for interpretability, but stability alone can be misleading if performance is low or if the procedure is effectively deterministic for trivial reasons [8].

F. Statistical Comparison

We apply paired Wilcoxon signed-rank tests [15] over the $n = 15$ outer AUC-ROC scores. On Breast Cancer, differences among top-performing methods are not practically distinguishable given overlapping confidence intervals. On Pima, small but consistent differences can appear under matched splits; however, because multiple pairwise comparisons can inflate Type-I error, multiplicity control (e.g., Holm correction) is recommended when reporting many p-values.

V. Discussion

All inferential statements in this section are based exclusively on outer-fold test predictions from the repeated nested CV protocol; ROC and calibration plots are included for qualitative illustration and should not be used as sole evidence of superiority. Nested CV is essential here because feature selection (and any calibration/tuning) is performed on training data, and evaluating on non-isolated splits can inflate performance estimates [3].

A key observation is the presence of a *ceiling regime* on `breast_cancer`. Multiple methods attain AUC-ROC close to 1.0, and confidence intervals overlap. In such cases, it is statistically and practically difficult to justify strong claims based on marginal AUC differences. This does not diminish QUBO; rather, it clarifies its value proposition: QUBO is often most useful when the practitioner must enforce explicit constraints (strict cardinality, domain rules, stability penalties, or cost terms) and wants a transparent optimization objective.

`pima_diabetes` operates in a moderate-difficulty regime where performance is lower and variability across folds is more informative. Here, QUBO-k remains competitive

TABLE II. BREAST CANCER (REPEATED NESTED CV; $n = 15$ OUTER TESTS): MEAN METRICS WITH 95% CI

Method	AUC-ROC [CI]	AUC-PR [CI]	MCC [CI]	Brier [CI]	k (mean)	Sel. time (s)
RFECV	0.9931 [0.9898, 0.9965]	0.9899 [0.9857, 0.9942]	0.9354 [0.9154, 0.9554]	0.0232 [0.0174, 0.0291]	17.93	0.000
PermImp-top10	0.9925 [0.9887, 0.9962]	0.9872 [0.9814, 0.9929]	0.9235 [0.9006, 0.9464]	0.0270 [0.0200, 0.0340]	10.00	0.000
L1-LogReg	0.9923 [0.9880, 0.9966]	0.9893 [0.9838, 0.9947]	0.9331 [0.9153, 0.9510]	0.0244 [0.0179, 0.0309]	9.13	0.000
QUBO-k (k=10)	0.9906 [0.9862, 0.9950]	0.9865 [0.9799, 0.9932]	0.9170 [0.9000, 0.9340]	0.0282 [0.0225, 0.0339]	10.00	1.193
QUBO-Wrapper-k (k=10)	0.9900 [0.9860, 0.9940]	0.9843 [0.9786, 0.9900]	0.9105 [0.8891, 0.9319]	0.0326 [0.0266, 0.0387]	10.00	1.611
MI-top10	0.9900 [0.9864, 0.9935]	0.9852 [0.9806, 0.9897]	0.8981 [0.8733, 0.9230]	0.0375 [0.0308, 0.0443]	10.00	0.000
QUBO-Base	0.9900 [0.9864, 0.9935]	0.9852 [0.9806, 0.9897]	0.8981 [0.8733, 0.9230]	0.0375 [0.0308, 0.0443]	10.00	1.562
QUBO-Stability	0.9893 [0.9833, 0.9954]	0.9862 [0.9790, 0.9933]	0.9308 [0.9081, 0.9536]	0.0273 [0.0193, 0.0352]	10.00	1.648

TABLE III. PIMA DIABETES (REPEATED NESTED CV; $n = 15$ OUTER TESTS): MEAN METRICS WITH 95% CI

Method	AUC-ROC [CI]	AUC-PR [CI]	MCC [CI]	Brier [CI]	k (mean)	Sel. time (s)
PermImp-top6	0.8363 [0.8257, 0.8468]	0.7097 [0.6930, 0.7263]	0.4665 [0.4419, 0.4912]	0.1566 [0.1522, 0.1611]	6.00	0.000
QUBO-k (k=6)	0.8356 [0.8251, 0.8461]	0.7092 [0.6914, 0.7270]	0.4702 [0.4496, 0.4907]	0.1570 [0.1526, 0.1613]	6.00	0.324
RFECV	0.8350 [0.8238, 0.8461]	0.7123 [0.6933, 0.7313]	0.4613 [0.4366, 0.4860]	0.1570 [0.1524, 0.1616]	4.60	0.000
QUBO-Stability	0.8347 [0.8247, 0.8447]	0.7080 [0.6913, 0.7247]	0.4643 [0.4438, 0.4848]	0.1580 [0.1535, 0.1626]	6.00	0.333
L1-LogReg	0.8343 [0.8237, 0.8450]	0.7043 [0.6887, 0.7198]	0.4552 [0.4273, 0.4831]	0.1581 [0.1537, 0.1625]	6.53	0.000
MI-top6	0.8328 [0.8212, 0.8445]	0.7115 [0.6941, 0.7289]	0.4593 [0.4316, 0.4871]	0.1584 [0.1533, 0.1635]	6.00	0.000
QUBO-Base	0.8306 [0.8192, 0.8419]	0.7089 [0.6925, 0.7253]	0.4638 [0.4335, 0.4941]	0.1589 [0.1539, 0.1640]	6.00	0.360
QUBO-Wrapper-k (k=6)	0.8283 [0.8178, 0.8388]	0.7050 [0.6921, 0.7180]	0.4646 [0.4363, 0.4930]	0.1596 [0.1551, 0.1641]	6.00	0.368

TABLE IV. MEAN SELECTION STABILITY (JACCARD) ACROSS OUTER FOLDS

Method	Breast Cancer	Pima Diabetes
QUBO-Base	0.9998	0.7388
QUBO-k	0.5215	0.8721
QUBO-Stability	0.5796	0.7592
QUBO-Wrapper-k	0.6938	0.7748
MI-top$k$	0.9998	0.7456
L1-LogReg	0.8189	0.7912
RFECV	0.5210	0.7894
PermImp-top$k$	0.5952	0.8803

with strong baselines (RFECV and permutation-importance top-k) while guaranteeing exactly $k = 6$ features. This is operationally important when each feature corresponds to acquisition cost or measurement availability.

The performance-weighted relevance variant does not necessarily improve generalization, particularly in low-dimensional settings. A plausible explanation is that univariate relevance can mismatch multivariate subset utility, and inner-loop scoring can inject additional variance that does not translate into better held-out outer-fold performance, a phenomenon consistent with known wrapper sensitivities [10]. This suggests that performance-weighted signals may require stronger regularization or multivariate scoring to consistently outperform simpler baselines.

Stability is crucial when feature selection supports interpretation. Unstable selectors can yield brittle explanations and reduce trust even if AUC is high. On Pima, QUBO-k shows high Jaccard similarity, indicating reproducible subset selection under repeated outer folds. Nevertheless, stability must be interpreted alongside discrimination and calibration [8].

Calibration can diverge from AUC: two models may rank instances similarly (similar AUC) yet produce different probability quality (different Brier/reliability), which matters whenever decisions depend on absolute risk thresholds [14]. Evaluating calibration under the same leakage-safe protocol is therefore essential.

Limitations: The empirical evaluation is restricted to two low-dimensional tabular datasets ($d \leq 30$), one of which exhibits near-ceiling performance. This is sufficient to demonstrate leakage-safe evaluation, budget control, and stability diagnostics, but it does not support strong claims about scalability to high-dimensional regimes (e.g., genomics, text, or finance). Runtime values are descriptive end-to-end measurements and are not decomposed into solver versus non-solver components.

VI. CONCLUSION

This work introduced a stability-aware QUBO-based feature selection pipeline for tabular binary classification and evaluated it under a rigorous repeated nested cross-validation protocol, reporting confidence intervals, calibration-aware metrics, and explicit subset stability. Across the Breast Cancer Wisconsin and Pima Indians Diabetes benchmarks, QUBO formulations achieved discrimination competitive with strong classical baselines (L1-regularized logistic regression, RFECV, mutual-information filtering, and permutation-importance ranking) while additionally providing direct and transparent control over subset cardinality through an explicit k-constraint. Beyond raw accuracy, the results underscore that in near-separable settings where performance approaches a ceiling and differences become statistically and operationally marginal, reproducibility of the selected subsets and the ability to enforce interpretable, budgeted feature sets become the most meaningful differentiators. In this sense, QUBO offers a unified optimization framework to encode relevance–redundancy trade-offs, strict feature budgets, and stability-oriented regularization within a single objective. Future work will extend evaluation to higher-dimensional and cost-sensitive datasets, study solver and compute-budget sensitivity with detailed runtime breakdowns, and explore multi-objective QUBO designs that jointly optimize discrimination, calibration, and stability to further strengthen robustness and interpretability in practical decision-making pipelines.

ACKNOWLEDGMENT

The authors thank the Instituto de Investigación en Inteligencia Computacional y Ciencia de Datos (IIICCD) and the Universidad Nacional del Altiplano for supporting research activities and computational resources.

REFERENCES

[1] A. Lucas, "Ising formulations of many NP problems," *Frontiers in Physics*, vol. 2, p. 5, 2014, doi: 10.3389/fphy.2014.00005.

[2] M. Mücke, N. Piatkowski, and K. Morik, "Feature selection on quantum computers," *Quantum Machine Intelligence*, vol. 5, no. 1, p. 11, 2023, doi: 10.1007/s42484-023-00099-z.

[3] G. C. Cawley and N. L. C. Talbot, "On over-fitting in model selection and subsequent selection bias in performance evaluation," *Journal of Machine Learning Research*, vol. 11, pp. 2079–2107, 2010.

[4] S. Varma and R. Simon, "Bias in error estimation when using cross-validation for model selection," *BMC Bioinformatics*, vol. 7, no. 1, p. 91, 2006, doi: 10.1186/1471-2105-7-91.

[5] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005, doi: 10.1109/TPAMI.2005.159.

[6] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[7] N. Meinshausen and P. Bühlmann, "Stability selection," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 72, no. 4, pp. 417–473, 2010.

[8] S. Nogueira, K. Sechidis, and G. Brown, "On the stability of feature selection algorithms," *Journal of Machine Learning Research*, vol. 18, no. 174, pp. 1–54, 2018.

[9] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983, doi: 10.1126/science.220.4598.671.

[10] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1–2, pp. 273–324, 1997, doi: 10.1016/S0004-3702(97)00043-X.

[11] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine Learning*, vol. 46, no. 1–3, pp. 389–422, 2002, doi: 10.1023/A:1012487302797.

[12] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*, A. J. Smola, P. Bartlett, B. Schölkopf, and D. Schurmans, Eds. Cambridge, MA, USA: MIT Press, 1999, pp. 61–74.

[13] B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates," in *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, AB, Canada, 2002, pp. 694–699, doi: 10.1145/775047.775151.

[14] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly Weather Review*, vol. 78, no. 1, pp. 1–3, 1950, doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2.

[15] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, *et al.*, "Scikit-learn: Machine learning in Python," arXiv:1201.0490 [cs.LG], 2011.

[17] Scikit-learn Developers, "`sklearn.datasets.load_breast_cancer` documentation," Scikit-learn Stable Documentation, 2025. [Online]. Available: https://scikit-learn.org/

[18] J. W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R. S. Johannes, "Using the ADAP learning algorithm to forecast the onset of diabetes mellitus," in *Proceedings of the Symposium on Computer Applications and Medical Care*, Washington, DC, USA, 1988, pp. 261–265.

Attention-Guided Bidirectional Temporal Modelling with Graph-Based Regional Spatial Context for Bajra Crop Yield Prediction

Mamta Kumari[1]*, Suman[2], Devendra Prasad[3]
Deenbandhu Chhotu Ram University of Science and Technology, Sonepat, India[1,2]
PIET Samalkha, India[3]

***Abstract*—Bajra (pearl millet) is a very important crop in Rajasthan, India, since it is drought-resistant, nutritious, and culturally important. But its productivity is becoming vulnerable to changes in climate, such as erratic rain and temperature changes, and thus precise estimation of yield is vital. Crop Yield Prediction (CYP) indicators like soil decomposition, rainfall and meteorological patterns are slowly evolving, exhibiting long-term temporal dependency and propagating over time. Conventional cropping prediction algorithms based on artificial intelligence process the historical data and these indicators in a unidirectional manner. While mapping the temporal dependencies, these algorithms consider each year independently and do not capture the delayed effect, like salt degradation. To address this issue, the study proposes a region-based spatiotemporal model with an attention-guided Bidirectional LSTM (Long-Short Term Memory) framework for CYP, termed as G-BiLSTM. The proposed model reproduces the spatial relationships between districts via GCN (Graph Convolution Network) -based immediate neighbour extraction. Further, a Bidirectional LSTM is used to model multi-year CYP temporal features, allowing each annual observation to be encoded using both past and future temporal context. A variance-reduced and comprehensible representation is produced by integrating an attention mechanism to adaptively highlight the most informative years within a temporal window. Using 15 agro-environmental characteristics, including understudied elements like saline and alkaline soil composition, the framework is assessed on a dataset that includes 32 districts in Rajasthan over 13 years (2007–2019). The suggested attention-enhanced BiLSTM consistently outperforms traditional temporal models, achieving lower prediction error and better generalisation, according to experimental results analysis using a three-year sliding temporal window. For regional crop yield forecasting, the suggested method offers a scalable solution.**

Keywords*—Bajra crop yield prediction; regional crop yield forecasting; attention-guided Bidirectional LSTM; saline and alkaline soil composition

I. Introduction

It is anticipated that by the year 2100, the global population will hit 9.6 billion. The need for farm goods increases [1]. By the year 2100, India's population is projected to increase from 1.4 billion to 1.7 billion. Agriculture covers 1.6 million square kilometres (about half the area of India), and the agricultural land ranks second behind China [1]. The government must always ensure food to protect its inhabitants. Modern agriculture relies on crop yield prediction for better resource allocation, to decrease risks, and to ensure food security. However, it has become challenging with the complex aspects affecting Crop Yield Prediction (CYP). To enhance agricultural and rural statistics, researchers in the field are working on crop yield prediction methodologies and investigating ways to improve agricultural data collection for yield prediction. Considering the recent survey, yield is mainly dependent on data collected based on factors like climatic conditions, soil quality, and water quality [2-4]. However, the study in papers mainly differs based on regions, crop types, weather conditions, and irrigation techniques. Thus, there is no standard dataset that is fully accessible for agricultural study in a particular region [5]. Advancements in technology and changes in climate significantly influence agricultural production levels. The impact of weather on crop yield can be explored by considering time (measured in years) or time-related functions [6]. Various elements are expected to gradually increase yield over time, yet weather is recognised for its unpredictable impact. Therefore, to forecast agricultural outputs across different years or within the same year, a majority of research utilizes time-series algorithms that leverage past data and are regarded as more suitable than various machine learning and deep learning methods [7-9]. This study centres on the CYP of Bajra (pearl millet), which is widely grown in India, particularly in the states of Rajasthan, Maharashtra, Haryana, Uttar Pradesh, and Gujarat. The crop is capable of enduring conditions such as intense heat, lack of moisture, low soil fertility, and minimal rainfall. Bajra thrives in areas where other cereal crops struggle due to its specific growing needs. This study examines the precise CYP related to the Bajra crop in Rajasthan.

A. Related Work

There is an increasing body of literature that analyses machine learning and deep learning in predicting crop yield in different regions and crops. Among the studies, [7] optimises crop yield prediction in Rajasthan state for 10 districts and 7 crops using random forest (RF) and decision trees with gradient boosting. From 1997 to 2018, the research analysed acreage, productivity, yield, and rainfall. Gradient-boosting outperformed all other methods with 92.7% accuracy, 21.58 MAE, and 15.01 RMSE. Only a few of Rajasthan's traits and districts were studied. Computer-intensive gradient boosting is subject to noise. Overfitting can occur with too many decision trees. Therefore, many papers avoid these methods. The study [8] predicts Jaipur's agricultural productivity for five crops. The analysis focuses on the data spanning from 1991 to 2020. The variables that were measured included rainfall,

*Corresponding author.

hours of sunshine, temperature, and humidity levels. The writers employed Random Forest, Support Vector Machines, and Artificial Neural Networks.

The authors in [8] discovered that the Random Forest model achieved an accuracy rate of 92.3%, along with MAE and RMSE figures of 1.68 and 2.19, respectively. The study only covered one Rajasthan region and utilized a few criteria. The study [9] employs random forest, SVM, gradient descent, LSTM, and Lasso regression algorithms. The dataset comprises five crops for 33 districts in Rajasthan. All algorithms were examined, and the findings revealed that RF, SVM, and Lasso regression best-predicted yield. In the study [9], the limitations include the possibility of using a richer and larger dataset for each crop to enhance forecasts. An Improved Optimization Function (IOF) is proposed in [6] and implemented with the LSTM model to enhance prediction accuracy. Temperature, precipitation, humidity, solar radiance and various crop related data attributes were used in the study. The article [10] predicts mustard production in 5 Rajasthan districts using long-term weather and yield data. This study uses random forest, SVM, and ANN deep learning. Feature selection in [5] utilizes Principal Component Analysis (PCA) and Stepwise Multiple Linear Regression (SMLR). Assessment was conducted on agricultural area, soil type, temperature, rainfall, and climate. The research assesses six different model combinations and concludes that PCA-SVM stands out as the most effective yield prediction model. A hybrid model driven by data was created by the authors in [11] to accurately predict paddy yield through the use of Multiple Linear Regression (MLR) intercepts. The Artificial Neural Network (ANN) utilizing Feed forward Back Propagation is trained with characteristics like the area, count of open wells, tank volumes, canal lengths, and peak temperatures throughout the season examined in the study.

The creators of [12] devised a deep learning approach using a Recurrent Neural Network to enhance the Q-Learning reinforcement learning technique. The study predicts paddy yield in Vellore with an accuracy of 93.7% by utilizing properties of soil, weather, and groundwater, all while maintaining the original distribution of the data. The study in [13] forecasted the yield of sugarcane in Karnataka by utilizing Long Term-Time-Series (LTTS), weather and soil characteristics, Normalized Vegetation Index (NDVI), along with supervised machine learning (SML) techniques. Support vector regression (SVR) was employed by researchers. The scientist's analysed data related to climate and soil, along with satellite images from the years 2008 to 2018, to assess remote sensing information. A study based on empirical data focused on forecasting agricultural production by employing regression and time-series models to predict the crop known as 'Bajra' or pearl millet [14]. Models for prediction comprised ARIMA and ARIMAX, incorporating an external variable. When evaluating 'Bajra', the ARIMAX model outperformed the regression time series model. The study [13] proposed a regression model utilizing ANN and SVM to forecast Kharif crop yields based on rainfall information from Visakhapatnam. This research focuses on four types of crops: Maize, Ragi, Rice, and Bajra. This research employs a modular artificial neural network to predict monsoon precipitation, subsequently utilizing support vector regression to assess the primary Kharif crop yields in relation to rainfall and cultivated area. Ensembles of CNN and DNN machine learning predict corn production in the US corn belt, utilizing corn yield data from 1980 to 2019 [15]. Data on meteorology, soil, and management at the county level for 12 states in the US Corn Belt is provided. Constructed using deep neural networks and CNNs, ensembles are formed. This model accurately predicted maize grain yields for 2019, achieving an 8.5% relative root mean square error. The study in [16] introduces a framework utilizing deep learning for predicting agricultural output, employing CNNs and RNNs that rely on environmental data and management strategies. The CNN-RNN framework, in conjunction with RF, DFNN, and LASSO, predicts corn and soybean yields in the US Corn Belt utilizing data from 2016 to 2018. The new model surpassed all prior methods with an RMSE of 9% and 8% of usual yields. However, computationally expensive ANNs require careful design and hyper parameter adjustments. Too many trees can over fit and make random forests computationally expensive, and SVR is susceptible to outliers and requires careful kernel function selection [17]. With the rise of deep learning, advanced deep network topologies like CNNs [18] and RNNs [19-20] have been developed. Neural network-based agricultural production forecast models have increased due to deep learning. Out of the surveyed literature, 69% of papers employed deep learning approaches, with just 20% using a CNN-based model. There have been very few or negligible studies that use the graph-based model to harness the spatial features from the crop data for CYP. Like in [21], the Graph Convolution Network (GCN) is used for learning the weed plant features from labeled and unlabeled images, which are already extracted from CNN. The authors also work towards using limited labeled data and learning from graph structure to improve recognition accuracy. However, this study does not directly relate to predicting the yield.

The overall survey for dataset attributes and characteristics used for CYP in the studies is majorly divided into meteorological [2], soil [3], phenotype [22], fertilizer, diseases, and pests [4] related attributes, as well as other attributes, as shown in Table I. All of these characteristics are significant for determining crop output. As a result, in the present manuscript, a summary is created of the data attributes utilized in earlier work to determine which attributes are least studied and which ones need to be worked on. It is observed from Table I, that phenotypic, irrigation accessibility, fertilization, and undesirable features (diseases, weeds, and pests) are the least investigated qualities (3%, 3%,3%, and 14%). However, all these characteristics are investigated within the Precise Agriculture application, which makes use of IoT [3]. But, because the work focuses on data specifically for yield prediction, the study will select the other attributes of the yield prediction process. The other attribute that is highly used for CYP is soil characteristics. The sub-categories in Table I, demonstrate that groundwater levels and soil moisture characteristics are also understudied, but the current study included rainfall because it directly affects yield [23]. Red, black, desert, clay, or a mix of soil types has been understudied, and data is collected in only a few studies; therefore, soil type is considered in the present study. The kind of soil determines which plants thrive and which struggle. Cotton, for example, thrives in dark soil [24]. Also, infertile or barren soil, which includes excessive salt levels, is not considered. This study also assessed saline and alkaline soil in numerous state districts. The research

TABLE I. SUMMARY OF ATTRIBUTES FROM PREVIOUS STUDIES

Attributes Category (%)	Attributes	Attributes Category (%)	Attributes
Meteorological (40%)	Rainfall	Phenotype (3%)	Lodging rate
	Temperature		Inversion Rate
	Humidity		Plant Height
	Sunshine duration		Ear Height
	Evapotranspiration		Empty Stalk Rate
	Dew points temp	Undesired Traits (14%)	Weed Determination
	Wind speed		Pest
Fertilizer (7%)	Fertilizer Strength		Disease
Area (3%)	Area under Cultivation	Soil (23%)	Groundwater
			Soil Water
			Soil Moisture
Irrigation Accessibilities (3%)	No. of Tanks		Soil Temperatures
	Canal Length		Nutrients
	Open Wells		Soil pH

also evaluated the local climate. The local climate impacts agricultural growth. Some crops thrive in arid climates, while others favour humidity [25].

Numerous works are conducted in the field of temporal modelling; the model in [26] uses an enhanced Gate Recurrent Unit through an LSTM model for the prediction of CYP. However, spatial connections are considered static by implementing a Grey Wolf Optimisation (GWO). Similar to the proposed model, the authors in [27] have proposed an attention mechanism for a CNN that is a convolutional neural network-based BiLSTM model for CYP. However, different from the proposed model, the spatial connections are considered independent through a convolutional neural network. The proposed model focuses towards a unified mapping of the spatial connection among districts affecting yield parameters through the GCN model and maps the enhanced bidirectional temporal dependencies. An enhanced deep learning model is presented in [28] through a Bayesian optimised CNN-BiLSTM model similar to the model in [26]. However, the parameters are considered independent, and the delayed effect of rainfall is not considered in [27-28]. Similar to the proposed model in the study and in [26], the authors in [29] applied channel attention to BiLSTM for CYP on image-based data. Moreover, unlike existing temporal models, the proposed model treats the spatial representations dependant as a graph and joint modelling of these with long-term bidirectional temporal dependencies, contributes to regional-aware temporal modelling.

Table I shows that phenotype characteristics, irrigation accessibility, fertilisation, and undesirable features (such as diseases, weeds, and pests) are the least studied attributes, with percentages of 3%, 3%,7%, and 14%, respectively. These attributes are integral to Precision Agriculture applications leveraging IoT [3]. However, focusing specifically on yield prediction, this study will emphasise other relevant attributes in the prediction process.

Soil characteristics are another key attribute widely utilised in CYP. Table I also indicates that groundwater levels and soil moisture characteristics are areas with limited research. However, the current study incorporates rainfall due to its direct impact on crop yield [23]. Soil types such as red, black, desert, clay, or a mixed soil have received less attention in data collection across studies, making soil type a focal point in this study. Soil type plays a critical role in determining which crops flourish and which struggle; for instance, cotton thrives in dark soil [24]. Furthermore, infertile or barren soils, including those with excessive salt levels, are not considered. This study also examines saline and alkaline soils across various state districts. Additionally, local climate conditions are evaluated, as they significantly influence agricultural productivity. Some crops thrive in arid climates, while others prefer humid environments [25]. These deep learning models have become increasingly popular in agricultural production forecasting. According to recent surveys, 69% of studies have adopted deep learning techniques, with CNN-based models being utilised in only 20% of cases.

1) Challenges in the existing studies need to be addressed: Existing studies in CYP focus on using data (images, new features, and focused data for yield prediction), spatial and temporal modelling of the data. However, the existing studies exhibit challenges that need to be addressed. These are discussed through a Table II, mapping the challenges on how the contribution of the study is connected.

B. Contributions of the Paper

As discussed in Table I, the study differs and addresses the challenges identified in existing studies. However, the main contributions of the study are:

- The suggested model performs CYP for Bajra (pearl millet) over 13 years (2007–2019) in 32 districts in the Indian state of Rajasthan. By adding a wide range of agro-environmental variables, such as weather patterns and underutilised soil characteristics, the study considerably broadens the scope of data collection, building on previous research.
- The proposed model suggests a three-phase framework for spatiotemporal learning that includes: 1) Data Collection, preprocessing and capturing Annual Regional-immediate neighbour information; 2) Joint Spatio-temporal feature extraction through attention-guided G-BiLSTM; and 3) yield prediction.
- The suggested framework uses an attention-guided Bidirectional LSTM (BiLSTM), in contrast to traditional temporal models, to encode each year from CYP data representation using both past accumulation and future-aware agro-environmental context, and the attention mechanism selectively highlights the most significant temporal points, resulting in a spatiotemporal embedding that is interpretable and reduces variance. The proposed framework minimises variance in the predictions across various regions or temporal windows by down-weighting uninformative years, which prevents the model from reacting excessively to inflated shifts in the training data (such as an anomalous rainfall year).
- Evaluation through RMSE, R2 score, correlation coefficient, and other common performance metrics is conducted to assess the suggested Attention-guided G-BiLSTM framework.

Study Organisation: the introduction, related work and contributions are explained in Section I, the detailed proposed

TABLE II. CONNECTING THE CHALLENGES WITH THE CONTRIBUTIONS OF THE PAPER

Aspect	Challenges in the Existing Studies	Contribution of the Paper to Address the Challenges
Data Reliability and Underutilization of Soil-related Attribution in Data for CYP	Soil attributes were often understudied or included in only a few analyses. Given the significant influence of soil quality on crop growth and productivity, overlooking these factors undermines the robustness and accuracy of yield prediction models [7], [8], [9], [10], [3], [24].	The paper introduces saline and alkaline soil parameters collected for Rajasthan through an open-source government website.
Regional Modelling of Districts in the Dataset	Existing studies like [7], [8], [10], [13], [15], [16], [17], [18], [19], [23], [24], [25] treat the annual yield data of regions independently, ignoring the temporal and changing dependency of features like rainfall and soil nutrients.	The study reproduces the spatial relationships between districts via GCN (Graph Convolution Network)-based immediate neighbour extraction. Different from models in [21], [24], [29], where GCN and Graphical Neural Network (GNN) models are used, the proposed model in the paper performs joint spatio-temporal modelling. This is done by considering year-wise embedding learnt through graphical deep learning and directly mapping it for temporal modelling.
Joint Spatio-temporal Dependency Handling	Existing temporal models like RNN, LSTM [9], [10], [11] do not account for delayed and cumulative agro-environmental effects of rainfall and such parameters because they process historical data in a unidirectional manner.	Different from models in [21], [24], [29], where GCN and Graphical Neural Network (GNN) models are used, the proposed model in the paper performs joint spatio-temporal modelling. This is done by considering year-wise embedding learnt through graphical deep learning and directly mapping it for temporal modelling. The proposed model introduces an attention mechanism that highlights the most influential temporal points from the joint future-aware and past information of the CYP data.

model is explained in Section II. The results and experimentation are explained in Section III. Section IV presents the conclusion and Section V details the limitations and future scope of the study.

II. PROPOSED METHODOLOGY

In this section, the overall flow of the proposed model is explained. The contributions stated in Section I are detailed and achieved in this section through Data Collection, Preprocessing, and capturing Annual Regional-immediate neighbor information given in Section II B and finally the Joint Spatio-temporal feature extraction through attention-guided G-BiLSTM for Yield Prediction in Section II C. However, the motivation of deriving the contributions and formulation of the study is explained in Section II A.

A. Motivation and Proposed Formulation of the Study

Let's represent the attributes/characteristics/features of each district as $x_{d,t}$ in the final data and the actual crop output as $y_{d,t}$, where d, t represent the district and the year, respectively. Here $x_{d,t} = \{x^w_{d,t}, x^s_{d,t}, \ x^l_{d,t}\}$ where meteorological attributes (annual rainfall, climate type) are represented as $x^w_{d,t}$, soil attributes (soil type 1, soil type 2, saline soil, alkaline soil, phosphorus, potassium) $x^s_{d,t}$, and yield-related attributes (area, production, yield) as $x^l_{d,t}$. From these features, yield (area, production, yield, soil characteristics) and meteorological attributes (annual rainfall) change both spatially and temporally. Whereas, soil attributes (like soil type-1, soil type-2), and climate type, remain stable over time, district-wise. Attributes like $x^w_{d,t}$ varies regionally and temporally, whereas attributes in $x^s_{d,t}$ are some of the slowly changing factors that affect yield in real-world agricultural datasets. Yield prediction is a challenging spatiotemporal learning problem due to its long-term period relationships and the delayed consequences [30][31]. Each district in the dataset considered in this study is represented by a series of 13-year annual observations that include a wide range of agro-environmental characteristics $x_{d,t}$. Understanding how these characteristics build up, interact, and spread over time to affect yield in subsequent years is the real-challenge to be addressed.

Deep learning models like LSTM, RNN, and CNN may assume independence among the districts even though attributes like soil type 1, soil type 2, and climate type remain stable over time, district-wise, are different, and while processing these models, they may not fully use the spatial structure of the original data. To address this issue, the goal of the study is to employ a graph-based model to extract the immediate neighbour information of the districts ($A_{d,t}$) over the years $h_{d,t}$ which is leveraged by the spatio-temporal model for historical time series yield prediction. This process is illustrated in Fig. 3. Hence, the task of CYP is formulated as $y_{d,t}$ with $x_{d,t}$ where $t = \Delta t$ and Δt is taken as 3 years (that is, training the proposed model over 3 years of historical data). The main aim of the proposed model is to address these issues through:

$$y_{d,t}(t+1) = \text{Prediction}\Big(\text{GBiLSTM}(\alpha\,\text{BiLSTM}(\text{GCN}(x_{d,t}, A_{d,t})))\Big) \quad (1)$$

In Eq. (1), the prediction is based on the MSE loss function. In the first stage the Annual Regional-immediate neighbour, information is learnt through CYP features and $A_{d,t}$ which carries the regional information. This data which is year wise, is learnt through BiLSTM model to get the spatiotemporal embedding. Further, these are learnt over attention model to highlight the most significant temporal points towards yeild prediction. In the upcoming sections, 1) Data Collection, preprocessing and capturing Annual Regional-immediate neighbour information; 2) Joint Spatio-temporal feature extraction through attention-guided G-BiLSTM; and 3) yield prediction are detailed. The overall flow of the proposed model is illustrated in Fig. 1.

B. Data Collection, Preprocessing, and Capturing Annual Regional-Immediate Neighbour Information

The proposed model is evaluated on the primary regions for Bajra cultivation in India, specifically Rajasthan. Among the regions like Maharashtra, Gujarat, Haryana, and Uttar Pradesh, Rajasthan leads as the foremost state for Bajra production.

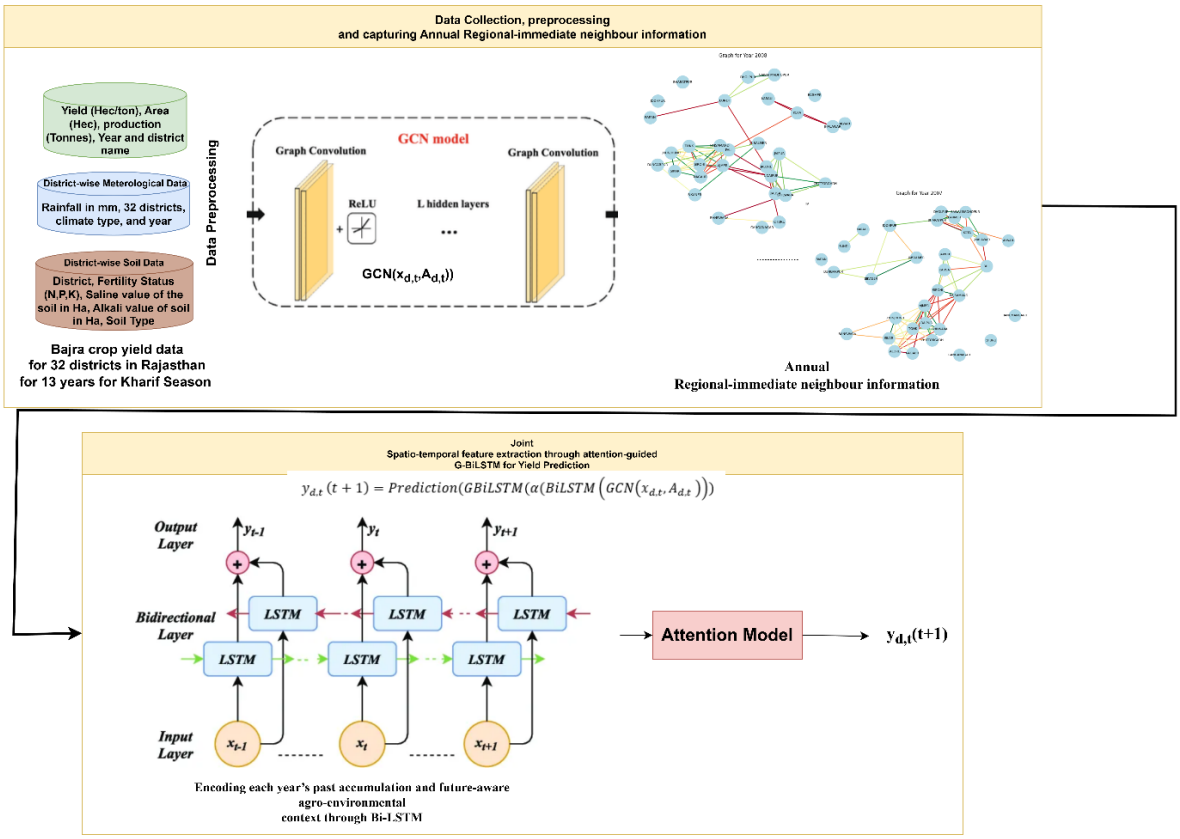


Fig. 1. Overall flow of the proposed model.

1) Geographical study of Rajasthan: In Rajasthan, the weather poses difficulties for growing various crops, particularly cereals. However, due to its resilience to harsh climatic conditions, millet farming has taken up a significant portion of the land in the region. Rajasthan accounts for 56.23% (4.43 million hectares) of the entire area dedicated to Bajra, contributing 41.40% (3.80 million tons) of India's total output. This fulfils the essential food needs of the state as well. From 2016 to 2017, Alwar district recorded the peak Bajra yield at 2149 kg/ha, surpassing all other Bajra-producing districts in Rajasthan. Traditionally, farmers depend on techniques like specific seed types, irrigation, and chemical fertilisers to sustain soil fertility and agricultural productivity. Nonetheless, these conventional methods are increasingly unreliable because of their vulnerability to the effects of climate change, further exacerbated by alterations in soil quality [14]. Therefore, examining the yield is crucial, considering elements such as soil characteristics and climatic conditions. This study collects data on soil and weather, which have influenced the Bajra crop in Rajasthan over the years, to effectively predict the yield. This study aims to analyse the yield fluctuations of Bajra and forecast its performance across the 32 districts of Rajasthan, as illustrated in Fig. 2.

This study focuses on capturing the yield variability of Bajra, as well as its prediction in Rajasthan and its 32 districts, as shown in Fig, 2.

2) Regional data collection for Rajasthan: To support the proposed approach, we curated a novel dataset that com-



Fig. 2. Geographical area of the study.

bines district-level soil properties, nutrient concentrations, and weather indicators over multiple cropping seasons. Data were sourced from verified government repositories and harmonised to ensure temporal and spatial consistency. This rich dataset serves as the foundation for the spatial modelling process and enables more accurate learning of Bajra yield patterns than the limited datasets used in prior research. In order to tackle the issues outlined in Section I, the process of data collection is aimed at obtaining thorough meteorological and soil information. This includes meteorological factors such as rainfall and climate patterns, particularly during the Kharif season, which is the primary growing season for crops like Bajra in regions across Asia and Africa. Additionally, soil data is being collected, including levels of potassium and phosphorus, as well as saline and alkaline content in the soil. Moreover, soil types and their sub-categories are being categorised to capture the diverse soil compositions in the study areas [35]. The main goal of gathering data on soil and weather is to comprehend the complex connection between these environmental elements and agricultural results. For example, changes in precipitation have a direct effect on the levels of water in the soil, which subsequently affects soil characteristics like the concentrations of nutrients and salts. By comprehensively analyzing soil and weather data, insights are aimed at being gained into how these factors interact and affect crop growth and yield. A more profound comprehension will facilitate the creation of precise predictive models and offer essential insights for agricultural choices, ultimately aiding farmers in adjusting to evolving environmental circumstances and enhancing crop yield. The data collection process involved sourcing various attributes from official Rajasthan state government websites.

3) Description of data sources: Firstly, the yield data is retrieved from the DACNET website of the Ministry of Agriculture and Farmers' Welfare [33]. This yields data for 32 districts of Rajasthan, over 23 years from 1997. The attributes collected are yield, area, production, year, and district name. To collect information about meteorological data, annual rainfall over the year [32] and climate type are collected [31]. Annual rainfall data is obtained from the Department of Water Resources, Rajasthan Government's official website [32]. This data was distributed across multiple PDF files, with each file containing daily rainfall records for different districts over the years. The data is present in the form of PDF (Portable Document Format) files, with different files for each year and each file containing daily rainfall for each district. Each one of these files had to be parsed in Python using libraries like PyPDF2 and regex. Whereas the climate data for yield prediction for each district in Rajasthan is obtained from [31]. According to [31], Rajasthan's territory is categorised into five climate types: arid, semi-arid, semi-humid, humid, and Mediterranean. Each of the 32 districts was assigned its respective climate classification based on this categorisation. This is achieved using regex functions, where each district is assigned the respective climate.

Now, the district-wise soil characteristics, including sodium, phosphorus, and potassium levels, as well as the extent of problematic soils such as saline and alkaline soils, were obtained from the Rajasthan Agriculture Statistics at a Glance report available on the official Rajasthan Government website [34]. However, a challenge arose as this data was presented in image formats. To overcome this, Microsoft's Optical Character Recognition (OCR) tool was utilized to extract the data from images and store it in CSV files. Additionally, soil type data was sourced from another reference [35], which provided information on 16 different categories of soil [35]. Each region within Rajasthan was found to have soils falling into two distinct categories. Finally, the final data obtained will consist of meteorological data, soil-related data, and yield data, which is 12 attributes for 32 districts over 13 years. The data is shown in Table III, and the overall steps are given as follows:

TABLE III. AGRO-ENVIRONMENTAL ATTRIBUTES DESCRIPTION

Column Name	Description
District	Names of the various districts of Rajasthan used in the present study
Year	The year for which the data was collected
Area (Hectare)	Total area in hectares of the district under cultivation for that particular year
Production (Tons)	Total production in tons of Bajra for that particular year
Yield (Tons/Hectare)	Amount of production in tons per unit hectare of area under cultivation for a particular year
Soil Type 1	One type of soil found in that district of Rajasthan
Soil Type 2	Another type of soil found in that district of Rajasthan
Phosphorous	Amount of phosphorous content in the soil for the given district
Potassium	Amount of potassium content in the soil for the given district
Saline Soil (Ha)	Total area of saline soil in hectares
Sodic/Alkaline Soil (Ha)	Total area of alkaline soil in hectares
Annual Rainfall (mm)	Rainfall for the given district in mm each year
Climate Type	The climate type of each district of Rajasthan like arid, humid, wet, etc.

- Yield Data Retrieval (DACNET Website): Attributes collected: yield, area, production, year, and district name for 23 years from 1997 [33].
- Then the annual rainfall data is collected from [32], and for each year, this is parsed and collected from PDF files using PyPDF2 and regex. Then it is incorporated with yield data, which leaves data for 13 years from 2007 to 2019 for 32 districts.
- Further, the climate type data is recognized with the data collected until step 2, where each district is explained with a certain type of climate as per [31]. Here, using regex, the climate type is added to the climate type column for each district in the data collected till now.
- Now, the data for soil type from [35] is added district-wise to the data collected until step 3. Then the district-wise low, medium, and high labels of Sodium, Phosphorus and Potassium in soil are merged with the data from [34]. This data was earlier present in the form of images, from which the data is extracted using OCR tools and merged with the data till now.
- The final data will have 12 attributes such as annual rainfall, district name, year, yield, area, production, soil type 1, soil type 2, phosphorus, potassium, saline soil, and alkaline soil, as given in Table III.

4) Regional data pre-processing: Initially, the data pre-processing involves verifying the presence of null values, and upon loading the dataset, it is observed that no null values

TABLE IV. DATA TYPE OF THE AGRO-ENVIRONMENTAL ATTRIBUTES IN THE CYP DATASET.

Attribute	Data Type
District	object
Year	object
Area (Hectare)	int64
Production (Tons)	int64
Yield (Tons/Hectare)	float64
Soil Type 1	object
Soil Type 2	object
Phosphorus	object
Potassium	object
Saline Soil (Ha)	float64
Alkaline Soil (Ha)	float64
Annual Normal Rainfall (mm)	float64
Climate Type	object

exist. The second step is to check for the object type of attributes and encode them. As per Table IV, a label encoder from the sklearn library is applied for the attributes with object type.

Some attributes labeled as "unnamed: 0", in the dataset, which contained serial numbers, were dropped during processing. Additionally, certain attribute columns such as potassium, climate type, saline and alkaline soil were named as unnamed and subsequently renamed to appropriate names. For instance, "unnamed: 8" representing potassium was renamed using Panda's library and regex expressions. The attribute "year" contained value in the format "2007-2008, 2008-2009." To standardize these values, a new list of individual years like ["2007", "2008" ...'"2019"] was created and merged with the existing dataset. Following this pre-processing, the initial step involves constructing graphs based on the years. Further details on these processes will be provided in the subsequent section.

The statistical summary of the attributes considered in this study is given in Section II. Rainfall is the attribute that is related to the climatic/meteorological type of data, as shown in Table IV, along with the climate type in Rajasthan for the 32 districts. While the climatic conditions affect the yield of the Bajra crop, soil attributes will also help framers take actions to improve yield. In Rajasthan, the rainfall varies from 0.82mm to 8734mm with a remarkable deviation from average, as seen in Table IV. This explains the change in climate from humid to semi-humid, which in turn affects the yield of Bajra [31]. Most of the districts in Rajasthan experience arid and humid climates [31]. The saline values in the soil also vary from 21.64 to 56369493. High saline levels in salts will affect the crops' ability to hold water content and remain hydrated. However, poor climatic conditions affect the soil conditions and lead to less yield. Thus, these conditions force farmers to use pesticides to boost the yield. Hence, a systematic process is carried out to collect the data from the respective sources.

Considering the preprocessed agro-environmental attributes data, the Annual Regional-immediate neighbour information is learnt as $GCN(x_{d,t}, A_{d,t})$, as in Eq. (1). The scenario of information considering the preprocessed agro-environmental attributes data, the Annual Regional-immediate neighbour information is learnt as regional information in $x_{d,t}$, which is explained through Fig. 3 . It displays the graph for the year 2008, and the agro-environmental attribute similarity across different districts/regions in Rajasthan.

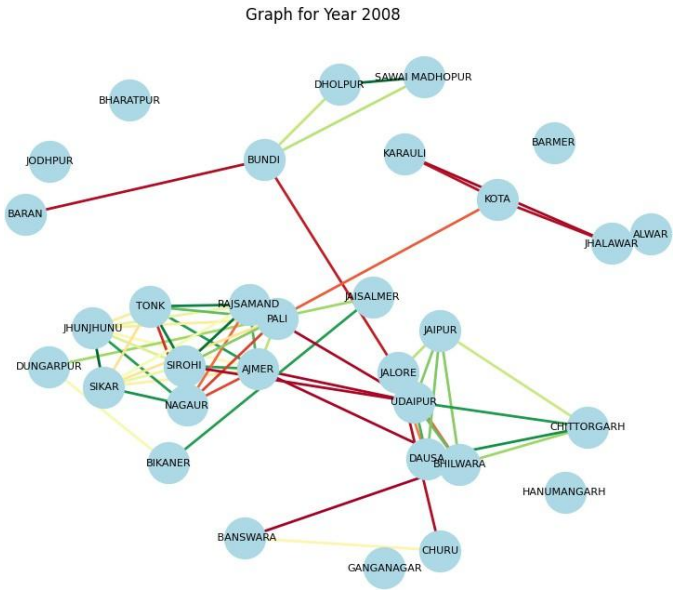


Fig. 3. Annual regional information as graph for the year 2008.

The lack of an edge linking Jodhpur and Bharatpur indicates a significant difference in the attributes described in Table II. This intriguing observation suggests that Jodhpur and Bharatpur differ from each other in certain ways. The striking differences, such as the lack of boundaries between these districts, highlight the importance of some characteristics and may indicate the distinct soil and climate types affecting their productivity. Such Regional immediate information is captured annually based on the data.

The scenario explained as similarity among the regions is captured as $A_{d,t}$. This is captured using Immediate Neighbor Extraction strategy. Let the spatial adjacency matrix $A_{d,t} \in \mathbb{R}^{N \times N}$, N is number of samples, it is be defined as:

$$A_{d,t} = \begin{cases} 1 & \text{if districts } d_i \text{ and } d_j \text{ share a boundary at year t} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

For Eq. (2), we consider the property of cosine similarity that yields of neighboring districts that often exhibit correlated variations due to shared rainfall patterns and soil structures.

For each annual-wise graph, as shown in Fig. 3, the graph's edges are converted into edge indices. This results in a tensor of shape [2, num_{edges}], where num_{edges} is the number of the graph's edges. This tensor's columns signify edges, where the initial row holds the indices of source nodes and the subsequent row features the indices of target nodes. Therefore, this tensor acts as a compact representation of the $A_{d,t}$ in Eq. (3). The propagation rule for a l-layer GCN is followed here as per in [20]:

$$\begin{aligned} GCN(x_{d,t}, A_{d,t}) &= H^{(l+1)} \\ &= \sigma\Big(D^{-\frac{1}{2}} A_{d,t} D^{-\frac{1}{2}} H^{(l)} W^{(l)}\, x_{d,t}\Big) \end{aligned} \tag{3}$$

where, $H^{(l)}$ represents the Annual Regional-immediate Neighbour Information at layer l , $A = R + I$ is the adjacency matrix with self-loops added, as shown in [20] and the notations are similar to those in [20].

The final output of the GCN model after passing through all l-layers, explained in Eq. (3), is stored in gcn_output. This output represents the node embeddings or features learned by the GCN model for the given input graph. Each row of gcn_output corresponds to the feature representation of a node (district), and each column represents a feature dimension. $\mathbf{D}$ is the degree matrix of $A_{d,t}$, $\mathbf{W}^{(l)}$ denotes the learnable weight matrix at layer l, and σ is the activation function, ReLU. Further, the embedding's array is generated after processing of GCN. Thus, this is substituted in Eq. (1), where this is processed through BiLSTM model. This process is detailed in next section.

C. Joint Spatio-Temporal Feature Extraction Through Attention-Guided G-BiLSTM for Yield Prediction

Joint spatio-temporal is explained in Eq. (1), for each district d the Annual Regional-immediate Neighbour Information enriched embeddings $H^{(l+1)}$ is learnt for each year t that is, $H_t^{(l+1)}$ is fed into the BiLSTM.

The base LSTM units is employed here from [6], this unit maintain a memory cell c_t and hidden state h_t, defined by the classical LSTM equations. The forget, input, and output gates are [see Eq. (4)]:

$$\begin{aligned} f_t &= \sigma\left(W_f H_t^{(l+1)} + U_f h_{t-1} + b_f\right) \\ i_t &= \sigma\left(W_i H_t^{(l+1)} + U_i h_{t-1} + b_i\right) \\ o_t &= \sigma\left(W_o H_t^{(l+1)} + U_o h_{t-1} + b_o\right) \end{aligned} \tag{4}$$

whereas, here the goal is to encode each year's past accumulation and future-aware agro-environmental context. This is achieved by updating the candidate cell state as Eq. (5):

$$\widetilde{c}_t = tanh\left(W_c H_t^{(l+1)} + U_c h_{t-1} + b_c\right) \tag{5}$$

and the actual cell state updates as Eq. (6):

$$c_t = f_t \odot c_{t-1} + i_t \odot \widetilde{c}_t \tag{6}$$

The hidden state is Eq. (7):

$$h_t = o_t \odot tanh\left(c_t\right) \tag{7}$$

BiLSTM is employed here to extend this by processing the year's past accumulation and future-aware agro-environmental context. If the forward states are $h_t^{(f)}$ and backward states $h_t^{(b)}$, then the combined representation is Eq. (8):

$$H_t = \left[h_t^{(f)} \| h_t^{(b)}\right] \tag{8}$$

Further, to get more comprehensive spatio-temporal embeddings, the attention mechanism is introduced here highlighting the most influential seasonal time points.

Given the sequence output $\{H_1, \ldots, H_T\}$, attention weights α_t as given in Eq. (1) is captured as Eq.(9):

$$\begin{aligned} e_t &= v^\top tanh\left(W_a H_t + b_a\right) \\ \alpha_t &= \frac{exp(e_t)}{\sum_{\tau=1}^{T} exp(e_\tau)} \end{aligned} \tag{9}$$

The context vector becomes a weighted sum of hidden states, as in Eq. (10):

$$c = \sum\nolimits_{t=1}^{T} \alpha_t H_t \tag{10}$$

This context vector encodes the spatio-temporal essence of each district's agricultural dynamics [36][37]. Thus, the final yield prediction is captured through substituting the information in Eq. (1). Thus, the overall process detailed as a pseudocode and the proposed model is given in Algorithm 1.

Algorithm 1 Input: Agro-environmental Attributes, model parameters (hidden size, learning rate, epochs), $x_{d,t}$ (features in dataset), N (number of samples) Output: predicted yield $y_{d,t}$

Input: Agro-environmental Attributes, model parameters (hidden size, learning rate, epochs), $x_{d,t}$ (features in dataset), N (number of samples)
Output: predicted yield $y_{d,t}$
for all $d \in \{1, \ldots, 9\}$ **do**
  for all $t \in \{1, \ldots, 12\}$ **do**
    for all $N \in \{1, \ldots, 32\}$ **do**
      a. Initialize $GCN\left(x_{d,t}, A_{d,t}\right)$ #Section 2.2.3
      b. Calculate similarity among the regions captured as $A_{d,t}$ as per Eq. (2) to get the Immediate Neighbor Extraction.
      c. Construct the Annual Regional information as graph using $A_{d,t}$ # Fig. 2
    end for
  end for
end for
for all $epoches \in \{1, \ldots, 500\}$ **do**
  To capture the Annual Regional-immediate Neighbour Information at layer l by training the GCN model using Eq. (3)
end for
Initialize the attention-guided G-BiLSTM for Yield Prediction for Joint Spatio-temporal feature extraction. # Section II C
for all $epoches \in \{1, \ldots, 60\}$ **do**
  a. Input the embeddings from step 4f
  b. Encode each year's past accumulation and future-aware agro-environmental context by updating the candidate cell as shown in Eq. (5).
  c. Generate the yield prediction of the crop as output ($Y_{d,t}$) from Eq. (1).
end for

TABLE V. SIMULATION PARAMETERS FOR PROPOSED MODEL

Model Name	Parameters	Value
GCN (GCN Conv)	Number of Hidden Channels	32
	Dropout Probability	0.3
	Weight Decay	1.00E-04
	Learning Rate	0.01
	Number of Epochs	500
	Input Size	32
	Hidden Size	64
BiLSTM Model with LSTM Set As	Output Size	32
	Learning Rate	0.001
	Number of Epochs	60
	Window Size	3

III. RESULTS AND DISCUSSION

The proposed G-BiLSTM model was evaluated to assess its ability to predict the yield of Bajra in 32 districts and across 12 multi-year agricultural cycles.

A. Experiment Scenario and Settings

The experiment involved the three combined sets of data including soil nutrient, soil type profiles and high-resolution weather records that were utilized in 12 years, given in Section II A. The train-validation-test split was on five randomized folds to make it robust. The given section is aimed at providing the quantitative outcomes, statistical analysis of significance, error distributions, performance tables, and comparative interpretation to make it clear that the given method is superior to current work.This part covers the evaluation of the proposed model's training and testing, alongside proposed model (excluding soil data), baseline LSTM from [9], baseline RNN model from [38], and CNN-RNN from [16]. The suggested model's performance is evaluated against these baseline models, where the GCN is trained using a cross entropy loss function [20], and LSTM is utilised for the final prediction. Parameters like the learning rate and the count of hidden layers are adjusted to enhance RMSE, R2 score, and correlation coefficient, which act as metrics for assessing model performance. These calculations originate from the libraries Sklearn and SciPy.

The proposed model is implemented on the Google Colab platform using Python and its libraries. The parameters set are given in Table V, where GCN provides the learned spatial embedding's for span of 13 years for the 32 districts. Graph construction utilizes the NetworkX library, the GCN employs torch-geometric package and the BiLSTM model is built using PyTorch.

The model is evaluated based on, RMSE, R^2 and correlation coefficients are detailed as, RMSE is a frequently used metrics in [7,9] to assess how well regression models function, RMSE is defined as Eq. (11):

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2} \tag{11}$$

R^2 is another metric for evaluating a regression model's quality of fit is the coefficient of determination. It calculates the percentage of the dependent variable's variation that can be predicted based on the independent variables (features) (goal (yield).R^2ranges from 0 to 1, where 1 indicates the perfect fit. Mathematically, $R^2$is calculated as Eq. (12):

$$R^2 = 1 - \frac{SSRes}{SSTot} \tag{12}$$

where, $SSRes$ and $SSTot$ is the overall sum of squares and the sum of squares of the residuals, respectively. R^2 provides insight into how effectively the model accounts for data variability. The total sum of squares indicates the overall variability in the observed yields, whereas residuals show the discrepancies between the observed (actual) yields and the yields predicted by the regression model. A higher R^2 value suggests that the model fits the data better since it accounts for a greater percentage of the variability in yields.

The degree and direction of the linear link between two variables are measured by the correlation coefficient, which is frequently represented by the symbol "Pearson". Regarding yield forecasting, the correlation coefficient measures how well the predicted yields from the proposed model align with the actual observed yields. Mathematically, it is given as Eq. (13):

$$pearsonr = \frac{\sum (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum (x_i - \overline{x})^2 \ \sum (y_i - \overline{y})^2}} \tag{13}$$

where, x_i and y_i represent the two variables' individual data points, and $\overline{x}$ and $\overline{y}$ represent their respective means. These metrics are computed for each of the four models in this study for both training and testing.

The errors in prediction were examined in the areas that received rainfall extensively, those experiencing droughts and those that were a combination of the two. The given structure was particularly strong when it comes to the capture of deviations of monsoons. Mathematically, it is explained by the attention mechanism which places substantial weights on sudden changes in rainfalls:

$$\alpha_t = \frac{exp(e_t)}{\sum e_\tau}, e_t \propto tanh\,(W_a H_t),$$

resulting in high sensitivity to time steps, where climatic volatility is highest. Only in extreme anomaly years when the rainfall varied more than 40% of the 15-year mean, error peaks were realized but the model still provided an RMSE of less than 60, which was better than LSTM and GNN models.

B. Analysis of the Proposed Model

When the model is evaluated for heatmap representations of Agro-environmental Attributes in Fig. 4, it is seen that the attributes related to meteorological type are mostly used for yield prediction. To understand the effect of the attributes in the dataset considered here over the yield of Bajra Pearson's correlation is conducted, and the heatmap is illustrated in Fig. 4. Before conducting this, the data is encoded into numbers using the label encoder library from the Sklearn package. Further, the Pearson correlation is also imported from Sklearn. The colour legend in the heatmap shows the high and low correlation between the attributes and yield based on the values varying

from -1 to 1. The analysis of the heatmap reveals notable patterns in the correlation between various attributes and the yield of the Bajra crop. Specifically, soil attributes, such as soil type 2, exhibit a high positive correlation with the yield attribute. Rainfall attribute also shows positive trend with yield, the values exhibits variations along the year as seen in Fig. 5.

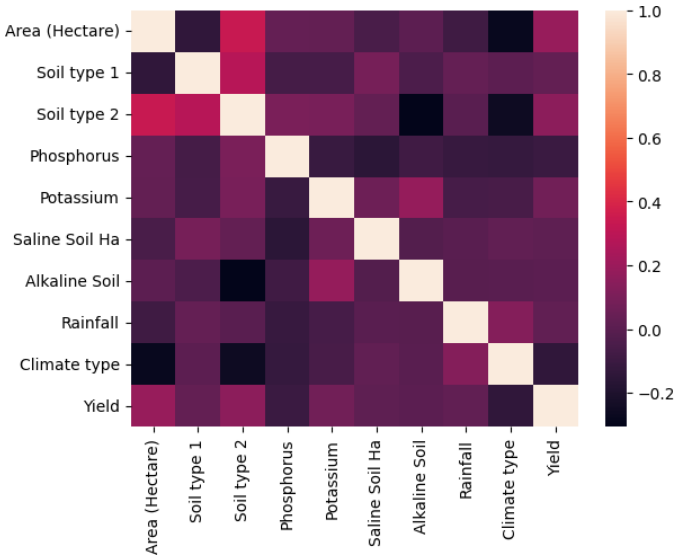


Fig. 4. Heatmap representation of the correlation matrix of the agro-environmental attributes.

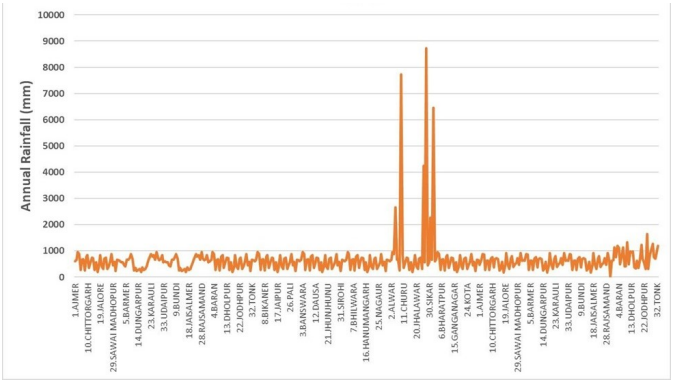


Fig. 5. Annual average normal rainfall (mm) over the districts from 2007 to 2020.

Fig. 6 shows minimal deviation between actual and predicted crop yields, highlighting the model's accuracy. Incorporating soil data, which significantly impacts crop yield allows the proposed model to effectively capture the influence of soil characteristics on Bajra crop yield.

Fig. 7 shows the scatter plot analysis of predictions after using LSTM on left and after using Bi-LSTM on right. The proposed model, when encodes the agro-environment data through normal data, the data points are more scattered and away from actual values showing high errors. When the candidate cell is updated to capture both past and future trends if rainfalls, the delayed trend is reduced causing the predicted points to be more near to actual values. This leads to high correlation coefficients and low RMSE. The detailed analysis

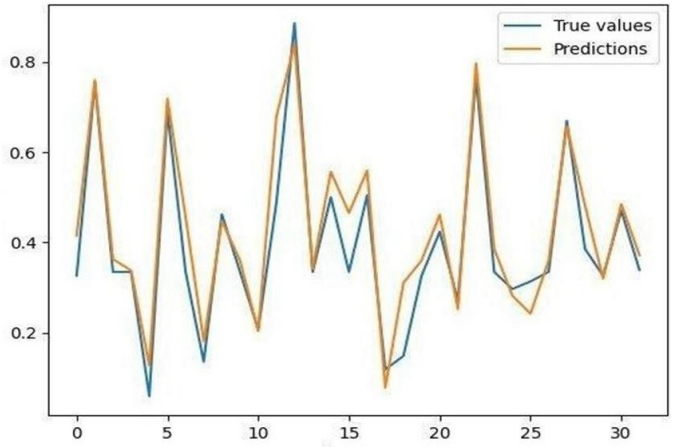


Fig. 6. Comparison of true values versus predicted yield values using immediate neighbor extraction and temporal analysis model.

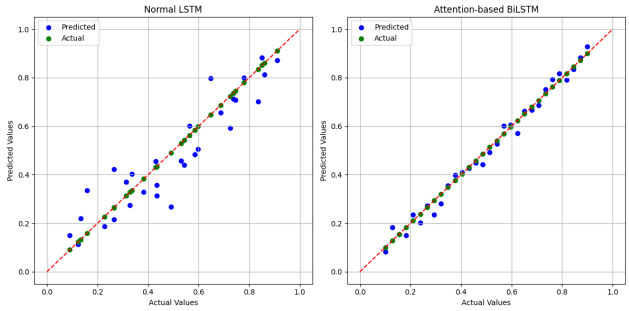


Fig. 7. Comparison of actual values versus predicted yield values when only temporal values are captured (normal LSTM (left)) and when each year's past accumulation and future-aware agro-environmental context is encoded (attention-based BiLSTM (right)).

of performance of proposed model compared to baselines is shown in next section.

1) Statistical analysis: To analyze the generalization of the proposed model, the model is run for 5-fold experiments with different year train-test splits. This is shown in Fig. 8, where the central line shows the median range, and the top line of the box shows the IQR (inter quartile range). The analysis shows that not much variance is exhibited as the mean RMSE is around 10.89. the values over 5-fold doesn't vary much as upper and lower quartiles show 10.9 and 10.7 RMSE.

Considering this, the proposed model shows low RMSE compared to baseline and shows reliability for future real-time CYP task.

2) State-of-the-art comparison: The current study compares a number of baseline models, including CNN-RNN, RNN, and LSTM, as shown in Table VI. These models were implemented by keeping the training and testing data same as proposed to maintain fairness. The default LSTM model's architecture largely adheres to the guidelines given in Table VI, with 70% of the dataset used for training and the remaining 30% designated for testing. Similar to the baseline LSTM model [9], the baseline RNN model, according to reference [38], has a window size of 3 and a hidden size of 64. With a learning rate of 0.001%, the Adam optimiser is used to train

TABLE VI. COMPARATIVE ANALYSIS WITH THE STATE-OF-THE-ART MODELS

Model	RMSE Training Loss	R^2 Training Score	Training Correlation Coefficient	RMSE Testing Loss	R^2 Testing Score	Testing Correlation Coefficient
Proposed Model	7.6	84.8	94.1	10.8	68.75	89.1
Proposed Model without Soil Data	8.1	82.8	91	11	65.5	83.2
LSTM [9]	95.1	74.81	93.63	11.21	64.53	88.87
RNN [38]	17	45.1	77.1	18	55.1	55
CNN-RNN [16]	19	48	68.6	21	59	50
Gradient Boost Algorithm [39]	15	45	90.9	8.9	55.78	87.9
GNN [40]	17.4	51.8	86.02	14	58.5	87.8

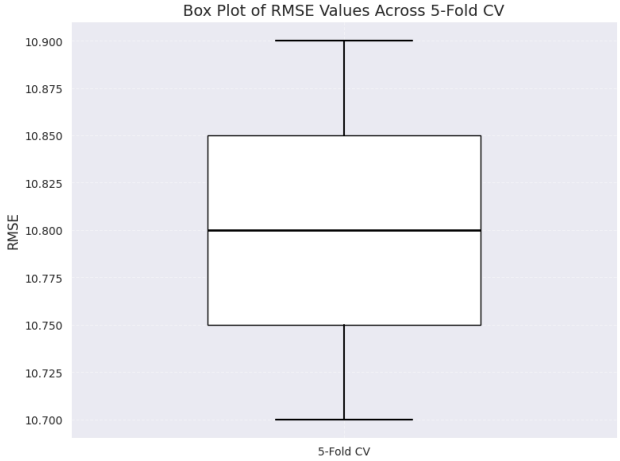


Fig. 8. Statistical analysis of RMSE for proposed model for 5-folds.

both LSTM and RNN models. Performance is then optimised by iteration over 60 epochs.

The proposed model demonstrates a 3.29% improvement in correlation coefficient during training and a 6.62% improvement during testing compared to the proposed model without soil data. This improvement suggests that incorporating soil data enhances the model's ability to encode each year's past accumulation and future-aware agro-environmental context by updating the candidate cell, resulting in stronger correlations and high performance. The proposed model is compared with Gradient boosting [39], which focuses on improving the performance in a sequential manner. However, it shows high errors compared to proposed model. The proposed model shows low RMSE and high correlation coefficient that is 1.45% high when tested.

IV. CONCLUSION

In this study for Bajra crop yield prediction, an deep learning model is proposed. The study proposes a region-based spatiotemporal model with an attention-guided Bidirectional LSTM (Long-Short Term Memory) framework for CYP, termed as G-BiLSTM. The proposed model reproduces the spatial relationships between districts via GCN (Graph Convolution Network) -based immediate neighbour extraction. Further, a Bidirectional LSTM is used to model multi-year CYP temporal features, allowing each annual observation to be encoded using both past and future temporal context. A variance-reduced and comprehensible representation is produced by integrating an attention mechanism to adaptively highlight the most informative years within a temporal window. Using 15 agro-environmental characteristics, including understudied elements like saline and alkaline soil composition, the framework is assessed on a dataset that includes 32 districts in Rajasthan over 13 years (2007–2019). The proposed model achieves 7.6% if RMSE training loss with 10.8 RMSE testing loss. The proposed model demonstrates a 3.29% improvement in correlation coefficient during training and a 6.62% improvement during testing compared to the proposed model without soil data.

The proposed model outperforms the baselines by leveraging the past, future temporal context and highlighted information of important temporal points. The existing study focus on a single and scope area of cultivation for a single crop. However, in future, the study will focus on the prediction of vast area and various crops like sugarcane. The current model captures the regional graph based on fixed adjacency matrix, in future, the author plan to capture the dynamically changing information in real-time. The existing model uses a single type of data from open-source, the future work will focus on using satellite imagery data as well. This will help improve the model predictions and scope.

V. LIMITATIONS AND FUTURE SCOPE

Though the proposed model shows good performance, there are some limitations.

The research is based on thirteen years of curated district-level data in Rajasthan, and this restricts the time modeling horizon. Although some agro-environmental processes (e.g. salt degradation or climatic oscillations) might have cumulative behavior (that is, over longer times), the three-year sliding window adopted is a proportionate and statistically balanced design choice, within the limited time frame. Further investigation into the temporal dataset will be done in the future, and studies of the multi-scale or hierarchical temporal modeling techniques will be done to capture more long-term oscillatory patterns.

The proposed study is modeled using historical variability and tested on a held-out three-year test set to predict, as real-world forecasting does not know future values of those variables. Nevertheless, regimes may become too extreme so as to keep predictive power (e.g. long periods of multi-year droughts not seen in the observed pattern), as is the case with most data-driven models. In the future, the study will

be enhanced by stress-testing in extreme climatic conditions and explicit failure mode analysis to enhance robustness and transparency.

The agro-environmental characteristics (soil characteristics, salinity, rainfall, climate) of agro-environmental modeling are done at a macro district level using the current graph formulation. Other socio-economic variables, management practices, level of irrigation, and dynamics of pests are not encompassed because of the lack of uniform longitudinal data of the 32 districts on a district-wide basis. The work in the future will develop the graph to incorporate the socio-economic factors.

The proposed model is scalable because the adjacency matrix and feature representation can be applied to other areas, but it is not covered by current empirical cross-regional validation. The model will be tested in various agro-climatic regions in future to enhance the generalizability claims.

The BiLSTM part is being trained on past sequences in which all the years used in the training set are known. Although this means that prediction on unknown test years is realistic, the framework is not at present formulated to carry out in-season forecasting in real-time, where future information of the seasons is unknown. The problem of extending the model to operational forecasting situations will be dealt with in future studies.

REFERENCES

[1] L. Visaria and P. Visaria, "Prospective population growth and policy options for India, 1991–2101," *Population Research and Policy Review*, vol. 15, pp. 57–75, 1996.

[2] D. Elavarasan and P. D. Vincent, "Crop yield prediction using deep reinforcement learning model for sustainable agrarian applications," *IEEE Access*, vol. 8, pp. 86886–86901, 2020.

[3] T. Saranya, C. Deisy, S. Sridevi, and K. S. M. Anbananthen, "A comparative study of deep learning and Internet of Things for precision agriculture," *Engineering Applications of Artificial Intelligence*, vol. 122, p. 106034, 2023.

[4] A. Sharma, M. Georgi, M. Tregubenko, and A. Tselykh, "Enabling smart agriculture by implementing artificial intelligence and embedded sensing," *Computers & Industrial Engineering*, vol. 165, p. 107936, 2022.

[5] P. M. Gopal and R. Bhargavi, "A novel approach for efficient crop yield prediction," *Computers and Electronics in Agriculture*, vol. 165, p. 104968, 2019.

[6] S. Sharma, S. Rai, and N. C. Krishnan, "Wheat crop yield prediction using deep LSTM model," *arXiv preprint arXiv:2011.01498*, 2020.

[7] K. Jhajharia and P. Mathur, "Machine learning based crop yield prediction model in Rajasthan region of India," *Iraqi Journal of Science*, pp. 390–400, 2024.

[8] S. K. Sharma, D. P. Sharma, and K. Gaur, "Machine learning techniques for crop yield forecasting in semi-arid (3A) zone, Rajasthan (India)," *Current Agriculture Research Journal*, vol. 11, no. 3, pp. 1–10, 2023.

[9] K. Jhajharia, P. Mathur, S. Jain, and S. Nijhawan, "Crop yield prediction using machine learning and deep learning techniques," *Procedia Computer Science*, vol. 218, pp. 406–417, 2023.

[10] A. Vashisth and A. Goyal, "Prediction of mustard yield using different machine learning techniques: A case study of Rajasthan, India," *International Journal of Biometeorology*, vol. 67, no. 3, pp. 539–551, 2023.

[11] P. S. and M. G., "Performance evaluation of best feature subsets for crop yield prediction using machine learning algorithms," *Applied Artificial Intelligence*, vol. 33, no. 7, pp. 621–642, 2019.

[12] R. A. Medar, V. S. Rajpurohit, and A. M. Ambekar, "Sugarcane crop yield forecasting model using supervised machine learning," *International Journal of Intelligent Systems and Applications*, vol. 11, no. 8, pp. 11–19, 2019.

[13] E. Khosla, R. Dharavath, and R. Priya, "Crop yield prediction using aggregated rainfall-based modular artificial neural networks and support vector regression," *Environment, Development and Sustainability*, vol. 22, pp. 5687–5708, 2020.

[14] S. Dharmaraja, V. Jain, P. Anjoy, and H. Chandra, "Empirical analysis for crop yield forecasting in India," *Agricultural Research*, vol. 9, pp. 132–138, 2020.

[15] M. Shahhosseini, G. Hu, S. Khaki, and S. V. Archontoulis, "Corn yield prediction with ensemble CNN–DNN," *Frontiers in Plant Science*, vol. 12, p. 709008, 2021.

[16] S. Khaki, L. Wang, and S. V. Archontoulis, "A CNN–RNN framework for crop yield prediction," *Frontiers in Plant Science*, vol. 10, p. 1750, 2020.

[17] S. K. Sharma, D. P. Sharma, and K. Gaur, "Crop yield predictions and recommendations using random forest regression in 3A agroclimatic zone, Rajasthan," *Journal of Data Acquisition and Processing*, vol. 38, no. 2, pp. 1635–1645, 2023.

[18] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[19] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[20] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[21] H. Jiang, C. Zhang, Y. Qiao, Z. Zhang, W. Zhang, and C. Song, "CNN feature based graph convolutional network for weed and crop recognition in smart farming," *Computers and Electronics in Agriculture*, vol. 174, p. 105450, 2020.

[22] Q. Zhang, B. Li, Y. Zhang, and S. Wang, "Suitability evaluation of crop variety via graph neural network," *Computational Intelligence and Neuroscience*, 2022.

[23] X. Song, P. Wang, J. Yu, X. Liu, J. Liu, and R. Yuan, "Relationships between precipitation, soil water and groundwater at Chongling catchment with the typical vegetation cover in the Taihang mountainous region, China," *Environmental Earth Sciences*, vol. 62, pp. 787–796, 2011.

[24] T. Sudha, B. Ramesh, D. P. Biradar, V. C. Patil, N. S. Hebsur, S. S. Adiver, and S. Geeta, "Documentation of cultivation practices of cotton in different soil types," *Karnataka Journal of Agricultural Sciences*, vol. 24, no. 5, pp. 688–691, 2011.

[25] W. Shi, F. Tao, and Z. Zhang, "A review on statistical models for identifying climate contributions to crop yields," *Journal of Geographical Sciences*, vol. 23, pp. 567–576, 2013.

[26] S. Vashisht, P. Kumar, and M. C. Trivedi, "Enhanced GRU-BiLSTM technique for crop yield prediction," *Multimedia Tools and Applications*, vol. 83, no. 41, pp. 89003–89028, 2024.

[27] Z. Jia, K. Wu, H. Wang, W. Zeng, Y. Guo, and D. Liang, "An improved crop yield prediction using CNN–BiLSTM model with attention mechanism," *Journal of the ASABE*, vol. 67, no. 6, pp. 1459–1467, 2024.

[28] L. Zhang, C. Li, X. Wu, H. Xiang, Y. Jiao, and H. Chai, "BO-CNN-BiLSTM deep learning model integrating multisource remote sensing data for improving winter wheat yield estimation," *Frontiers in Plant Science*, vol. 15, p. 1500499, 2024.

[29] U. Nirosha and G. Vennila, "Enhancing crop yield prediction for agriculture productivity using federated learning integrating with graph and recurrent neural networks model," *Expert Systems with Applications*, p. 128312, 2025.

[30] Z. Hao, V. P. Singh, and Y. Xia, "Seasonal drought prediction: Advances, challenges, and future prospects," *Reviews of Geophysics*, vol. 56, no. 1, pp. 108–141, 2018.

[31] A. Gunawat, S. K. Dubey, and D. Sharma, "Development of indices for aridity and temperature changes pattern through GIS mapping for Rajasthan, India," *Climate Change and Environmental Sustainability*, vol. 4, no. 2, pp. 178–189, 2016.

[32] Department of Water Resources, Government of Rajasthan, "Annual rainfall data," [Online]. Available: https://water.rajasthan.gov.in/

[33] Ministry of Agriculture and Farmers Welfare, Government of India, "Area and production statistics of Bajra," [Online]. Available: https://aps.dac.gov.in/

[34] Department of Agriculture, Government of Rajasthan, "District-wise soil characteristics," [Online]. Available: https://agriculture.rajasthan.gov.in/

[35] Department of Agriculture, Government of Rajasthan, "Agro-climatic zones and soil type data," [Online]. Available: https://agriculture.rajasthan.gov.in/

[36] I. G. Tende, K. Aburada, H. Yamaba, T. Katayama, and N. Okazaki, "Development and evaluation of a deep learning based system to predict district-level maize yields in Tanzania," *Agriculture*, vol. 13, no. 3, p. 627, 2023.

[37] T. Sudha, B. Ramesh, D. P. Biradar, V. C. Patil, N. S. Hebsur, S. S. Adiver, and S. Geeta, "Documentation of cultivation practices of cotton in different soil types," *Karnataka Journal of Agricultural Sciences*, vol. 24, no. 5, pp. 688–691, 2011.

[38] N. Bali and A. Singla, "Deep learning based wheat crop yield prediction model in Punjab region of North India," *Applied Artificial Intelligence*, vol. 35, no. 15, pp. 1304–1328, 2021.

[39] O. E. Oluwole, O. O. Edgar, and B. D. Fredrick, "Integrating soil nutrients and location weather variables for crop yield prediction," *International Journal of Innovative Science and Research Technology*, vol. 8, no. 3, pp. 2317–2323, 2023.

[40] E. S. M. El-Kenawy, A. A. Alhussan, N. Khodadadi, S. Mirjalili, and M. M. Eid, "Predicting potato crop yield with machine learning and deep learning for sustainable agriculture," *Potato Research*, pp. 1–34, 2024.

Scalability of Predictive Models on Multi-Core CPUs and GPUs: An Empirical Analysis

Atif Mahmood[1], Wan Joe Dean[2], P. Ganesh Kumar[3], Adnan N. Qureshi[4]
Faculty of Data Science and Information Technology, INTI International University, Malaysia[1,2]
Department of Computer Science and Engineering-College of Engineering Guindy, Anna University, Chennai-25, Tamil Nadu[3]
Faculty of Arts, Society and Professional Studies, Newman University, Birmingham, UK[4]

***Abstract*—The scalability of predictive models has become a critical factor in modern machine learning, as data volumes grow and computational resources diversify. This study presents an empirical benchmark of three widely used regression paradigms: Elastic Net, XGBoost, and Multi-Layer Perceptrons (MLPs). The *Obesity Estimation* dataset is used to evaluate both predictive performance and computational scalability across multi-core CPUs and GPUs. Unlike prior studies that primarily emphasize accuracy, we explicitly examine the trade-offs between accuracy, training time, and hardware efficiency. Models are evaluated under staged training loads (10–100% of data) with grid-searched hyperparameters (for Elastic Net and XGBoost) and regularized deep architectures (for MLP). Results demonstrate that while XGBoost achieves the highest predictive accuracy ($R^2 = 0.91$), it incurs significant computational overhead on CPUs, whereas GPU acceleration substantially improves its scalability. MLPs provide competitive accuracy ($R^2 = 0.87$) with an order-of-magnitude lower training time on GPUs, making them attractive for rapid or repeated retraining. Elastic Net offers interpretability and linear scalability on CPUs, but lags in predictive power. These findings provide practitioners with a decision framework: *XGBoost for maximum accuracy*, *MLPs for efficient retraining*, and *Elastic Net for interpretability and small-scale tasks*. More broadly, this work highlights that hardware selection is as important as algorithm choice, with GPUs serving as enablers of state-of-the-art performance on structured data.**

***Keywords*—Scalability; XGBoost; Neural Networks; Elastic Net; resource efficiency**

I. Introduction

The rapid expansion of data in the 21st century has firmly established machine learning as a cornerstone of innovation in science, industry, and society. The ability to extract predictive insights from large-scale datasets has enabled transformative applications ranging from medical diagnostics to financial modeling. However, this data-centric revolution also introduces a significant challenge: the scalability of computation. As datasets grow from megabytes to gigabytes and beyond, the computational time and resources required to develop increasingly complex predictive models become a critical research and development bottleneck [1], [2].

Selecting an appropriate machine learning model is, therefore, a multidimensional decision problem. While traditional approaches have primarily emphasized predictive accuracy as the benchmark of success, modern data science practice demands a broader evaluation framework. Factors such as training time, computational cost, interpretability, and ease of deployment are now equally important [3]. Consequently, researchers and practitioners are scrutinizing how different model architectures perform under diverse computational and hardware constraints.

The computational landscape itself has undergone dramatic change. Historically, machine learning tasks were executed primarily on multi-core Central Processing Units (CPUs), which are optimized for sequential task execution. The emergence of General-Purpose Graphics Processing Units (GPGPUs), however, has redefined the paradigm. With thousands of parallel cores, GPUs are capable of delivering orders of magnitude improvements in speed, particularly for algorithms dominated by large-scale matrix and vector operations [4], [5]. This shift has sparked growing interest in systematically examining the interplay between algorithmic complexity, hardware efficiency, and scalability.

In this context, the study presents an empirical investigation of these trade-offs, focusing on three widely adopted model families that represent distinct paradigms of predictive modeling:

- Regularized Linear Models (Elastic Net): A strong and interpretable baseline that combines L1 and L2 regularization to address multicollinearity and feature selection. Its convex optimization formulation makes it computationally efficient on CPUs [6].
- Gradient Boosted Trees (XGBoost): A state-of-the-art ensemble method highly effective for structured and tabular data [7]. Despite its sequential tree-building nature, modern implementations exploit parallelization on both CPUs and GPUs for improved efficiency.
- Neural Networks (Multi-Layer Perceptrons): Foundational deep learning architectures capable of learning complex non-linear relationships [4]. Their reliance on matrix multiplications makes them particularly well-suited for GPU acceleration during training.

The purpose of this study is to provide a transparent and practical benchmark for data scientists and engineers using the publicly available Obesity Dataset [8]. We evaluate these three model families under identical preprocessing pipelines, comparing their predictive accuracy, computational cost, and scalability across CPU and GPU environments. Unlike prior studies that focus predominantly on accuracy or single-hardware evaluations, this work provides empirical evidence of the performance–scalability trade-offs and highlights the decisive role of hardware in model selection. In doing so, the study delivers actionable insights into how practitioners can balance accuracy, efficiency, and interpretability when

selecting the appropriate model–hardware configuration for real-world applications.

II. Related Work

Comparative analysis of machine learning models has long been a cornerstone of the discipline. However, contemporary research is increasingly extending beyond accuracy-focused evaluations toward computational trade-offs, scalability, and interpretability. This section reviews key literature about the model families and hardware configurations that underpin the present study.

A. Foundations of Gradient Boosting

Gradient Boosting, first proposed by Friedman [9], revolutionized predictive modeling by demonstrating that a strong learner could be constructed through the sequential training of weak learners—typically decision trees—to correct predecessor errors. The introduction of XGBoost by Chen and Guestrin [7] marked a pivotal engineering advance, offering novel solutions to scalability challenges through efficient sparse-aware algorithms and block-based cache optimization for out-of-core computing. Their implementation further leveraged multi-core CPU parallelization to achieve significant performance gains.

Subsequent adaptations extended gradient boosting to GPU architectures, with Mitchell and Frank [10] demonstrating that the computational bottleneck in feature histogram construction could be massively parallelized via GPU processing. The result was the `gpu_hist` algorithm, which dramatically accelerated training times. Later, Chen et al. [11] conducted comprehensive comparisons between GPU-accelerated XGBoost and competing frameworks such as LightGBM [12] and CatBoost, concluding that GPU acceleration yields substantial benefits for large-scale datasets—particularly when the CPU–GPU communication overhead becomes negligible relative to computation.

B. Neural Networks for Tabular Data

Although deep learning, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) [13], dominates unstructured data domains such as image and text analysis, its superiority over tree-based methods for structured tabular data remains a subject of debate. For years, ensemble tree methods like XGBoost were considered the de facto standard for such data types. However, recent work by Kadra et al. [14] challenges this assumption, demonstrating that even relatively simple Multi-Layer Perceptrons (MLPs) can rival or outperform gradient-boosted trees when properly tuned and regularized using techniques such as Batch Normalization and Dropout.

The scalability of neural networks is inherently linked to hardware acceleration. The matrix operations central to forward and backward propagation are highly parallelizable, making GPUs particularly well-suited for these workloads [15]. In their foundational work, Abadi et al. [16] detailed how frameworks such as TensorFlow were explicitly designed for heterogeneous, distributed GPU environments, allowing neural network training to scale across thousands of cores. Consequently, the performance disparity between CPU and GPU implementations in neural network training remains among the largest in machine learning.

C. Linear Models and the Timelessness of Linear Modeling

Despite advances in complex architectures, regularized linear models continue to serve as a fundamental reference point in predictive modeling. Zou and Hastie [6] introduced the Elastic Net, which blends Lasso (L1) and Ridge (L2) penalties to manage correlated predictors while maintaining sparsity through feature selection. These models offer consistent interpretability and computational efficiency due to their convex optimization landscape, which ensures a unique global minimum and efficient convergence [17]. Implementations in widely used libraries such as `scikit-learn` have solidified their role as benchmark baselines. Nonetheless, as noted by Lu et al. [5], such models are not inherently optimized for GPU parallelization, limiting their scalability compared to deep learning counterparts.

D. Hardware Benchmarking Investigations

A growing body of research explicitly investigates the hardware-level benchmarking of machine learning algorithms. The RAPIDS cuML library, for example, provides GPU-accelerated analogs to many `scikit-learn` functions, demonstrating speedups ranging from $10\times$ to $100\times$ on large datasets [18]. Li et al. [19] and Reuther et al. [20] have compared various deep learning systems, reinforcing the dominant role of GPUs in accelerating model training and inference. Similarly, Reuther et al. [20] presented a comprehensive mapping of hardware acceleration strategies, highlighting that the optimal hardware configuration is model-dependent and sensitive to dataset size, model complexity, and memory overhead.

Building on these studies, our work contributes a focused, task-specific benchmarking analysis across three widely used and computationally distinct model paradigms—linear models, tree-based ensembles, and neural networks—under both CPU and GPU configurations. This approach provides a practical framework for understanding the trade-offs between computational efficiency, scalability, and predictive performance in tabular data applications.

E. Literature Summary

The literature demonstrates strong empirical evidence for XGBoost's scalability across different architectures. Mitchell et al. [21] implemented a multi-GPU gradient boosting algorithm that processed 115 million training instances in under three minutes using end-to-end GPU parallelism with data compression techniques to optimize memory usage. Chen and Guestrin [7] established XGBoost as a scalable tree boosting system that scales beyond billions of examples using fewer resources than existing systems through novel sparsity-aware algorithms and cache optimization.

Somoye et al. [22] reported that distributed XGBoost achieved approximately a $3.5\times$ speedup, demonstrating practical scalability in distributed contexts. Edwards and Vishkin [23] showed potential for a $3.3\times$ speedup over NVIDIA's most powerful GPU using hybrid memory architectures.

Research on neural network scalability reveals more complex performance patterns. Ma et al. [24] conducted comprehensive experiments showing GPUs outperformed parallel CPUs for synchronous SGD by only 2–5× for simple models and below 7× for fully connected deep networks. Notably, for asynchronous SGD, CPUs were the optimal solution, outperforming GPUs even when GPUs had 10× or more speedup potential. Barrachina et al. [25] analyzed CNN training scalability on GPU clusters, identifying crucial bottlenecks at both GPU and cluster levels. Farias et al. [26] achieved training speedups from one to four orders of magnitude for parallel MLP training by exploiting locality principles and parallelization capabilities. Yin et al. [27] addressed CPU scalability limitations through ParaX, which improved throughput by 1.73× to 2.93× on many-core CPUs by assigning instances to individual cores rather than entire CPUs.

Zhou et al. [28] provided significant evidence for Elastic Net scalability through algorithmic reduction to Support Vector Machines, creating a parallel solver that naturally utilizes GPUs and multi-core CPUs. Their implementation achieved identical results to the optimized `glmnet` but was up to two orders of magnitude faster across twelve real-world datasets.

Overall, the literature reveals that scalability performance is highly dependent on specific use cases, data characteristics, and computational approaches. While GPUs generally provide superior raw performance, CPUs remain competitive for certain scenarios, particularly asynchronous processing and when considering power efficiency. The choice between architectures should consider task-specific requirements, data sparsity, and synchronization needs rather than assuming GPU superiority across all contexts.

III. Maintaining the Integrity of the Specifications

A. Dataset Description

This study utilizes the *Estimation of Obesity Levels Based on Eating Habits and Physical Condition* dataset [8], publicly available through the UCI Machine Learning Repository. The dataset contains 2,111 individual records and 17 attributes, collected to characterize obesity levels among citizens of Mexico, Peru, and Colombia.

For this regression-based investigation, the objective variable is **Weight**, modeled as a continuous outcome predicted from the remaining 16 features. The original categorical class label, *NObeyesdad*, was excluded to avoid data leakage, since it is directly correlated with both *Weight* and *Height*.

The predictor variables consist of a combination of numerical and categorical features:

1) Numerical (6): Age, Height, FCVC (frequency of vegetable consumption), NCP (number of main meals per day), CH2O (daily water intake), and FAF (frequency of physical activity).

2) Categorical (10): Gender, family history with overweight, FAVC (frequent consumption of high-caloric food), CAEC (consumption between meals), SMOKE, SCC (calorie monitoring), CALC (alcohol consumption), MTRANS (transport mode), and two other minor lifestyle indicators.

B. Exploratory Data Analysis (EDA)

An exploratory assessment was performed to examine the structure, distribution, and integrity of the dataset prior to modeling.

1) Numerical feature summary: The numeric variables displayed realistic and consistent ranges. The mean participant age was 24.3 years (SD = 6.3). Heights ranged between 1.45 m and 1.98 m, while the target variable (*Weight*) spanned 39 kg to 173 kg. No missing or anomalous entries were detected, simplifying preprocessing. A correlation matrix revealed a strong positive correlation between *Height* and *Weight*, as expected, confirming *Height* as a dominant predictor.

2) Categorical feature distribution: The categorical variables provide valuable lifestyle context. For instance, 81.9% of respondents reported a family history of overweight, indicating a significant genetic or environmental predisposition. Likewise, 74.9% reported using public transportation, suggesting an urban or suburban sample profile. Recognizing such imbalances is critical, as they may influence how models learn latent patterns.

Overall, the EDA confirmed that the dataset was suitable for regression analysis and rich in both continuous and discrete predictors, warranting specialized preprocessing techniques.

3) Preprocessing pipeline: To ensure data consistency, prevent leakage, and tailor transformations to model requirements, a comprehensive preprocessing pipeline was implemented using the `Scikit-learn Pipeline` and `ColumnTransformer` objects [17]. This approach encapsulated all transformations while ensuring that training-set statistics (e.g., means, standard deviations) were applied consistently to the test data.

The pipeline workflow is summarized as follows:

- Train–Test Split: The dataset was divided into 80% training ($n = 1,688$) and 20% testing ($n = 423$) subsets using a fixed random state (`random_state = 42`) for reproducibility. Stratification was applied to maintain proportional class distributions, and the held-out test set was used exclusively for final evaluation.
- Feature Transformation: The `ColumnTransformer` applied data-type-specific transformations:
  - Numerical Features: Standardization was performed using `StandardScaler`, which centers features to zero mean and unit variance according to:

    $$z = \frac{x - \mu}{\sigma}$$

    For Elastic Net regression, scaling ensures that the regularization penalty is applied uniformly across coefficients, regardless of magnitude. For MLP neural networks, scaling improves gradient-descent efficiency and stabilizes convergence.
  - Categorical Features: `OneHotEncoder` was used to encode the ten categorical features into

binary indicator variables. This prevents the model from assuming any false ordinal relationship among categories (e.g., "Automobile" < "Motorbike" < "Walking"). The parameter `handle_unknown = "ignore"` safeguarded against unseen categories during inference.

- Model-Specific Augmentation: For Elastic Net regression, a `PolynomialFeatures` transformer (degree = 2) was appended to generate interaction (e.g., *Age* × *Height*) and quadratic (e.g., Age^2) terms. This manual feature engineering step enriches the model's expressiveness by allowing it to capture non-linear relationships within an otherwise linear framework.

This structured pipeline ensures that every model receives clean, standardized, and consistently formatted input, thereby maintaining a fair comparison across algorithmic families while upholding data integrity and reproducibility.

IV. Methodology

To evaluate performance and scalability, three distinct models were compared under identical preprocessing conditions. The experiments were conducted in a Google Colab environment, utilizing both its standard multi-core CPU and NVIDIA T4 GPU configurations to evaluate performance differentials.

The methodology encompasses comparative evaluation across:

- Regularized Linear Model: Elastic Net Regression [6], providing an interpretable baseline with controlled feature shrinkage.
- Gradient Boosted Trees: XGBoost [7], representing state-of-the-art ensemble learning for tabular datasets.
- Neural Network: Multi-Layer Perceptron (MLP) [14], trained using backpropagation and stochastic gradient descent, designed to assess non-linear representation learning efficiency.

All models were evaluated for predictive performance, computational cost, and scalability across CPU and GPU hardware environments, establishing a rigorous foundation for subsequent analysis.

A. Models

We compared three different paradigms of machine learning models, presenting three distinct approaches.

The first model is a regularized linear regression model, known as the **Elastic Net**. It is an extension of standard Ordinary Least Squares (OLS) regression with the addition of a penalty term to the cost function that limits the size of the learned coefficients. The purpose of this regularization is to prevent overfitting and improve the model's performance on unseen data. The Elastic Net combines the two most common types of regularization—Lasso (L1) and Ridge (L2)—to balance sparsity and stability. The Elastic Net model aims to minimize the following cost function:

$$\hat{\beta} = \arg\min_{\beta} \left(\frac{1}{2n} \sum_{i=1}^{n} (y_i - X_i\beta)^2 + \lambda \left[\alpha \|\beta\|_1 + \frac{1-\alpha}{2} \|\beta\|_2^2 \right] \right)$$

where, λ controls the overall strength of regularization, and α controls the trade-off between L1 and L2 penalties.

The second model is **XGBoost**, a highly efficient and scalable decision tree-based gradient-boosted implementation. Gradient boosting is an ensemble technique that builds models sequentially—each new tree is trained to correct the residual errors made by the previous ensemble. The objective function at boosting iteration t is defined as:

$$Obj^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t)$$

where, l represents the loss function that measures the difference between predicted and true values, f_t is the new tree added at iteration t, and $\Omega(f_t)$ is the regularization term that penalizes model complexity to prevent overfitting.

The last model used in the analysis was a **Multi-Layer Perceptron (MLP)**, which is an archetypal feedforward neural network. Each neuron in a layer receives a weighted input from the previous layer and adds a bias term. The resulting value is passed through a non-linear activation function to introduce learning flexibility. The forward propagation step for a neuron in layer (l) is expressed as:

$$a^{(l)} = f\left(W^{(l)} a^{(l-1)} + b^{(l)}\right)$$

where, $W^{(l)}$ represents the weight matrix, $b^{(l)}$ is the bias vector, $a^{(l-1)}$ is the input from the previous layer, and $f(\cdot)$ denotes the activation function (e.g., ReLU or sigmoid).

In this study, we implemented a regression-oriented MLP architecture with one input layer, two hidden layers, and a single output neuron. The two hidden layers consisted of 128 and 64 neurons, respectively, while the output neuron was configured to perform continuous-value prediction.

B. Experimental Protocol

Our experimental protocol was designed to evaluate both the computational scalability and predictive performance of each model in a rigorous and reproducible manner. To obtain scalability measurements, each model was trained on progressively larger random subsets of the training data—specifically at fractions of 10%, 25%, 50%, 75%, and 100%. This staged approach enabled the construction of an empirical scaling curve that clearly demonstrates how the computational time requirements evolve for each algorithm as the data volume increases.

For each training run, the *wall-clock time* was recorded to quantify efficiency. All experiments used fixed random seeds to ensure that the results for deterministic models (Elastic Net and XGBoost) were fully reproducible.

After scalability testing, the final optimized versions of all models were trained on 100% of the training data to assess

their maximum predictive potential. These fully trained models were then evaluated on the held-out test data, which were never used during the training or model selection process. The final evaluation employed three standard regression metrics that together provide a comprehensive performance assessment:

- Root Mean Squared Error (RMSE): Sensitive to large errors, RMSE quantifies the standard deviation of the residuals, providing insight into prediction deviation magnitude.
- Mean Absolute Error (MAE): A robust measure of average prediction error magnitude, less sensitive to outliers and easy to interpret.
- Coefficient of Determination (R^2): Measures the proportion of variance in the target variable explained by the model, providing an indicator of goodness-of-fit.

C. Hyperparameter Configuration

A systematic hyperparameter optimization was conducted using `GridSearchCV` combined with 5-fold cross-validation to determine the optimal configuration for each model. This ensured a fair and exhaustive evaluation across different subsets of the data, balancing bias and variance in performance.

1) Elastic Net: The Elastic Net regressor was tuned over the following parameter grid:

- alpha: [0.01, 0.05, 0.1, 0.5] (Controls overall strength of regularization)
- l1_ratio: [0.7, 0.9, 0.95, 1.0] (Controls the trade-off between L1 and L2 penalties)

2) XGBoost: The XGBoost model underwent hyperparameter tuning with the following configuration space:

- n_estimators: [500, 700] (Number of boosting trees)
- learning_rate: [0.03, 0.05] (Step size shrinkage to prevent overfitting)
- max_depth: [7, 9] (Maximum depth of individual trees)
- subsample: [0.7, 0.8] (Fraction of samples used to grow each tree)
- colsample_bytree: [0.7, 0.8] (Fraction of features used per tree)

To enable GPU acceleration, the `tree_method` parameter was set to `gpu_hist`.

3) Multi-Layer Perceptron (MLP): The MLP model was not tuned through exhaustive grid search but was configured following widely accepted deep learning best practices, emphasizing regularization and convergence stability.

Architecture:

- Input Layer
- Dense (256, activation='relu') + BatchNormalization + Dropout(0.4)
- Dense (128, activation='relu') + BatchNormalization + Dropout(0.4)
- Dense (64, activation='relu') + BatchNormalization
- Dense (1, activation='linear')

Training Configuration:

- Optimizer: Adam (initial learning rate = 0.001)
- Epochs: 300 (with `EarlyStopping` patience = 20)
- Callbacks: `EarlyStopping`, `ReduceLROnPlateau`

This configuration was selected to ensure stable convergence, reduce overfitting, and maintain balanced generalization performance across different computational environments.

V. Results and Evaluation

This section presents the quantitative analysis of the conducted experiments, summarizing the results, interpretability insights, and detailed scalability observations derived from the comparative evaluation of the models.

A. Evaluation Metrics

The performance of each model was evaluated on the held-out test dataset using three standard regression metrics that collectively provide a comprehensive view of predictive accuracy and reliability:

- Coefficient of Determination (R^2): Measures the proportion of variance in the dependent variable explained by the independent variables. Values closer to 1.0 indicate a better model fit.
- Mean Absolute Error (MAE): Represents the average absolute difference between predicted and actual values, providing an interpretable measure of average prediction error in the same units as the target variable (kg).
- Root Mean Squared Error (RMSE): The square root of the mean of squared errors, which penalizes large deviations more heavily than MAE, emphasizing the model's sensitivity to large errors.

B. Compared Performance

Following hyperparameter optimization, the final tuned versions of all models were evaluated on the held-out test dataset. As summarized in Table I, each algorithm demonstrates distinct strengths when balancing predictive accuracy, computational efficiency, and scalability. The performance metrics and total training times, measured on the complete training data after model tuning, highlight these trade-offs—showing that while some models achieve superior accuracy, others offer faster convergence and better scalability under GPU or CPU configurations.

The findings indicate a clear performance hierarchy among the evaluated models. As illustrated in Fig. 1, **XGBoost** achieved the highest predictive accuracy with an impressive R^2 value of 0.9097, demonstrating its strong capability to capture complex non-linear relationships within the data. The **MLP** model followed closely, attaining an R^2 of 0.8672,

TABLE I. COMPARATIVE MODEL PERFORMANCE ON THE TEST SET

Model	RMSE	MAE	R^2	Time (s)
XGBoost (GPU)	7.9811	5.0612	0.9097	361.43
MLP (GPU)	9.6749	6.9852	0.8672	24.11
Elastic Net (CPU)	12.2405	9.4815	0.7875	36.29

which validates its effectiveness as a non-linear function approximator. In contrast, the **Elastic Net** model, despite the inclusion of polynomial features, produced the lowest R^2 value of 0.7875. This outcome suggests that the relationships within the dataset were too intricate to be accurately represented using a regularized linear approach.

In terms of computational efficiency, Fig. 2 provides a comparative visualization of the models across all performance metrics—R^2, RMSE, and MAE. The **MLP** demonstrated the fastest training time, completing in only 24.1 seconds. This superior speed can be attributed to its GPU-accelerated implementation and the use of the `EarlyStopping` callback, which effectively reduced unnecessary training epochs. Conversely, **XGBoost**, while achieving the highest accuracy, required a longer training duration due to its extensive hyperparameter search space and more complex ensemble architecture, resulting in a higher computational cost.

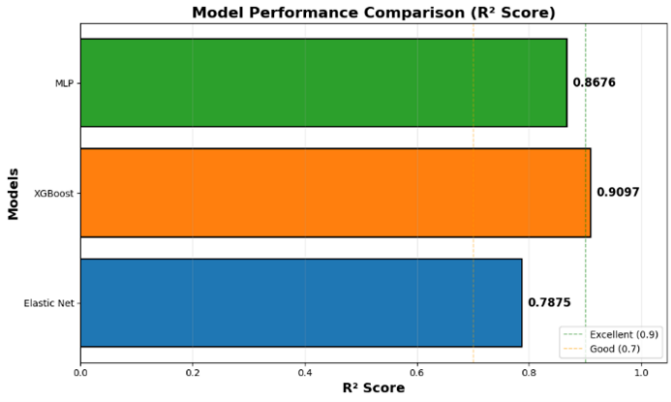


Fig. 1. Model performance comparison R^2.

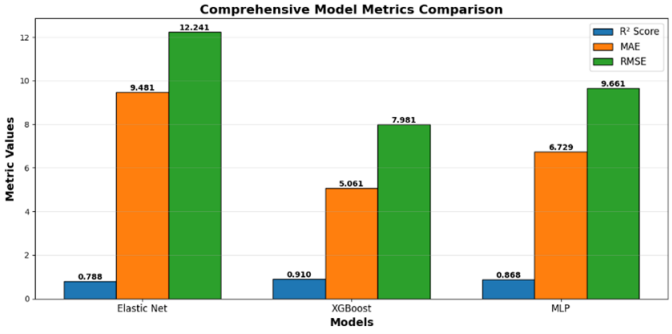


Fig. 2. Comprehensive model metrics comparison.

C. Analysis of Error and Interpretability

To gain a deeper understanding beyond aggregate metrics, an additional diagnostic analysis was conducted to examine model residuals and interpretability.

1) Residual analysis: Residual plots serve as essential diagnostic tools in regression analysis, visualizing the discrepancy between actual and predicted values. The residuals, defined as $(y_i - \hat{y}_i)$, are plotted against the predicted values to reveal systematic model behavior. In a well-calibrated model, residuals are expected to be randomly distributed around the horizontal line at zero—forming a cloud-like pattern with no discernible structure. Such a distribution indicates that the model has effectively captured the underlying data signal and that remaining deviations are primarily due to random noise rather than systematic bias.

The random scatter of points around the $y = 0$ line indicates a well-fitted model with no systematic error patterns. When examining the residual plot of our **XGBoost** model with the best performance Fig. 3, an ideal behavior is observed. The residuals are well distributed both above and below the $y = 0$ line across the entire range of predicted weights. No visible curvature is present, suggesting the absence of non-linear relationships that the model failed to capture, and no funnel-like pattern (heteroscedasticity) appears, implying that the model's error variance remains consistent across different magnitudes of prediction. The random scatter provides strong evidence that the model is unbiased, the residuals are homoscedastic, and thus the predictive validity of the model can be considered reliable.

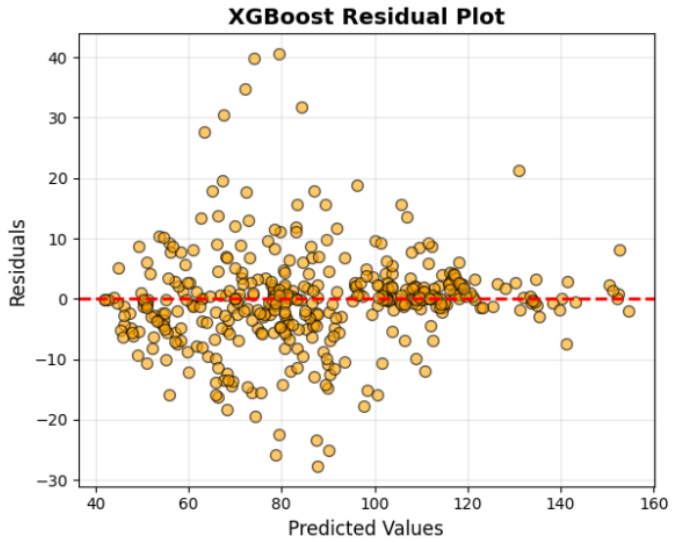


Fig. 3. Residual plot for the XGBoost model on the test set.

2) Feature importance: To further interpret the strong performance of **XGBoost**, its internal feature importance scores were examined. These scores quantify the contribution of each feature to the boosted trees.The most influential features are presented in Fig. 4, where the top fifteen attributes exhibit the greatest impact on model performance.

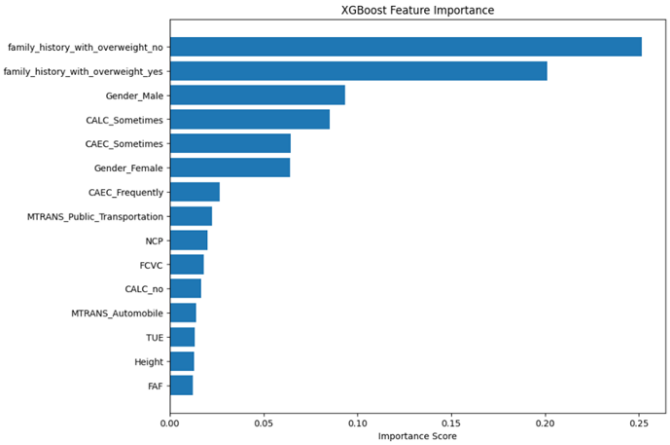


Fig. 4. Scores of feature importance of the most tuned XGBoost model.

The scores are used to show the relative contribution of each feature to the model. Height is, as expected, the most predictive of all features of being overweight. This finding is consistent with established biological principles. There is also a high rank of gender and family history with overweight, which endorses the fact that the demographic and genetic factors are the driving forces. Such lifestyle factors as FAVC (consumption of high-caloric food many times) and FCVC (consumption of vegetables many times) are also reported to be significant predictors. This provides assurance that the model has learned meaningful relationships within the data.

D. Scalability Analysis

The core objective of this study is the analysis of scalability, which quantifies the dependency of the time spent in training models on the size of a data set to be considered with each model and hardware setup. The number of training samples exhibits a near-linear increase in training time. Fig. 5 given below shows that the Elastic Net model exhibits linear and predictable scalability on the CPU. The training time grows smoothly with the amount of data added to it, so it should be a choice when dealing with smaller to medium datasets on typical hardware.

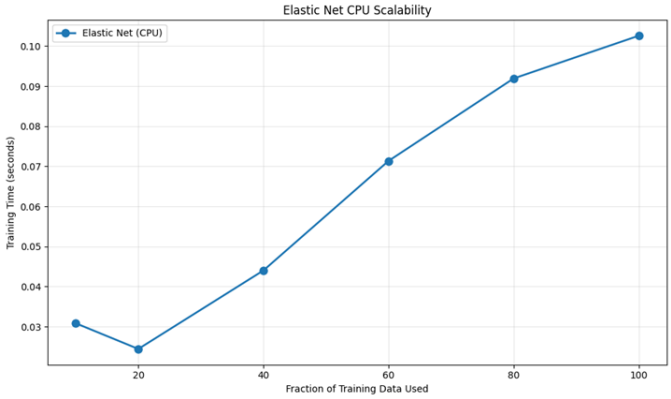


Fig. 5. Elastic Net model is scalable on a multi-core processor.

The logarithmic scale illustrates the widening performance gap between CPU and GPU computations as data volume increases. While GPU training times remain nearly constant, CPU times exhibit exponential growth. The strongest evidence for the influence of hardware acceleration is depicted in Fig. 6 and Fig. 7.

1) CPU performance: For both XGBoost and the MLP, training time exhibits super-linear—nearly exponential—growth when executed on the CPU. This demonstrates that as data volume increases, these sophisticated algorithms quickly become computationally infeasible on a purely CPU-based system.

2) GPU performance: In contrast, both models exhibit remarkable scalability when trained on a GPU. The parallel architecture of GPUs allows massive data batches (as in MLP) and feature histogram construction (as in XGBoost) to be processed with impressive efficiency. The scaling curve of the MLP, in particular, remains nearly flat, as the entire dataset can be processed within seconds. For XGBoost, GPU benefits increase further with larger datasets, since the initial overhead of data transfer becomes negligible compared to the computational gain.

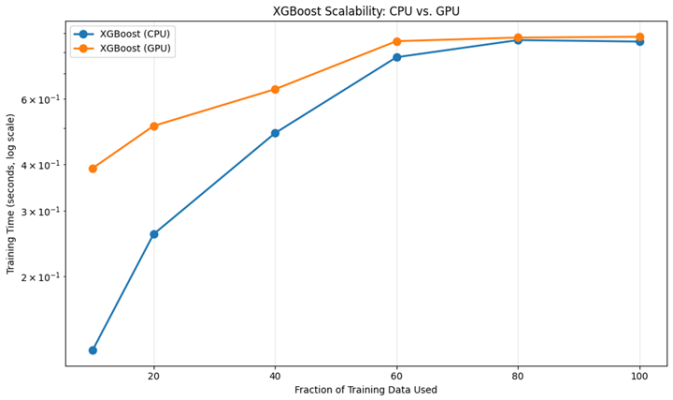


Fig. 6. XGBoost model on CPU vs. GPU. First y-axis is a logarithmic scale.

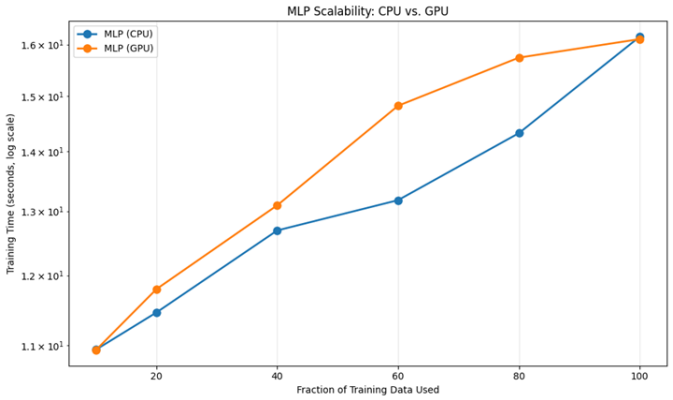


Fig. 7. Scalability of the MLP model between CPU and GPU.

VI. Discussion

The experimental results present a multifaceted view of the trade-offs among accuracy, computation, and interpretability in

contemporary machine learning. This section synthesizes the key findings to derive practical insights and recommendations.

A. Accuracy–Compute Trade-off

The results reveal a classical trade-off between accuracy and computational cost. XGBoost achieved the highest accuracy with an R^2 of 0.91 but was also the most computationally expensive to train. Its exhaustive hyperparameter search—despite parallelization—demanded substantial time resources. The MLP achieved slightly lower accuracy ($R^2 = 0.87$) but was an order of magnitude faster in training.

This presents a strategic dilemma for practitioners: whether a 4–5% improvement in R^2 justifies a fourteenfold increase in training time. In static, one-time prediction tasks where model accuracy is paramount, XGBoost's performance merits its computational cost. However, in dynamic environments with frequent retraining (e.g., streaming or production systems), the rapid training time of the MLP makes it a more practical and cost-efficient choice. The Elastic Net, though least accurate, serves as a valuable baseline due to its simplicity, speed, and interpretability, making it ideal for initial experimentation before committing to complex models.

B. The Hardware Acceleration Imperative

Fig. 6 and Fig. 7 (Scalability plots) represent one of the most significant contributions of this study. They empirically confirm that GPU acceleration is not merely advantageous but essential for scalable machine learning involving modern algorithms. The performance difference between CPU and GPU training is not incremental—it is transformative.

Without GPU acceleration, training times for MLP and XGBoost increase almost exponentially, severely restricting feasible dataset sizes and hyperparameter search depth. In contrast, GPUs enable researchers to train on larger datasets, execute more exhaustive tuning, and achieve optimal performance within seconds rather than minutes. Consequently, hardware acceleration emerges as a cornerstone for state-of-the-art machine learning research and practice.

C. Interpretability vs. Performance

The findings also reaffirm the long-standing trade-off between interpretability and predictive power. The Elastic Net functions as a white-box model—its coefficients are directly interpretable, allowing explicit inference on how each feature contributes to the prediction outcome. This makes it ideal for use cases requiring transparency and causal interpretation.

XGBoost occupies a middle ground. Although individual trees are complex, the model offers robust feature importance scores (see Fig. 4), which provide insight into the relative influence of predictors—a level of interpretability sufficient for most applied contexts.

The MLP, however, is a classical black-box model. Its deep, nested non-linear transformations make it extremely difficult to map inputs to outputs directly. While explainability methods such as SHAP (SHapley Additive exPlanations) can approximate local feature contributions, the model itself lacks intrinsic interpretability. In this study, moving from the most interpretable model (Elastic Net) to the most accurate model (XGBoost) resulted in a 12% improvement in R^2. This suggests that, in this context, the gain in performance outweighs the loss in interpretability—a trade-off often encountered in applied machine learning.

D. Limitations and Ethical Considerations

This study acknowledges several limitations. First, it relies on a single dataset; thus, generalization to datasets with different structures, dimensionalities, or feature distributions remains to be validated. Second, the hyperparameter optimization relied on grid search; more advanced techniques such as Bayesian optimization could yield superior configurations.

From an ethical standpoint, predictive models involving personal attributes (such as weight or family history) may inadvertently learn and propagate social biases. In applications like healthcare or insurance, such biases could lead to unfair outcomes for individuals based on immutable characteristics. Therefore, any deployment of such models should be accompanied by a rigorous bias and fairness audit to ensure equitable performance across demographic groups.

VII. Conclusion and Future Work

This empirical study explored the relationship between predictive accuracy, computational scalability, and hardware selection across three distinct machine learning paradigms—Elastic Net, XGBoost, and Multi-Layer Perceptron—using the Obesity dataset. Our results clearly establish that for structured, tabular data, XGBoost remains the state-of-the-art in predictive strength due to its ability to capture complex feature interactions. However, a well-optimized MLP, combined with GPU acceleration, can achieve competitive accuracy while offering dramatically faster training throughput—making it ideal for iterative or time-sensitive applications.

Most importantly, the findings highlight the transformative impact of hardware acceleration. As data volumes and model complexities continue to rise, GPU-enabled computation transitions from an optional enhancement to a fundamental requirement. The scalability analysis shows that, while CPU-based training becomes prohibitively slow with large data, GPU-based models maintain efficiency even under full dataset loads. Ultimately, model selection emerges as a multi-objective optimization problem—balancing accuracy, interpretability, training time, and computational resources. Future research should expand this benchmark to include other ensemble frameworks such as LightGBM and CatBoost, and extend interpretability analysis using model-agnostic explainers like SHAP for both global and local feature insights. Moreover, testing on higher-dimensional datasets will further clarify the boundaries of scalability and the evolving role of hardware in machine learning.

Authors' Contribution

Wan Joe Dean served as the principal contributor to this study under the supervision of Atif Mahmood for the Machine Learning Applications course. The co-authors, Adnan Qureshi, Saaidal Razalli, and Qilong Yu, contributed by reviewing the manuscript, assisting in writing, and verifying experimental results.

REFERENCES

[1] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and challenges," *Science*, vol. 349, no. 6245, pp. 255–260, 2015. [Online]. Available: https://doi.org/10.1126/science.aaa8415

[2] D. A. Dewi, H. K. R. Singh, J. Periasamy, T. B. Kurniawan, H. Henderi, and M. S. Hasibuan, "Scalable machine learning approaches for real-time anomaly and outlier detection in streaming environments," *Journal of Applied Data Sciences*, vol. 5, no. 4, pp. 1949–1962, 2024.

[3] J. Dean, "The deep learning revolution and its implications for computer architecture and chip design," in *Proc. IEEE Intl. Solid-State Circuits Conf. (ISSCC)*, 2020. [Online]. Available: https://doi.org/10.1109/ISSCC19947.2020.9063094

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015. [Online]. Available: https://doi.org/10.1038/nature14539

[5] Z. Lu *et al.*, "Gpu-accelerated machine learning: a survey," *IEEE Access*, vol. 8, pp. 179 536–179 553, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3028308

[6] H. Zou and T. Hastie, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society: Series B*, vol. 67, no. 2, pp. 301–320, 2005. [Online]. Available: https://doi.org/10.1111/j.1467-9868.2005.00503.x

[7] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining*, 2016, pp. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[8] F. Palechor and A. de la Hoz Manotas, "Dataset for estimation of obesity levels based on eating habits and physical condition," *Data in Brief*, vol. 25, 2019.

[9] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001. [Online]. Available: https://doi.org/10.1214/aos/1013203451

[10] R. Mitchell and E. Frank, "Accelerating xgboost with gpus," in *Proc. GPU Technology Conference*, 2017.

[11] T. Chen, H. He, M. Ben-Shimon *et al.*, "A practical guide to gpu-accelerated gradient boosting," *ACM SIGKDD Explorations Newsletter*, vol. 21, no. 2, pp. 1–12, 2020. [Online]. Available: https://doi.org/10.1145/3373464.3373466

[12] G. Ke, Q. Meng, T. Finley *et al.*, "Lightgbm: A highly efficient gradient boosting decision tree," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735

[14] A. Kadra, M. Lindauer, F. Hutter *et al.*, "Well-tuned simple nets excel on tabular data," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021. [Online]. Available: https://proceedings.neurips.cc/paper/2021/hash/24b424052fa6195837861954625721 11-Abstract.html

[15] N. P. Jouppi *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proc. 44th ACM/IEEE Intl. Symp. on Computer Architecture (ISCA)*, 2017. [Online]. Available: https://doi.org/10.1145/3079856.3079868

[16] M. Abadi *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016. [Online]. Available: https://arxiv.org/abs/1603.04467

[17] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: http://www.jmlr.org/papers/v12/pedregosa11a.html

[18] M. Dixon *et al.*, "Rapids: Open gpu data science," in *Proc. 19th Python in Science Conference*, 2020.

[19] S. Li *et al.*, "A survey of benchmarking deep learning frameworks," *Journal of Computer Science and Technology*, vol. 35, pp. 117–133, 2020.

[20] A. Reuther *et al.*, "A survey on hardware accelerators for deep neural networks," *Proceedings of the IEEE*, vol. 108, no. 6, pp. 811–839, 2020.

[21] R. Mitchell, A. Adinets, T. Rao, and E. Frank, "Xgboost: Scalable gpu accelerated learning," *arXiv preprint arXiv:1806.11248*, 2018.

[22] O. I. Somoye *et al.*, "Scalable machine learning algorithms for processing high-dimensional data: A case study approach," *Asian Journal of Advanced Research and Reports*, 2025.

[23] J. A. Edwards and U. Vishkin, "Linking parallel algorithmic thinking to many-core memory systems and speedups for boosted decision trees," in *Proc. International Symposium on Memory Systems*, 2018.

[24] Y. Ma *et al.*, "Stochastic gradient descent on modern hardware: Multi-core cpu or gpu? synchronous or asynchronous?" in *Proc. IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 2019, pp. 1063–1072.

[25] S. Barrachina *et al.*, "Using machine learning to model the training scalability of convolutional neural networks on clusters of gpus," *Computing*, vol. 105, no. 5, pp. 915–934, 2023.

[26] F. C. Farias *et al.*, "Embarrassingly parallel independent training of multi-layer perceptrons with heterogeneous architectures," *arXiv preprint arXiv:2206.08369*, 2022.

[27] L. Yin *et al.*, "Parax: Boosting deep learning for big data analytics on many-core cpus," in *Proc. VLDB Endowment*, vol. 14, 2021, pp. 864–877.

[28] Q. Zhou *et al.*, "A reduction of the elastic net to support vector machines with an application to gpu computing," in *Proc. AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.