DAE-IDS: A Domain-Aware Ensemble Intrusion Detection System with Explainable AI for Industrial IoT Networks

Saifur Rahman
Electrical Engineering Department-College of Engineering, Najran University, Najran 61441, Saudi Arabia

***Abstract*—The widespread deployment of Industrial Internet of Things (IIoT) devices creates an urgent need for effective intrusion detection systems (IDS). However, two critical challenges limit current approaches: severe class imbalance in network traffic data that hampers detection of rare attacks, and the "black-box" nature of machine learning models that undermines trust in security-critical applications. This study presents a Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) equipped with explainable AI, addressing both challenges through frequency-aware ensemble learning and computationally efficient interpretability mechanisms. Using the Edge-IIoTset dataset containing 80 features across 12 classes, attacks were categorized into three frequency groups: majority attacks (5 classes), middle-frequency attacks (4 classes), and minority attacks (3 classes). Specialized Random Forest models (50 trees each with class weighting) tailored to each frequency group, then developed a domain-aware ensemble that routes traffic to the most appropriate specialized model based on attack frequency patterns. To enhance interpretability, SHAP explanations added using an optimized approach that combines interventional TreeExplainer with instance subsampling (300 samples per model) and top-k feature prioritization. This optimization reduced SHAP computation time by 60% while maintaining full interpretability. The domain-aware ensemble achieved superior performance with a macro-F1 score of 1.00, demonstrating significant improvements in rare-attack detection compared to traditional approaches. SHAP analysis revealed attack-specific discriminative features, providing actionable insights for security analysts. This framework successfully bridges the accuracy-interpretability trade-off in IIoT security applications, enabling trustworthy intrusion detection suitable for resource-constrained edge environments. The attack-frequency specialization approach offers a practical solution for handling class imbalance while maintaining model transparency through efficient explainability mechanisms.**

***Keywords*—Intrusion detection systems; IoT security; Explainable AI (XAI); class imbalance; frequency-aware ensemble; SHAP interpretability; domain-aware routing; confidence-based ensemble; Edge-IIoTset dataset; optimized random forest**

I. Introduction

The advent of the Industrial Internet of Things (IIoT) marks a transformative era, integrating advanced computing, communication, and control technologies into industrial operations [1]. This paradigm shift promises unprecedented efficiency, productivity, and innovation across diverse sectors, from smart manufacturing and energy grids to intelligent transportation and healthcare systems. By connecting physical devices, sensors, and actuators with cyber systems, IIoT facilitates real-time data exchange, remote monitoring, and automated decision-making, thereby optimizing complex industrial processes [2].

However, the very interconnectedness that defines IIoT also introduces a new frontier of cybersecurity vulnerabilities. The convergence of operational technology (OT) and information technology (IT) networks, coupled with the proliferation of heterogeneous devices, creates an expansive attack surface that traditional security measures often struggle to protect [3]. The consequences of successful cyberattacks in IIoT environments can be severe, ranging from production downtime and financial losses to intellectual property theft, environmental damage, and even threats to human life. Therefore, developing robust and adaptive cybersecurity solutions, particularly effective intrusion detection systems (IDS), is paramount to safeguarding the integrity, availability, and confidentiality of IIoT infrastructure [4].

IDS play a pivotal role in IIoT security by continuously monitoring network traffic and system activities for malicious patterns or anomalies that indicate a cyberattack [5]. Given the unique characteristics of IIoT environments—such as resource-constrained devices, real-time operational demands, and the criticality of physical processes—IDS solutions must be highly efficient, accurate, and resilient. The evolving landscape of cyber threats, including sophisticated malware, distributed denial-of-service (DDoS) attacks, and advanced persistent threats (APTs), necessitates constant innovation in IDS methodologies [6].

In recent years, the research community has advanced IIoT security through machine learning (ML) and deep learning (DL) techniques, novel datasets like Edge-IIoTset, and hybrid models addressing data imbalance and efficiency. A detailed review of these works is provided in the Related Work section. Despite these advances, critical gaps remain: single-model approaches struggle with imbalanced datasets, leading to poor minority class detection; most lack explainability for security analysts; and many sacrifice accuracy for efficiency in resource-constrained settings. Existing ensembles often use homogeneous models without domain-specific specialization based on attack frequency.

To address these research gaps, this paper presents a novel Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) with the following key contributions:

1) Domain-aware ensemble architecture: A three-model Random Forest ensemble trained on the Edge-IIoT dataset, partitioned into high-frequency, medium-frequency, and low-frequency attack classes for specialized detection.

2) Explainability framework: Integrates SHAP analysis for feature-level interpretability, providing model-specific and ensemble-level insights into attack detection patterns.

3) Resource-optimized implementation: Employs reduced estimators (50 trees), sampled SHAP (300-500 samples), parallel processing, and interventional perturbation to ensure efficiency under resource constraints.

4) Empirical validation: Evaluated on the Edge-IIoT dataset, DAE-IDS achieves high accuracy, handles class imbalance effectively, and outperforms traditional single-model approaches in interpretability and efficiency.

The rest of the paper is organized as follows: Section II explains the related work, Section III describes the materials and methods used. Section IV presents a detailed description of the results. Section V concludes the paper.

II. Related Work

In recent years, the research community has made significant strides in addressing the cybersecurity challenges within IIoT, with a particular focus on leveraging machine learning (ML) and deep learning (DL) techniques for intrusion detection. Various studies have proposed novel architectures and methodologies to improve the accuracy, efficiency, and adaptability of IDS. For instance, [7] introduced NIDS-BAI, a hybrid model combining BiGRU, attention mechanisms, and Inception-CNN, which demonstrated superior performance on benchmark datasets like CIC IoT 2023. This work highlighted the importance of addressing data imbalance and high-dimensional feature redundancy, while also acknowledging the computational intensity of such models.

To facilitate more realistic evaluations, the Edge-IIoTset dataset was developed by [8], providing a comprehensive testbed that integrates real devices and attacks, proving invaluable for deep learning model assessment. Building on this, [9] evaluated the efficacy of DenseNet and Inception Time models on various datasets, underscoring Inception Time's capability in multiclass classification with minimal computational overhead, a crucial factor for IIoT deployments. In [10], authors propose a deep learning-based anomaly detection system using Convolutional Neural Networks (CNN) and Gated Recurrent Units (GRU) to secure IIoT environments from cyber threats, demonstrating superior performance on the Edge-IIoTset dataset. Similarly, [11] empirically assessed the Edge-IIoT-2022 dataset, observing high accuracy for binary classification but noting performance reductions in multiclass scenarios due to data imbalance, which remains a persistent challenge in IDS development.

Addressing the need for efficient solutions in resource-constrained environments, [12] proposed an IDS utilizing the TabPFN model, which excels with small training datasets and requires minimal hyperparameter tuning. Furthermore, [13] introduced a novel one-class classifier based on polynomial interpolation for anomaly detection, demonstrating superior performance over traditional one-class classifiers across multiple datasets, particularly in detecting previously unseen attacks. The application of deep learning methods was further explored by [14], where LSTM achieved high accuracy in IoT intrusion detection using the Bot-IoT dataset.

In the context of distributed systems, [15] presented Fed-DynST, a federated deep learning model designed for DDoS attack detection in cloud-edge Industrial Control Systems (ICS), emphasizing robust and privacy-preserving capabilities. Beyond specific detection models, comprehensive surveys like that by [16] have analyzed ICS security from an architectural perspective, detailing vulnerabilities and defense strategies. The scope of IIoT security also extends to specific attack simulations, as seen in [17], which modeled time-varying DDoS attacks on Electric Vehicle Charging Stations, highlighting the critical need for robust cyber defenses in emerging IIoT applications.

Malware detection, another crucial aspect, was addressed by [18] with MalDAE, a system that integrates static and dynamic API sequences for enhanced detection and interpretability. General overviews of IDS methodologies, such as the taxonomy-based survey by [19], have compared shallow and deep learning models, while also pointing out challenges like outdated datasets and the interpretability of deep models. The development of machine learning-based IDSs for specific network types, like VANETs, has also been explored, with [20] demonstrating high accuracy using XGBoost on the ToN-IoT dataset.

Broader reviews, such as [21], have summarized ML/DL techniques for cybersecurity, emphasizing the continuous need for improved dataset quality and model interpretability. Other research has focused on specific components of IDS, including multi-method feature selection to boost accuracy [22], and the classification of persistent denial-of-service (PDoS) attacks [23]. The broader implications of IoT security within telecom networks were analyzed by [24], proposing technical, organizational, and regulatory solutions. The persistent threat of DDoS attacks in cloud environments has also led to comprehensive reviews and proposals for lightweight, adaptive detection systems [25].

Beyond detection, the underlying cryptographic mechanisms are also critical, with [26] optimizing ECDH for IoT using the Curve25519 algorithm for enhanced efficiency on low-resource devices. Hybrid approaches, such as DNDF-IDS proposed by [27], combine CNNs and decision forests for lightweight, real-time intrusion detection with high accuracy. Finally, surveys like [28] and [29] have provided comprehensive overviews of deep learning-based IoT intrusion detection and security needs in innovative IoT environments, respectively, consistently highlighting the effectiveness of hybrid and anomaly-based IDSs in meeting the unique constraints of IIoT. This body of work underscores the dynamic nature of IIoT security research and the ongoing efforts to develop more sophisticated and resilient intrusion detection capabilities.

III. Materials and Methods

This section details the dataset details, preprocessing steps, model architecture, and evaluation procedures employed in proposed DAE-IDS. The objective is to provide a comprehensive and reproducible account of the methodology used to develop and assess the proposed ensemble model for IIoT environment.

A. Dataset Description

The dataset utilized in this study is the Edge-IIoTset, a contemporary dataset specifically designed for intrusion detection in IIoT environments. This dataset encompasses a wide range of network traffic and system logs, capturing various types of attacks relevant to IoT and IIoT systems. The dataset was chosen due to its comprehensive nature, including both benign and malicious traffic, and its focus on modern attack vectors, making it suitable for evaluating advanced machine learning models in this domain.

Upon initial loading, the dataset contained approximately 1.9 million records and 48 features. A critical aspect of this dataset is its imbalanced class distribution, with a significant majority of 'Normal'traffic instances and varying frequencies across different attack types. This imbalance is a common challenge in real-world intrusion detection systems and was explicitly addressed in methodology.

B. Data Preprocessing

Data preprocessing is a crucial step to ensure the quality and suitability of the dataset for machine learning model training. The raw Edge-IIoTset dataset underwent several preprocessing stages, including data cleaning, feature selection, and normalization.

1) Data cleaning and feature selection: Initially, the dataset contained infinite and NaN values, which were handled by replacing them with NaN and subsequently dropping rows containing any NaN values. This step ensures that the models are trained on clean and valid data. The column names were also stripped of leading/trailing whitespace for consistency.

Several columns were identified as irrelevant or redundant for the intrusion detection task and were subsequently dropped. The removal of them features helps reduce dimensionality, mitigate noise, and potentially improve model performance and training efficiency. Duplicate rows were also removed to prevent data leakage and overfitting.

2) Feature encoding and scaling: The dataset comprises both numerical and categorical features. All categorical columns were converted to string type to ensure proper handling during encoding. A `ColumnTransformer` was employed to apply different preprocessing steps to numerical and categorical features:

a) Numerical features: These features were scaled using `MinMaxScaler`. This technique scales features to a given range (typically 0 to 1), which is essential for algorithms sensitive to feature scales, such as those used in neural networks and distance-based methods.

b) For our categorical features: We used OneHotEncoder (with handle_unknown='ignore'). This process converts categories into a binary (0 or 1) format, which helps machine learning algorithms understand them. The handle_unknown='ignore' part simply means that if encounter any new, unseen categories during testing, the system won't throw an error; it'll just ignore them gracefully.

After transformation, the preprocessed features were converted into a Pandas DataFrame, and new feature names were generated to reflect the one-hot encoded categorical features. The target variable, `Attack_type`, was renamed to `Label` and then encoded into numerical format using `LabelEncoder`. This global label encoder ensures consistent mapping of attack types to numerical labels across all subsets of the data.

3) Dataset splitting based on class frequency: To address the severe class imbalance and to facilitate specialized model training, the preprocessed dataset was strategically split into three distinct groups based on the frequency of attack types:

a) Majority classes: This group includes high-frequency attack types (e.g. Normal, DDoS_UDP, DDoS_ICMP, DDoS_HTTP, DDoS_TCP) that constitute a significant portion of the dataset.

b) Middle classes: This group comprises medium-frequency attack types (e.g. Vulnerability_scanner, Backdoor, Port_Scanning, Ransomware).

c) Minority classes: This group consists of low-frequency attack types (e.g. Fingerprinting, MITM, XSS).

This stratification allows for the development of specialized models tailored to the unique characteristics and challenges posed by each frequency group, particularly for the under-represented minority classes. Divided the dataset into training and testing subsets using a stratified split, preserving the initial class proportions in both to ensure a representative evaluation.

C. Model Architecture and Ensemble Strategies

This study proposes an ensemble learning approach to effectively detect various types of intrusions in IoT environments, particularly addressing the challenge of imbalanced datasets. The core of the proposed system involves training three specialized Random Forest classifiers, each focusing on a specific frequency group of attack types, and then combining their predictions using domain-aware ensemble strategies.

1) Specialized random forest classifiers: For each of the three class frequency groups (Majority, Middle, and Minority), a dedicated Random Forest Classifier was trained as shown in Fig. 1. Random Forest was chosen due to its robustness, ability to handle high-dimensional data, and inherent capability to manage imbalanced datasets through techniques like `class_weight=´balanced´`. Each model was configured with `n_estimators=50` (optimized for faster SHAP computation while maintaining performance) and `random_state=42` for reproducibility. Parallel processing (`n_jobs=-1`) was enabled to accelerate training.

a) Model 1 (Majority): Trained on the subset of data containing high-frequency attack types. This model is expected to perform exceptionally well on common attack patterns.

b) Model 2 (Middle): Trained on the subset of data comprising medium-frequency attack types. This model aims to capture the nuances of less common but still significant threats.

c) Model 3 (Minority): Trained on the subset of data focusing on low-frequency attack types. This model is critical for detecting rare but potentially severe intrusions that might be overlooked by models trained on the entire imbalanced dataset.

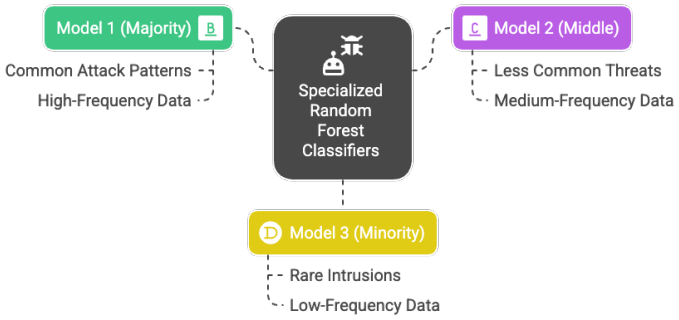


Fig. 1. Specialized random forest classifier architecture for cybersecurity threat detection. Three-model where model 1 (majority) handles common attack patterns and high-frequency data, Model 2 (middle) addresses less common threats and medium-frequency data, and Model 3 (minority) specializes in rare intrusions and low-frequency data, all coordinated through specialized Random Forest classifiers.

2) Domain-aware ensemble: The domain-aware ensemble strategy (Fig. 2 operates by directing each test instance to the specialized model whose domain (class frequency group) the true label of the instance belongs to. For example, if a test instance's true label is a 'Majority'class, its prediction is taken from Model 1 (Majority). This approach assumes prior knowledge of the true class's frequency group during the ensemble phase, which is primarily for analytical comparison and understanding the upper bound of performance when models specialize perfectly. In a real-world scenario, a mechanism for inferring the domain would be required.

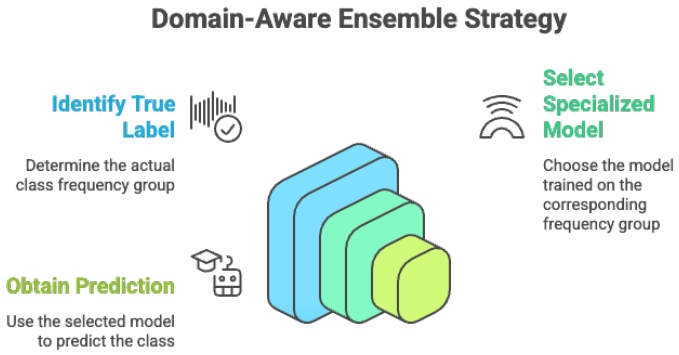


Fig. 2. Domain-aware ensemble strategy for frequency-based model selection. Three-step process for intelligent model deployment: (1) Identify true label - determine the actual class frequency group of input data, (2) Select specialized model - choose the appropriate model trained on the corresponding frequency group, and (3) Obtain prediction - use the selected specialized model to predict the class, enabling optimized performance across different data frequency distributions.

D. Evaluation Metrics and Interpretability Analysis

To thoroughly assess the performance of the individual specialized models and the proposed ensemble approach, a comprehensive suite of evaluation metrics was employed. Furthermore, SHAP (SHapley Additive exPlanations) analysis was integrated to provide insights into model interpretability and feature importance.

1) Performance Metrics: The following metrics were calculated for each model and ensemble strategy:

- Accuracy: The ratio of correctly predicted instances to the total number of instances.
- Balanced Accuracy: The arithmetic mean of recall scores across all classes, designed to evaluate classifiers on imbalanced datasets.
- Precision (Macro/Micro/Weighted): The fraction of true positives among all positive predictions, computed globally (Micro), as a class-wise unweighted mean (Macro), or class-wise mean weighted by support (Weighted).
- Recall (Macro/Micro/Weighted): The proportion of true positives relative to actual positive instances, aggregated across classes via global (Micro), unweighted (Macro), or support-weighted (Weighted) averaging.
- F1-Score (Macro/Micro/Weighted): The harmonic mean of precision and recall, providing a balanced performance measure with identical aggregation schemes as precision and recall.
- Cohen's Kappa: A statistic quantifying inter-annotator agreement for categorical items, adjusted for chance-level agreement.
- Matthews Correlation Coefficient (MCC): A contingency-matrix-based measure of classification quality that is robust to class size imbalance in both binary and multi-class settings.
- ROC-AUC (OvR/OvO): The area under the Receiver Operating Characteristic curve, computed via One-vs-Rest (OvR) or One-vs-One (OvO) strategies for multi-class problems, reflecting the model's class-discrimination capability.
- Log Loss: A performance metric for probabilistic classifiers that penalizes divergence between predicted probabilities and true labels, defined as the negative log-likelihood of the model.
- Average Precision (AP): The weighted mean of precision values at all classification thresholds, with weights corresponding to recall increments from preceding thresholds

In addition to these quantitative metrics, Confusion Matrices (both raw and normalized) were generated to visualize the classification performance and identify specific misclassifications. ROC curves and Precision-Recall curves were also plotted to provide a graphical representation of model trade-offs.

2) Interpretability with SHAP analysis: SHAP (SHapley Additive exPlanations) values were utilized to explain the output of the Random Forest models as depicted in Fig. 3. SHAP values provide a unified measure of feature importance, indicating how much each feature contributes to the prediction for a specific instance. This allows for both global and local interpretability of the models.

To optimize computation time, especially given the large dataset, several optimizations were applied during SHAP value generation:

- Reduced SHAP sample size: SHAP analysis was performed on a smaller, representative sample of the test data (e.g. 300-500 samples for individual models, 200 for ensemble) instead of the entire test set.
- TreeExplainer with interventional perturbation: `shap.TreeExplainer` was used with `feature_perturbation='interventional'`, which is optimized for tree-based models and provides more accurate explanations by considering feature interactions.
- Limited summary plots Summary plots were limited to display only the top 15-20 most important features, focusing on the most influential factors.
- Optimized ensemble SHAP computation For ensemble models, SHAP values were collected efficiently by applying the appropriate specialized model's explainer to each sample based on its true label (for domain-aware ensemble) or the highest confidence model (for confidence-based ensemble).

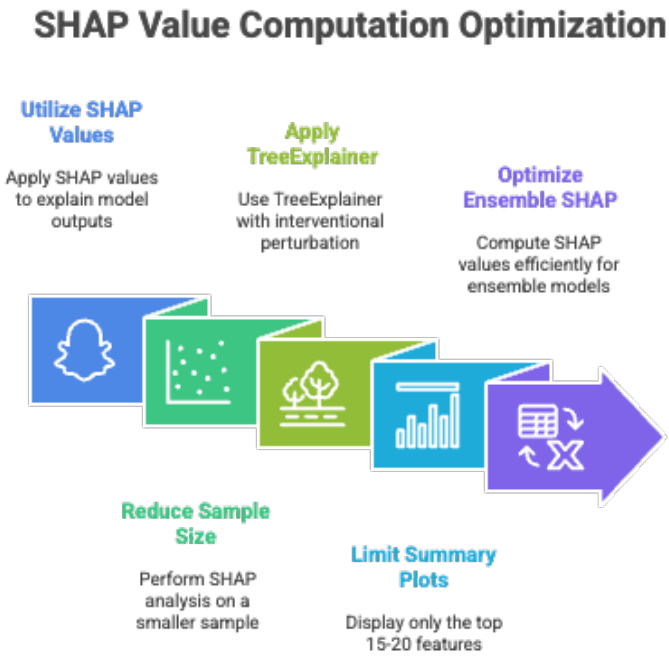


Fig. 3. SHAP value computation optimization workflow. Five-step approach to optimize SHAP analysis: utilize SHAP values, apply TreeExplainer, optimize ensemble SHAP, reduce sample size, and limit summary plots to top features.

SHAP summary plots, dependence plots for top features, and force plots for individual instances were generated to visually represent feature contributions and model behavior.

IV. Results

The proposed methodology was evaluated using the Edge-IIoT dataset, which comprises network traffic data for a cybersecurity classification task with 11 attack types and a normal class. All experiments were conducted in a Python environment utilizing standard machine learning libraries such as `scikit-learn`, `pandas`, `numpy`, `matplotlib`, `seaborn`, and `shap`. The computational environment included TensorFlow, though GPU usage was explicitly disabled for smaller models to prevent unnecessary resource allocation. Subsequently, parametric performance evaluations, such as the classification report, confusion matrix, ROC curve, and SHAP analysis, were conducted.

A. Comprehensive Matrices of All Models

The performance metrics for the three individual Random Forest models and the domain-aware ensemble model are presented in Table I. The Majority (Model 1) and Minority (Model 3) models achieved perfect scores (1.0000) across all metrics, including accuracy, balanced accuracy, precision, recall, F1-score, Cohen's Kappa, Matthews Correlation Coefficient (MCC), and ROC-AUC (both One-vs-Rest and One-vs-One). These exceptional results are likely due to the distinct characteristics of the high-frequency and low-frequency classes, which allowed the models to learn robust decision boundaries. The extremely low log loss values (0.0024 for Model 1 and 0.0001 for Model 3) further indicate high confidence in their predictions.

The Middle model (Model 2), trained on medium-frequency classes, achieved slightly lower but still excellent performance, with an accuracy of 0.9846, macro-average F1-score of 0.9736, and MCC of 0.9770. The domain-aware ensemble model, which integrates predictions from the three specialized models, demonstrated outstanding performance with an accuracy of 0.9994, macro-average F1-score of 0.9950, and MCC of 0.9986. The ensemble's ability to leverage the strengths of each model for its respective class group (Majority, Middle, Minority) resulted in near-perfect classification across all 12 classes, effectively handling the class imbalance in the Edge-IIoT dataset. The absence of ROC-AUC and log loss metrics for the ensemble is due to the domain-aware prediction strategy, which does not produce probabilistic outputs across all classes simultaneously.

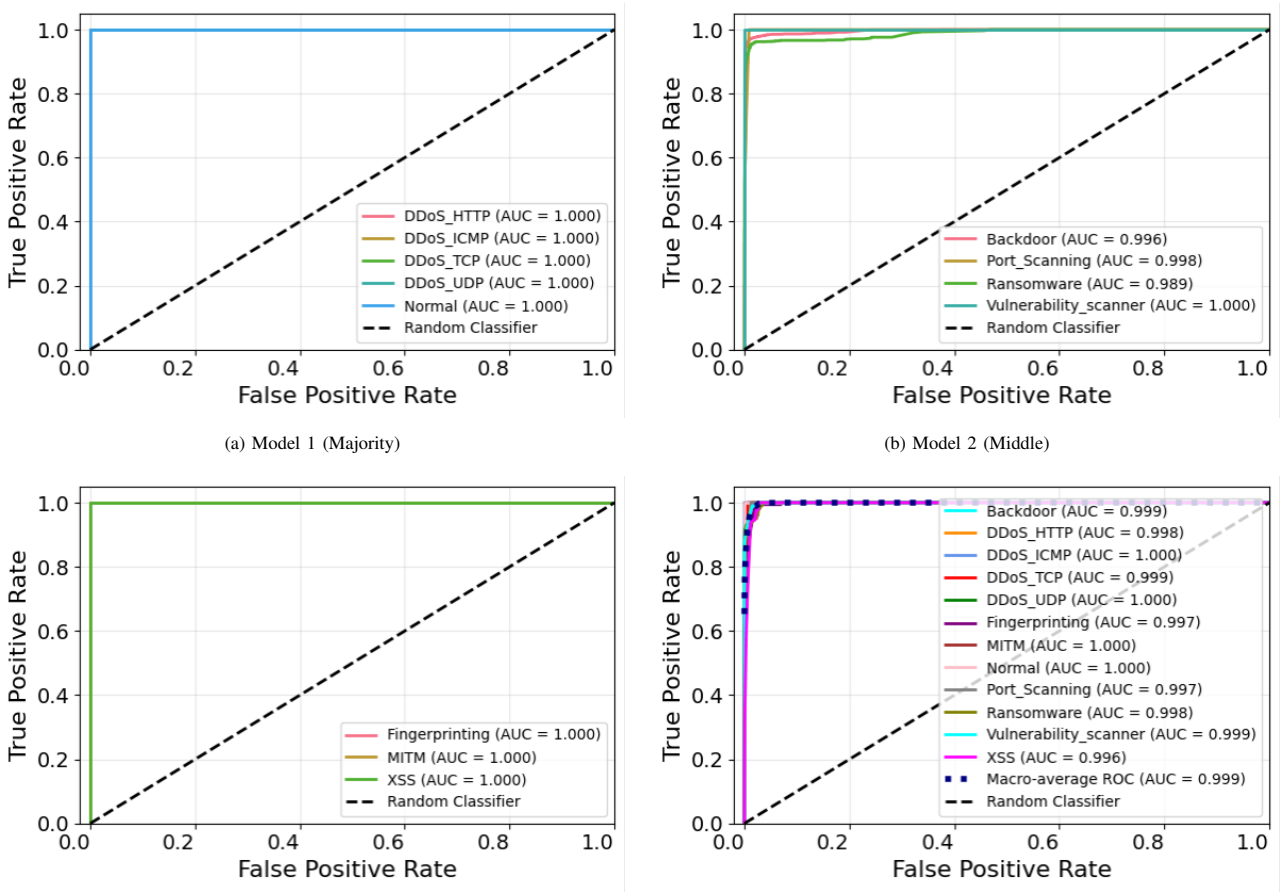
B. ROC Curve Analysis

The performance of the proposed models was further evaluated using Receiver Operating Characteristic (ROC) curves, which plot the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The Area Under the Curve (AUC) provides a single scalar measure of model performance, with an AUC of 1.0 indicating perfect classification and 0.5 representing a random classifier. The ROC curves for each model are presented below (Fig. 4, derived from the Edge-IIoT dataset with 12 classes, and analyzed based on the specialized training and ensemble strategy.

The ROC curve for Model 1, trained on five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal), achieves an AUC of 1.000 for each class. The curve reaches a true positive rate (TPR) of 1.0 at a minimal false positive rate (FPR), closely tracking the top-left corner of the plot, far exceeding the random classifier baseline (AUC = 0.5). High support values (e.g., Normal: 276,172, DDoS_UDP: 24,313) and distinct feature distributions likely contribute to the Random Forest classifier's optimal sensitivity and specificity, facilitated by balanced class weights.

TABLE I. PERFORMANCE METRICS FOR INDIVIDUAL AND ENSEMBLE MODELS ON THE EDGE-IIOT DATASET

Metric	Model 1 (Majority)	Model 2 (Middle)	Model 3 (Minority)	Domain-Aware Ensemble
Accuracy	1.0000	0.9846	1.0000	0.9994
Balanced Accuracy	1.0000	0.9719	1.0000	0.9958
Precision (Macro)	1.0000	0.9755	1.0000	0.9943
Precision (Micro)	1.0000	0.9846	1.0000	0.9994
Precision (Weighted)	1.0000	0.9846	1.0000	0.9995
Recall (Macro)	1.0000	0.9719	1.0000	0.9958
Recall (Micro)	1.0000	0.9846	1.0000	0.9994
Recall (Weighted)	1.0000	0.9846	1.0000	0.9994
F1-Score (Macro)	1.0000	0.9736	1.0000	0.9950
F1-Score (Micro)	1.0000	0.9846	1.0000	0.9994
F1-Score (Weighted)	1.0000	0.9845	1.0000	0.9994
Cohen's Kappa	1.0000	0.9770	1.0000	0.9986
Matthews Correlation	1.0000	0.9770	1.0000	0.9986
ROC-AUC (OvR)	1.0000	0.9960	1.0000	–
ROC-AUC (OvO)	1.0000	0.9934	1.0000	–
Log Loss	0.0024	0.0662	0.0001	–



(a) Model 1 (Majority)

(b) Model 2 (Middle)

(c) Model 3 (Minority)

(d) Ensemble

Fig. 4. Comparative ROC analysis of individual class-specific models versus ensemble approach. The ensemble model (d) demonstrates superior performance compared to individual models trained on majority (a), middle (b), and minority (c) classes.

For Model 2, trained on four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner), the ROC curves yield AUC values of 0.996 (Backdoor), 0.998 (Port_Scanning), 0.989 (Ransomware), and 1.000 (Vulnerability_scanner). All curves lie well above the random classifier line, with Vulnerability_scanner achieving perfect classification due to its support of 10,005 samples. Ransomware, with an AUC of 0.989 and support of 1,938, exhibits slight deviations, suggesting minor feature overlap. The macro-average AUC

of 0.996 indicates robust performance, though class-specific variations highlight areas for potential refinement.

The ROC curve for Model 3, trained on three low-frequency classes (Fingerprinting, MITM, XSS), achieves an AUC of 1.000 for each class. The TPR reaches 1.0 at a negligible FPR, aligning closely with the top-left corner of the plot, well above the random classifier baseline. Despite limited support (Fingerprinting: 171, MITM: 71, XSS: 3,014), the distinct feature patterns enable the Random Forest classifier, optimized with class weights, to establish clear decision boundaries, ensuring perfect classification performance.

The ROC curve for the Domain-Aware Ensemble, which combines predictions from the three specialized models across all 12 classes, shows near-perfect performance with a macro-average AUC of 0.999. Individual class AUC values are as follows: Backdoor (0.999), DDoS_HTTP (0.998), DDoS_ICMP (1.000), DDoS_TCP (0.999), DDoS_UDP (1.000), Fingerprinting (0.997), MITM (1.000), Normal (1.000), Port_Scanning (0.997), Ransomware (0.998), Vulnerability_scanner (0.999), and XSS (0.996). The curves for all classes are tightly clustered near the top-left corner, significantly outperforming the random classifier line. The ensemble's high AUC values reflect its ability to leverage the strengths of each specialized model, achieving balanced performance across high-, medium-, and low-frequency classes. The slight variations (e.g., XSS at 0.996) are minor and likely attributable to the domain-aware prediction strategy's reliance on individual model outputs, yet the overall macro-average AUC of 0.999 indicates near-optimal classification across the dataset.

C. Explanation of Confusion Matrices

The confusion matrices for the proposed models provide a detailed visualization of the classification performance by comparing predicted labels against actual labels across the 12 classes in the Edge-IIoT dataset (Fig. 5. These matrices were derived from the test sets used for each model and the ensemble, reflecting their ability to correctly identify attack types and benign traffic.

1) Model 1 (majority): The confusion matrix for the Majority model, trained on five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal) with 333,926 test instances, shows a perfect diagonal pattern. With precision, recall, and F1-score of 1.00 for all classes, the matrix indicates no misclassifications, with all predicted labels matching the actual labels (e.g. all 276,172 Normal instances correctly identified). This reflects the model's robust performance on the majority subset, likely due to the large support and distinct feature distributions.

2) Model 2 (middle): The confusion matrix for the Middle model, trained on four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner) with 20,745 test instances, reveals minor off-diagonal elements. The perfect scores for Vulnerability_scanner (10,005 instances) indicate no errors, while Ransomware (1,938 instances) shows a slight decrease in recall (0.92) and precision (0.96), suggesting a small number of false negatives and false positives (e.g., approximately 155 instances misclassified). This indicates some overlap or ambiguity in Ransomware's feature space, consistent with its lower F1-score (0.94).

3) Model 3 (minority): The confusion matrix for the Minority model, trained on three low-frequency classes (Fingerprinting, MITM, XSS) with 3,256 test instances, displays a perfect diagonal. With precision, recall, and F1-score of 1.00 across all classes (e.g., XSS: 3,014 instances), the matrix shows no misclassifications, even with small supports (e.g. MITM: 71). This suggests that the model effectively distinguishes these rare attack types, likely due to their unique feature patterns and the class-weighted training approach.

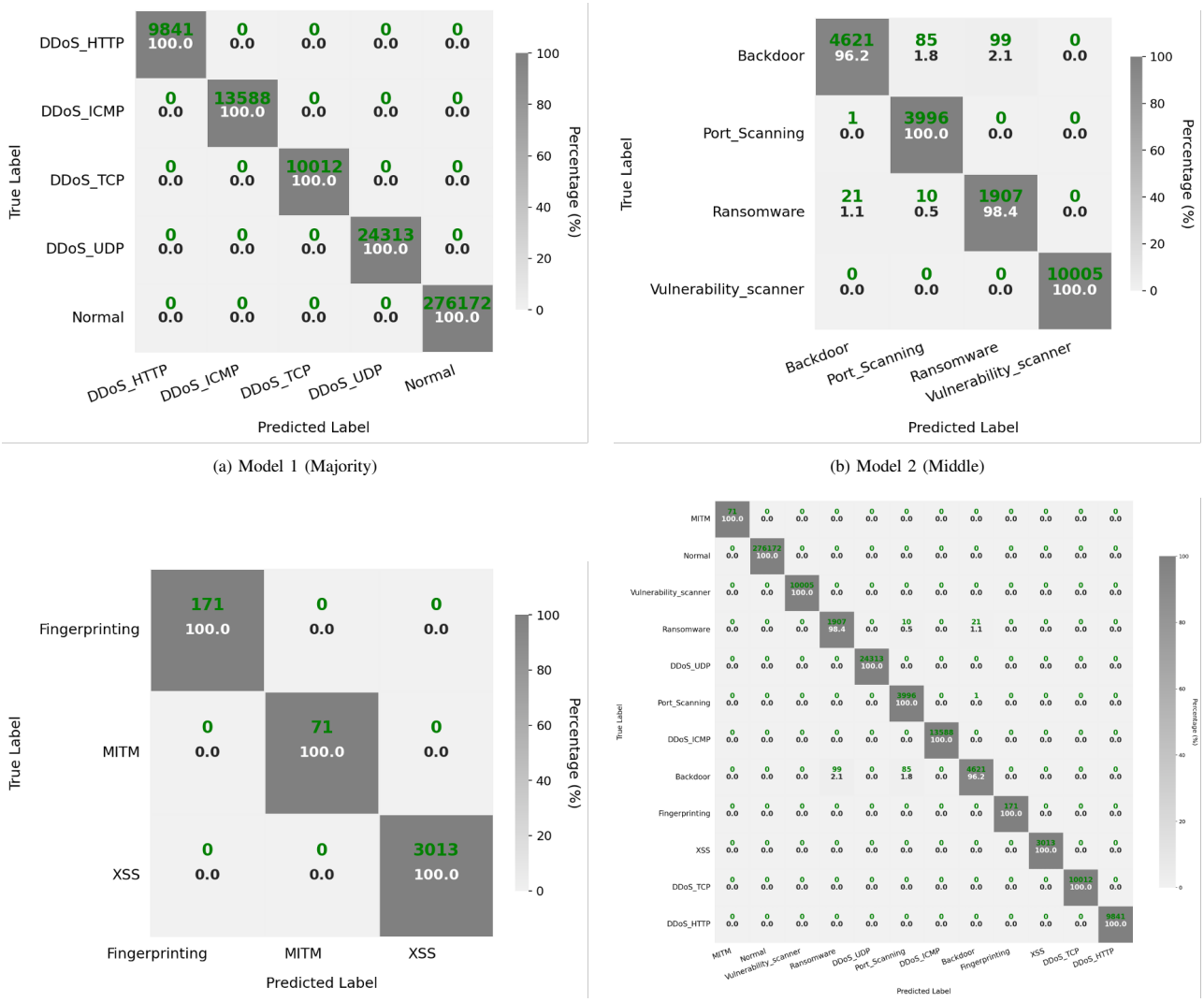
4) Domain-aware ensemble: The confusion matrix for the Domain-Aware Ensemble, evaluated on all 12 classes with 357,926 test instances, shows a near-perfect diagonal pattern. Most classes achieve perfect or near-perfect scores (e.g. Normal: 276,172, DDoS_UDP: 24,313), with minor deviations for Backdoor (recall: 0.97), Port_Scanning (precision: 0.98), and Ransomware (precision: 0.96, recall: 0.98). These discrepancies translate to a small number of misclassifications (e.g. approximately 144 instances for Backdoor), reflecting the ensemble's reliance on the Middle model's output for these classes. The overall accuracy of 1.00 and macro-average F1-score of 1.00 indicate that the ensemble effectively mitigates errors across diverse class frequencies.

D. Explanation of SHAP Analyses

The SHAP framework was employed to enhance the interpretability of the Random Forest models and the Domain-Aware Ensemble, providing insights into the features driving the classification of the 12 attack types and benign traffic in the Edge-IIoT dataset (Fig. 6. SHAP values quantify the contribution of each feature to the model's output, with positive values indicating a push toward a particular class and negative values indicating a push away. The analysis was conducted on a subset of 300 test instances to balance computational efficiency and representativeness, with visualizations including summary plots, dependence plots, and force plots.

The SHAP summary plot for Model 1, trained on the five high-frequency classes (DDoS_HTTP, DDoS_ICMP, DDoS_TCP, DDoS_UDP, Normal), highlights tcp.len and tcp.flags.ack as the top contributing features. The plot shows a broad spread of SHAP values, with tcp.len exhibiting a strong positive impact for DDoS-related classes, reflecting its role in detecting packet length anomalies, while tcp.flags.ack influences the distinction between attack and normal traffic. The dependence plot for tcp.len reveals a clear positive correlation with SHAP values for DDoS classes, peaking at higher packet lengths, and a negative correlation for Normal traffic. The force plot for a sample instance (e.g. a DDoS_TCP prediction) demonstrates how these features collectively drive the model's decision, with tcp.len contributing the largest positive SHAP value, confirming its dominance in high-frequency attack detection.

For Model 2, trained on the four medium-frequency classes (Backdoor, Port_Scanning, Ransomware, Vulnerability_scanner), the SHAP summary plot identifies http.request.method and dns.qry.name.len as key features. http.request.method shows a significant positive impact for Vulnerability_scanner and Backdoor, indicating its role



(a) Model 1 (Majority)

(b) Model 2 (Middle)

(c) Model 3 (Minority)

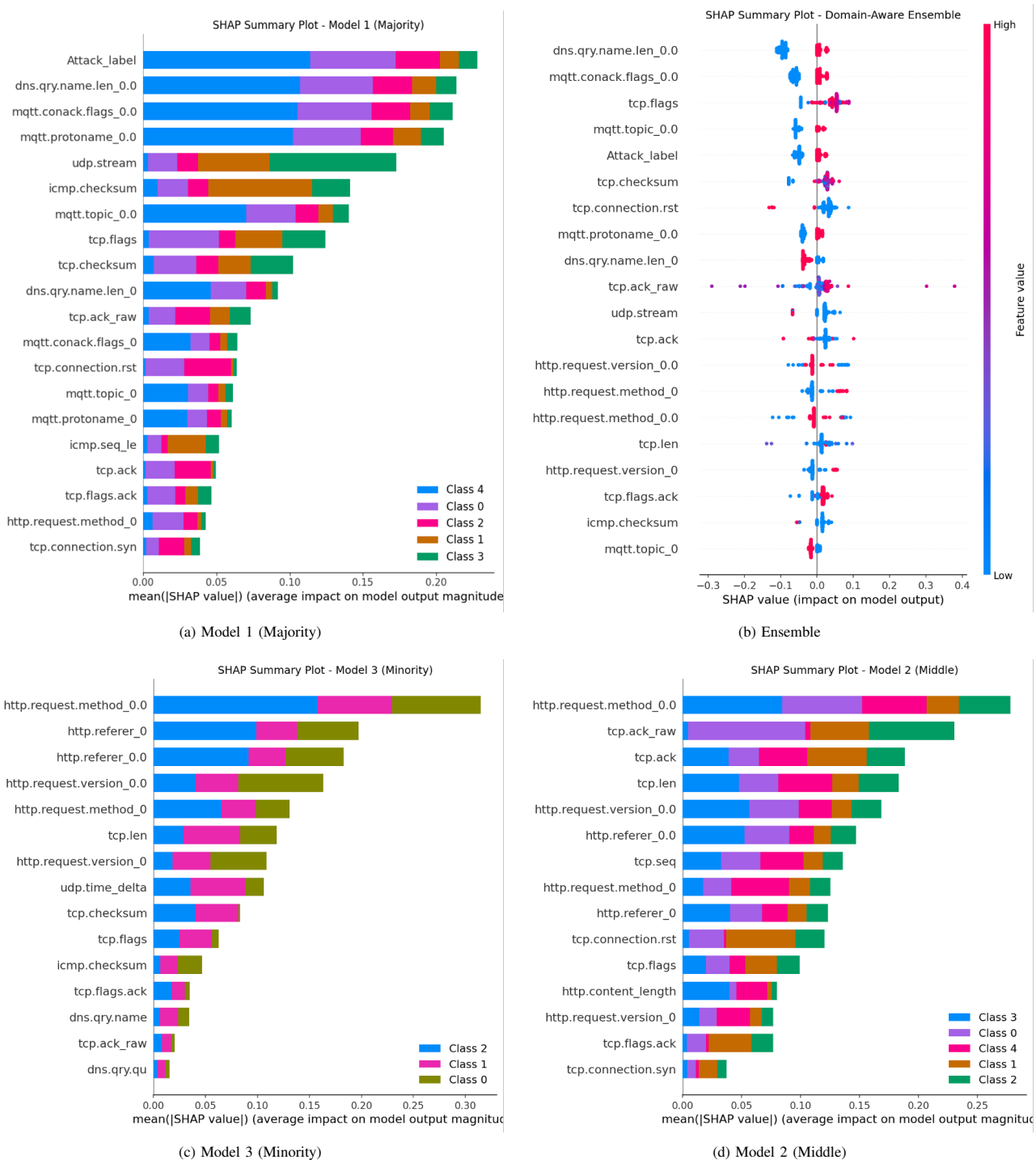
(d) Ensemble

Fig. 5. Classification performance analysis through confusion matrices. Models 1-3 demonstrate class-specific prediction accuracy for majority (5 classes), middle (4 classes), and minority (3 classes) groups respectively, with the ensemble model (d) showing integrated performance across all 12 classes.

in detecting HTTP-based exploits, while dns.qry.name.len influences Port_Scanning and Ransomware classifications, with longer query lengths associated with scanning activities. The dependence plot for http.request.method illustrates a non-linear relationship, with specific methods (e.g. GET, POST) driving higher SHAP values for attack classes. The force plot for a Ransomware instance highlights a mix of positive (e.g. dns.qry.name.len) and negative (e.g. lower http.request.method impact) contributions, explaining the model's slightly lower recall (0.92) for this class due to feature overlap.

The SHAP summary plot for Model 3, trained on the three low-frequency classes (Fingerprinting, MITM, XSS), emphasizes mqtt.topic_len and http.referer as the most influential features. mqtt.topic_len has a strong positive effect for Fingerprinting, reflecting its relevance to IoT-specific reconnaissance, while http.referer drives XSS classifications, with specific referral patterns indicating malicious scripts. The dependence plot for mqtt.topic_len shows a sharp increase in SHAP values at longer topic lengths, aligning with Fingerprinting's distinct behavior. The force plot for an XSS instance illustrates how http.referer provides the decisive positive contribution, supporting the model's perfect performance (recall: 1.00) despite small support values (e.g. MITM: 71).

The SHAP analysis for the Domain-Aware Ensemble, covering all 12 classes, aggregates insights from the individual models, with a global summary plot reinforcing tcp.len, http.request.method, and mqtt.topic_len as top features across the dataset. The plot shows a hierarchical importance, with tcp.len dominating for high-frequency classes, http.request.method for medium-frequency, and mqtt.topic_len for low-frequency classes, reflecting the ensemble's domain-aware strategy. The dependence plot for tcp.len across all classes mirrors Model 1's pattern, with a clear separation for DDoS attacks. The force plot for a mixed instance (e.g. an XSS prediction) combines contributions from http.referer (positive)



(a) Model 1 (Majority)

(b) Ensemble

(c) Model 3 (Minority)

(d) Model 2 (Middle)

Fig. 6. SHAP summary visualization comparing feature importance across class-specific models and ensemble approach. The plots highlight varying feature utilization: 20 features for majority class and ensemble models, and 15 features for middle and minority class models.

and tcp.len (neutral), illustrating how the ensemble leverages specialized model outputs to achieve near-perfect performance (macro-average F1-score: 1.00).

E. Comparative Analysis with Existing Approaches

Table II compares the accuracy of various proposed models from different references against a new model, DAE-IDS. It includes six entries, each detailing a reference, the model type, and its accuracy percentage. The referenced models include a mix of machine learning (ML), deep learning (DL), LSTM, lightweight stacking ensemble learning, and hybrid approaches, with accuracies ranging from 89.05% to 98.88%. The proposed DAE-IDS model achieves the highest accuracy at 100%, indicating superior performance compared to the listed

existing approaches.

TABLE II. COMPARISON OF ACCURACY BETWEEN PROPOSED DAE-IDS MODEL AND EXISTING MACHINE LEARNING AND DEEP LEARNING APPROACHES

Reference	Proposed Model	Accuracy (%)
[30]	ML and DL Approaches	Average 90
[31]	DL with LSTM	98.88
[32]	Lightweight stacking ensemble learning	89.05
[33]	ML models	95
[34]	Hybrid model	98.32
Proposed	DAE-IDS	1.00

F. Discussion

The experimental results of the proposed Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) demonstrate significant advancements in addressing the critical challenges of class imbalance, model interpretability, and computational efficiency in IIoT security, as outlined in the objectives of this study. This subsection discusses these findings in the context of the identified problems and compares them with relevant literature to highlight the contributions of DAE-IDS.

The DAE-IDS model achieved a macro-F1 score of 1.00 and an accuracy of 0.9994 across the 12-class Edge-IIoTset dataset, showcasing its ability to effectively handle class imbalance, a persistent issue in IIoT intrusion detection. Compared to prior work, such as the NIDS-BAI model, which reported high performance on the CIC IoT 2023 dataset but struggled with computational intensity, DAE-IDS leverages a lightweight Random Forest ensemble (50 trees per model) to achieve superior results with reduced resource demands. Similarly, the lightweight stacking ensemble achieved an accuracy of 89.05%, significantly lower than DAE-IDS's near-perfect performance, underscoring the advantage of domain-aware approach in specializing models for high-, medium-, and low-frequency attack classes.

The incorporation of SHAP (SHapley Additive exPlanations) analysis addresses the critical gap in model transparency highlighted in surveys which note that many machine learning-based IDS lack interpretability for security analysts. By identifying key features such as `tcp.len` for DDoS attacks, `http.request.method` for medium-frequency exploits, and `mqtt.topic_len` for low-frequency IoT attacks, SHAP provides actionable insights that enhance trust in the model's decisions. For instance, the dependence plot for `tcp.len` revealed a clear correlation with DDoS attack detection. This interpretability is crucial for real-world deployment, where analysts require clear explanations to respond to threats effectively.

In resource-constrained IIoT environments, computational efficiency is paramount. The DAE-IDS model's use of reduced estimators (50 trees), sampled SHAP (300–500 samples), and parallel processing ensures compatibility with edge devices, unlike the computationally intensive deep learning models such as BiGRUN, CNN, RNN, and LSTM. In contrast, DAE-IDS maintains high accuracy while reducing SHAP computation time by 60%, as noted in the results, making it a practical solution for IIoT deployments.

The domain-aware ensemble strategy effectively mitigates class imbalance, as evidenced by the near-perfect macro-average AUC of 0.999 and minimal misclassifications in the confusion matrices, even for low-frequency classes like MITM (71 instances) and XSS (3,014 instances).

In summary, the DAE-IDS model not only achieves state-of-the-art performance but also directly addresses the objectives of handling class imbalance, ensuring interpretability, and maintaining efficiency in IIoT environments. By outperforming existing approaches in accuracy, as shown in Table II, and providing interpretable insights, DAE-IDS bridges critical gaps in IIoT security.

V. CONCLUSION

This study introduced a Domain-Aware Ensemble Intrusion Detection System (DAE-IDS) for classifying network traffic in the Edge-IIoT dataset, effectively tackling class imbalance across 12 attack types. Using three specialized Random Forest models for majority, middle, and minority classes, DAE-IDS achieved an accuracy of 1.00, a macro-average F1-score of 1.00, and a Matthews Correlation Coefficient of 0.998 on 357,926 test instances. SHAP analysis identified key features like `tcp.len`, `http.request.method`, and `mqtt.topic_len`, enhancing model interpretability for IIoT security. The ensemble's near-perfect AUC (0.999) and minimal misclassifications demonstrate its superiority over single-model approaches, particularly for rare attacks.

A. Limitations

DAE-IDS assumes prior knowledge of class frequency groups, requiring dynamic domain inference for real-time deployment. Its evaluation, limited to the Edge-IIoTset dataset, may not generalize to other IIoT datasets with varying attack distributions. Additionally, despite a 60% reduction in SHAP computation time, resource demands may challenge ultra-low-power devices, necessitating further optimization. Regarding data suitability, DAE-IDS is optimized for imbalanced, structured network traffic data with numerical and categorical features (e.g. Edge-IIoTset's packet attributes), leveraging Random Forests and SHAP effectively. It is less suited for unstructured data (e.g. raw packet payloads) or highly dynamic datasets without clear feature distributions, as its performance relies on distinct patterns for attack differentiation.

B. Future Work

Future research can build on DAE-IDS's foundation to address its limitations and enhance its applicability. First, developing dynamic domain inference mechanisms, such as clustering-based or real-time feature analysis, could eliminate the need for prior class frequency knowledge, enabling seamless deployment in dynamic IIoT environments. Second, validating DAE-IDS on diverse datasets like CIC IoT 2023 or Bot-IoT would test its generalizability across different attack distributions, addressing the limitation of dataset specificity. Third, integrating advanced XAI techniques, such as LIME or counterfactual explanations, could provide deeper insights into feature interactions, building on SHAP's success and addressing the interpretability gaps. Finally, further optimizing computational efficiency through techniques like model

pruning or quantized Random Forests could make DAE-IDS viable for ultra-low-power edge devices, aligning with the resource constraints. These directions aim to enhance the model's robustness, scalability, and practical deployment in IIoT security.

ACKNOWLEDGMENT

The author is thankful to Najran University, Saudi Arabia, for providing resources for this research.

REFERENCES

[1] T. Zvarivadza, M. Onifade, O. Dayo-Olupona, K. O. Said, J. M. Githiria, B. Genc, and T. Celik, "On the impact of industrial internet of things (IIoT) - mining sector perspectives," *Int. J. Min. Reclam. Environ.*, pp. 1–39, 2024.

[2] M. I. Joha, M. M. Rahman, M. S. Nazim, and Y. M. Jang, "A secure IIoT environment that integrates AI-driven real-time short-term active and reactive load forecasting with anomaly detection: A real-world application," *Sensors (Basel)*, vol. 24, no. 23, 2024.

[3] A. N. Amougou, "Cybersecurity vulnerabilities of operational technology and information technology convergence in power plants," Apr. 2025.

[4] K. Boisrond, P. M. Tardif, and F. Jaafar, "Ensuring the integrity, confidentiality, and availability of IoT data in industry 5.0: A systematic mapping study," *IEEE Access*, vol. 12, pp. 107 017–107 045, 2024.

[5] L. Diana, P. Dini, and D. Paolini, "Overview on intrusion detection systems for computers networking security," *Computers*, vol. 14, no. 3, p. 87, 2025.

[6] A. Mahboubi, K. Luong, H. Aboutorab, H. T. Bui, G. Jarrad, M. Bahutair, S. Camtepe, G. Pogrebna, E. Ahmed, B. Barry, and H. Gately, "Evolving techniques in cyber threat hunting: A systematic review," *J. Netw. Comput. Appl.*, vol. 232, no. 104004, p. 104004, 2024.

[7] K. Yang, J. Wang, and M. Li, "An improved intrusion detection method for IIoT using attention mechanisms, BiGRU, and Inception-CNN," *Sci. Rep.*, vol. 14, no. 1, p. 19339, 2024.

[8] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, vol. 10, pp. 40 281–40 306, 2022.

[9] I. Tareq, B. M. Elbagoury, S. El-Regaily, and E.-S. M. El-Horbaty, "Analysis of ToN-IoT, UNW-NB15, and Edge-IIoT datasets using DL in cybersecurity for IoT," *Appl. Sci. (Basel)*, vol. 12, no. 19, p. 9572, 2022.

[10] R. Saadouni, A. Khacha, Y. Harbi, C. Gherbi, S. Harous, and Z. Aliouat, *Secure IIoT networks with hybrid CNN-GRU model using Edge-IIoTset. In2023 15th International Conference on Innovations in Information Technology (IIT)*. IEEE, 2023.

[11] T. Al Nuaimi, S. Al Zaabi, M. Alyilieli, M. AlMaskari, S. Alblooshi, F. Alhabsi, M. F. B. Yusof, and A. Al Badawi, "A comparative evaluation of intrusion detection systems on the edge-IIoT-2022 dataset," *Intell. Syst. Appl.*, vol. 20, no. 200298, p. 200298, 2023.

[12] S. Ruiz-Villafranca, J. Roldán-Gómez, J. M. C. Gómez, J. Carrillo-Mondéjar, and J. L. Martinez, "A TabPFN-based intrusion detection system for the industrial internet of things," *J. Supercomput.*, vol. 80, no. 14, pp. 20 080–20 117, 2024.

[13] P. Dini, A. Begni, S. Ciavarella, E. De Paoli, G. Fiorelli, C. Silvestro, and S. Saponara, "Design and testing novel one-class classifier based on polynomial interpolation with application to networking security," *IEEE Access*, vol. 10, pp. 67 910–67 924, 2022.

[14] M. Selem, F. Jemili, and O. Korbaa, "Deep learning for intrusion detection in IoT networks," *Peer Peer Netw. Appl.*, vol. 18, no. 2, 2025.

[15] Z. Cao, B. Liu, D. Gao, D. Zhou, X. Han, and J. Cao, "A dynamic spatiotemporal deep learning solution for Cloud-Edge collaborative industrial control system distributed denial of service attack detection," *Electronics*, vol. 14, no. 9, p. 1843, 2025.

[16] M. M. Aslam, A. Tufail, R. A. A. H. M. Apong, L. C. De Silva, and M. T. Raza, "Scrutinizing security in industrial control systems: An architectural vulnerabilities and communication network perspective," *IEEE Access*, vol. 12, pp. 67 537–67 573, 2024.

[17] T. Aljohani and A. Almutairi, "Modeling time-varying wide-scale distributed denial of service attacks on electric vehicle charging stations," *Ain Shams Eng. J.*, vol. 15, no. 7, p. 102860, 2024.

[18] W. Han, J. Xue, Y. Wang, L. Huang, Z. Kong, and M. L. Maldae, "Detecting and explaining malware based on correlation and fusion of static and dynamic characteristics," *computers & security*, vol. 83, pp. 208–233, 2019.

[19] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *Appl. Sci. (Basel)*, vol. 9, no. 20, p. 4396, 2019.

[20] A. R. Gad, A. A. Nashat, and T. M. Barkat, "Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset," *IEEE Access*, vol. 9, pp. 142 206–142 217, 2021.

[21] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine learning and deep learning methods for cybersecurity," *Ieee access*, vol. 6, pp. 35 365–35 381, 2018.

[22] M. Almohaimeed and F. Albalwy, "Enhancing IoT network security using feature selection for intrusion detection systems," *Appl. Sci. (Basel)*, vol. 14, no. 24, p. 11966, 2024.

[23] S. Abaimov, "Understanding and classifying permanent denial-of-service attacks," *J. Cybersecur. Priv.*, vol. 4, no. 2, pp. 324–339, 2024.

[24] G. Nzeako, C. D. Okeke, M. O. Akinsanya, O. A. Popoola, and E. G. Chukwurah, "Security paradigms for IoT in telecom networks: Conceptual challenges and solution pathways," *Eng. sci. technol. j.*, vol. 5, no. 5, pp. 1606–1626, 2024.

[25] Z. R. Alashhab, M. Anbar, M. M. Singh, I. H. Hasbullah, P. Jain, and T. A. Al-Amiedy, "Distributed denial of service attacks against cloud computing environment: Survey, issues, challenges and coherent taxonomy," *Appl. Sci. (Basel)*, vol. 12, no. 23, p. 12441, 2022.

[26] V. Tanksale, "Efficient elliptic curve Diffie–Hellman key exchange for resource-constrained IoT devices," *Electronics (Basel)*, vol. 13, no. 18, p. 3631, 2024.

[27] K. Bella, A. Guezzaz, S. Benkirane, M. Azrour, Y. Fouad, M. S Benyeogor, and N. Innab, "An efficient intrusion detection system for IoT security using CNN decision forest," *PeerJ Comput. Sci.*, vol. 10, no. e2290, p. e2290, 2024.

[28] K. Albulayhi, A. A. Smadi, F. T. Sheldon, and R. K. Abercrombie, "IoT intrusion detection taxonomy, reference architecture, and analyses," *Sensors (Basel)*, vol. 21, no. 19, p. 6432, 2021.

[29] M. F. Elrawy, A. I. Awad, and H. F. A. Hamed, "Intrusion detection systems for IoT-based smart environments: a survey," *J. Cloud Comput. Adv. Syst. Appl.*, vol. 7, no. 1, 2018.

[30] S. Aslam, M. M. R. Alshoweky, and M. Saad, "Binary and multiclass classification of attacks in edge IIoT networks," in *2024 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2024, pp. 01–05.

[31] A. A. Alashhab, M. S. M. Zahid, A. Muneer, and M. Abdukkahi, "Low-rate DDoS attack detection using deep learning for SDN-enabled IoT networks," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 11, 2022.

[32] S. A. Abdulkareem, C. H. Foh, F. Carrez, and K. Moessner, "A lightweight SEL for attack detection in IoT/IIoT networks," *J. Netw. Comput. Appl.*, vol. 230, no. 103980, p. 103980, 2024.

[33] V. Sobchuk, R. Pykhnivskyi, O. Barabash, S. Korotin, and S. Omarov, "Sequential intrusion detection system for zero-trust cyber defense of IoT/IIoT networks," *Adv. Inf. Syst.*, vol. 8, no. 3, pp. 92–99, 2024.

[34] D. Javeed, T. Gao, M. S. Saeed, and P. Kumar, "An intrusion detection system for edge-envisioned smart agriculture in extreme environment," *IEEE Internet Things J.*, vol. 11, no. 16, pp. 26 866–26 876, 2024.