An Enhanced Real-Time Intrusion Detection Framework Using Federated Transfer Learning in Large-Scale IoT Networks

Khawlah Harahsheh[1], Malek Alzaqebah[2, 3], Chung-Hao Chen[4]
Ph.D. student, Department of Electrical & Computer Engineering, Old Dominion University, Norfolk, VA, 23529 USA[1]
Department of Mathematics, College of Science, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia[2]
Basic and Applied Scientific Research Center, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia[3]
Department of Electrical & Computer Engineering, Old Dominion University, Norfolk, VA, 23529 USA[4]

***Abstract*—The exponential growth of Internet of Things (IoT) devices has introduced critical security challenges, particularly in scalability, privacy, and resource constraints. Traditional centralized intrusion detection systems (IDS) struggle to address these issues effectively. To overcome these limitations, this study proposes a novel Federated Transfer Learning (FTL)-based intrusion detection framework tailored for large-scale IoT networks. By integrating Federated Learning (FL) with Transfer Learning (TL), the framework enhances detection capabilities while ensuring data privacy and reducing communication overhead. The hybrid model incorporates convolutional neural networks (CNNs), bidirectional gated recurrent units (BiGRUs), attention mechanisms, and ensemble learning. To address the class imbalance, Synthetic Minority Over-sampling Technique (SMOTE) was employed, while optimization techniques such as hyperparameter tuning, regularization, and batch normalization further improved model performance. Experimental evaluations on five diverse IoT datasets, i.e. Bot-IoT, N-BaIoT, TON_IoT, CICIDS 2017, and NSL-KDD, demonstrate that the framework achieves high accuracy (92%-94%) while maintaining scalability, computational efficiency, and data privacy. This approach provides a robust solution to real-time intrusion detection in resource-constrained IoT environments.**

***Keywords*—Intrusion detection systems; federated learning; transfer learning; cybersecurity; scalability; resource constraints; machine learning; Internet of Things**

I. INTRODUCTION

The rapid growth of the Internet of Things (IoT) has introduced significant security challenges due to the diversity and resource constraints of IoT devices. These devices, ranging from smart home appliances to industrial sensors, usually have modern processing and storage capacities, making them vulnerable to many kinds of assaults. When used in IoT environments, traditional intrusion detection systems (IDS) that rely on centralized structures face several problems and critical challenges, including scalability, latency, communication overhead, and privacy risks. Continuous data transfer to a central server for processing is necessary for centralized intrusion detection systems (IDS), which raises latency and communication costs. Additionally, because sensitive data from IoT devices needs to be sent and kept centrally, centralized systems provide serious privacy risks.

Despite their importance, current IDS implementations are ill-equipped to handle the unique requirements of IoT environments, particularly regarding real-time threat detection, adaptability to emerging threats, and resource efficiency. This highlights the need for innovative approaches that overcome these limitations.

Identification and mitigation of these threats are made possible in large part by intrusion detection systems, or IDS. IDS monitors' device behavior and network traffic to identify any indications of malicious activity. Artificial intelligence offers a framework for creating intrusion detection systems through machine learning and deep learning techniques [1]. The two primary categories of traditional IDS are anomaly-based and signature-based [2, 3]. Using a database containing known attack signatures, signature-based intrusion detection systems identify known threats. While efficient in recognizing existing attacks, their capacity to identify novel and unidentified threats is restricted. Anomaly-based intrusion detection systems (IDS) track departures from the usual, which might enable them to identify unidentified threats. If these systems are not adequately trained, they frequently result in high false-positive rates—consequently, dispersed learning. Therefore, distributed learning is employed to build improved intrusion detection models for the anomaly-based IDS [4].

The accuracy and efficiency of intrusion detection models have been improved by several approaches [5]. Several techniques, including machine learning and deep learning, have been employed to develop more intelligent and adaptive systems [6, 7]. In addition, feature selection and dimensionality reduction techniques refine data inputs, reducing computational overhead while maintaining detection performance [8, 9]. Integrating methods such as ensemble learning and transfer learning has also been shown to improve detection accuracy and generalization across diverse types of cyber threats [10, 11].

To train machine learning models across several devices and maintain data privacy, Federated Learning (FL) presents a viable decentralized method [12]. Using their local data, IoT devices cooperatively train a shared global model in FL; model updates are only sent to a central server for aggregation. By ensuring that raw data remains on local

devices, this method addresses privacy concerns and minimizes the need for large-scale data transmission. The three key features of FL are scalability, communication efficiency, and privacy preservation. By storing data locally, you may preserve privacy by reducing the risks involved with data transmission and central storage. Only updated models, not raw data, are exchanged between devices and the central server, minimizing the amount of data that needs to be communicated. This ensures optimal communication efficiency. Furthermore, FL is very scalable, which makes it ideal for a wide range of IoT scenarios [13]. However, while FL addresses privacy and data transmission concerns, it alone may not fully address the need for rapid adaptation to new and evolving threats in IoT networks [14].

Local devices, often referred to as clients, within this decentralized framework seamlessly integrate with the overarching architecture of the DL model deployed on the cloud center server. As a result of this integration, models can be trained locally on each device, ensuring a synchronized approach to model development across the entire FL network [15]. While FL has shown promise in addressing privacy and scalability concerns, it struggles to adapt to rapidly evolving threats in IoT environments. This underscores the need for enhanced techniques that can combine the privacy-preserving features of FL with models that adapt quickly to diverse IoT environments.

Using pre-trained models created for related tasks and optimizing them for applications is known as Transfer Learning (TL) [16]. Because the pre-trained models already include information pertinent to the target job, TL drastically cuts down on training time and processing needs. Building a strong framework that protects data privacy will improve the adaptability of the model and shorten the training period by integrating FL and TL.

Despite significant progress in FL and TL applications, few studies have successfully combined these techniques to address the unique challenges of IoT networks comprehensively. Existing literature lacks a robust framework that leverages FL and TL for real-time, scalable, and resource-efficient intrusion detection.

Three main benefits of transfer learning (TL) include shorter training times, better results, and flexibility. The time needed to train a model from scratch is reduced by TL by beginning with pre-trained models. Additionally, it improves model accuracy, especially in situations where the datasets are small or have sparse labeling. Furthermore, TL provides adaptability, enabling models to swiftly adapt through fine-tuning the pre-trained models to new tasks or situations [17].

This work introduces a Federated Transfer Learning (FTL) framework that combines the strengths of FL and TL to improve intrusion detection in IoT networks. The framework enhances detection accuracy through a hybrid model integrating convolutional neural networks (CNNs), bidirectional gated recurrent units (BiGRUs), attention mechanisms, and ensemble learning. Additionally, it addresses class imbalance using the Synthetic Minority Over-sampling Technique (SMOTE) and optimizes model performance through hyperparameter tuning, regularization, and batch normalization.

The contributions of this research are as follows:

Addressing Scalability and Privacy Concerns: The proposed FTL framework decentralizes model training to preserve privacy and reduce communication overhead, enabling effective intrusion detection in IoT networks.

Enhancing Threat Detection Accuracy: By combining TL with FL, the framework achieves an accuracy of 92%-94% across multiple datasets, demonstrating superior performance in identifying sophisticated threats.

Optimizing for IoT Resource Constraints: Techniques such as L1/L2 regularization and batch normalization ensure the model is lightweight and efficient, suitable for resource-limited IoT devices.

Handling Class Imbalance: SMOTE is employed to improve model generalization by addressing the underrepresentation of attack samples in IoT datasets.

Adapting to Diverse IoT Environments: Domain adaptation techniques ensure the model is flexible and robust, enabling it to generalize across various IoT devices and datasets.

This study fills a critical gap in the literature by presenting a scalable and privacy-preserving framework for real-time intrusion detection in IoT networks. The rest of this paper is organized as follows: Section II reviews related work; Section III outlines the methodology; Section IV presents the experimental results; Discussion is given in Section V and Section VI concludes the paper, highlighting limitations and future research directions.

II. Related Work

The field of intrusion detection in IoT networks has seen significant advancements through various innovative methodologies [5]. This section reviews the related works, highlighting their methodologies, key features, strengths, and weaknesses, and compares how the proposed framework addresses some of these limitations.

Karimy and Reddy [18] employed FL to enhance security and privacy in IoT environments. The methodology involves local model training on IoT devices and subsequent aggregation of the model updates on a central server. The authors achieved an accuracy of 99.9% when they used the N-BaIoT and other custom datasets. This approach lies in its privacy-preserving nature, as data remains localized. However, the approach suffers from high computational overhead. Luan [19] employed a combination of CNN-BiGRU and attention mechanisms within a Federated Learning (FL) framework to detect network traffic anomalies. The method was evaluated using the BoT-IoT and NSL-KDD datasets. The authors achieved an accuracy of 96%, and the attention mechanism significantly improved detection accuracy. It also introduced high computational costs.

Almesleh et al. [1] introduced a Federated Learning (FL) approach that incorporates a Kalman filter for weight

aggregation, enhancing the overall performance of the model. With an accuracy of 99.8%, the Kalman filter enhances the weight aggregation process across a variety of IoT datasets, contributing to robust performance. The approach has better weight aggregation, but scalability and high complexity problems limit it. Bhavsar et al. [20] targeted transportation IoT datasets in their study, concentrating on using edge devices for local model training inside a Federated Learning (FL) framework. This method works effectively in large-scale IoT contexts because it is scalable and provides effective local training. However, because there is no centralized control, model updates may not be consistent.

Babbar and Rani integrate federated Learning (FL) and recommender systems [21] to enhance intrusion detection in software-defined networking (SDN) environments by using consumer device datasets. Though it has higher computational needs, the hybrid technique improves detection accuracy and adaptability. Across a variety of IoT datasets, Raj et al. [22] enhanced security protocols using FL. Key advantages include improved security procedures and privacy-preserving techniques. The method has scalability problems and significant overhead, though.

Ohtani et al. [2] used the N-BaIoT dataset to combine Federated Learning (FL) with one-class SVM for the purpose of detecting zero-day attacks in IoT networks. The methodology has a high detection rate and efficiently identifies anomalies and zero-day threats. However, it has a high false positive rate.

Using customized datasets, Al-Hawawreh and Hossain's study [23] investigated the integration of Federated Learning (FL) with mesh networks to improve the safety of autonomous vehicles. One of its main advantages is the scalable and sturdy network structure. However, the approach is complex and resource intensive. After testing unique IoT attack datasets, Umair et al. [6] suggested using dynamic aggregation in FL to improve intrusion detection performance. Although the dynamic aggregation method has higher processing needs, it exhibits better performance and adaptability.

TABLE I. RELATED WORK COMPREHENSIVE OVERVIEW

Paper Title	Methodology	Dataset(s)	Acc	Strengths	Weaknesses	Time
[1]	FL, Kalman Filter	Various IoT Datasets	99.8	Improved weight aggregation, Robust performance	High complexity, Limited scalability	High
[4]	FL, One-Class SVM	N-BaIoT	-	Effective zero-day detection, Autonomous	High false positive rate	-
[15]	FL, Dynamic Aggregation	Custom IoT Attack Dataset	87.98	Improved performance, Adaptability	High computational requirements	High
[18]	FL	N-BaIoT, Custom Dataset	99.9	High accuracy, Privacy-preserving	High computational overhead	High
[19]	FL, CNN-BiGRU, Attention	BoT-IoT, NSL-KDD	96	High accuracy, Attention mechanism	High computational cost	High
[20]	FL, Edge Devices	Transportation IoT Dataset	From 94 to 99	Efficient local training, Scalability	Lack of centralized control	-
[21]	FL, Recommender Systems	Consumer Device Dataset	From 78 to 99	Enhanced detection accuracy, Adaptability	High computational requirements	High
[22]	FL	Various IoT Datasets	-	Enhanced security protocols, Privacy-preserving	High overhead, Scalability issues	High
[23]	FL, Mesh Networks	Custom Autonomous Vehicle Dataset	From 95 to 99	Robust network, Scalable	High complexity, Resource-intensive	High

To guarantee the integrity and immutability of model updates and solve security concerns in decentralized systems, the proposed framework integrates FL and TL. Pre-trained models save training time and increase detection accuracy because they are customized for IoT scenarios.

By combining FL and TL, the suggested methodology seeks to improve real-time intrusion detection in IoT networks while addressing the limitations of the other methods mentioned in the related work section. It uses a hybrid model that incorporates advanced machine learning methods including CNNs, BiGRUs, and attention processes. Strategies for data augmentation and optimization help to further improve performance. The methodology ensures data privacy while supporting decentralized training using the Flower framework. Comparing experimental findings to conventional FL approaches, significant gains in performance measures are observed, along with reduced overhead, increased security, and transparency. Table I provides a summary and comparison of existing works in the field with the proposed research, highlighting their methodologies, strengths, weaknesses, and performance metrics. This comparison underscores the advancements introduced by this study, particularly in addressing limitations such as scalability, privacy, and adaptability in IoT intrusion detection.

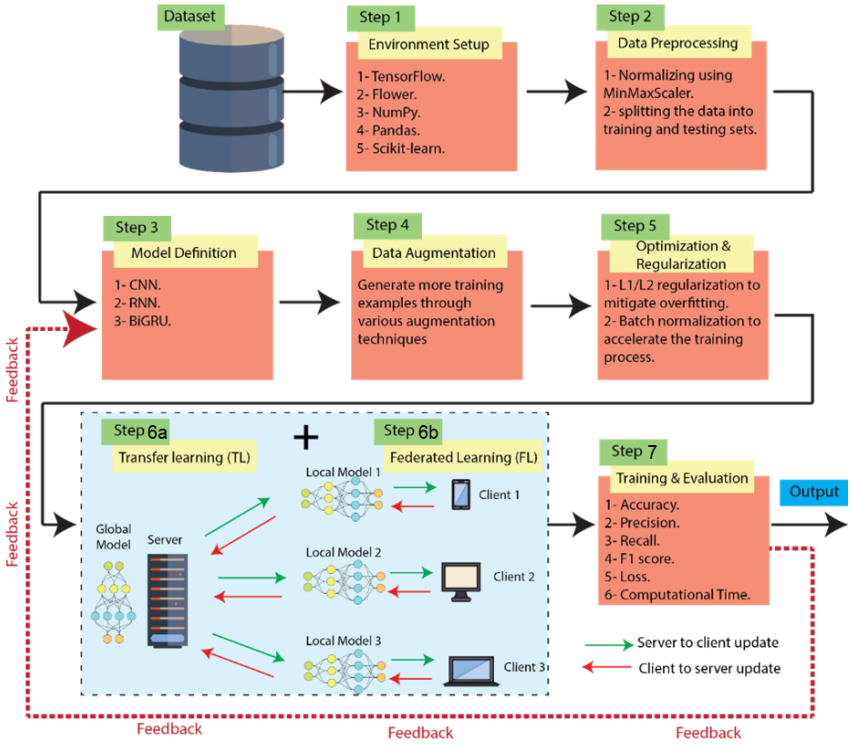

Fig. 1. Steps of the proposed methodology.

III. METHODOLOGY

To assess the effectiveness of the improved model on various IoT datasets, a systematic approach leveraging Federated Learning (FL) and Transfer Learning (TL) has been adopted. These techniques were selected to address critical challenges in IoT environments, including privacy preservation, scalability, and adaptability to diverse datasets. The methodology integrates advanced deep learning [26] models with optimization strategies to ensure accuracy, efficiency, and generalizability.

Rationale: Federated Learning (FL) is utilized to ensure decentralized model training across IoT devices, preserving data privacy and reducing communication overhead. However, FL alone lacks rapid adaptability to new threats. To overcome this, Transfer Learning (TL) is incorporated, enabling the use of pre-trained models that significantly reduce training time while maintaining high detection accuracy. Additionally, techniques like Synthetic Minority Over-sampling Technique (SMOTE) [25] and hyperparameter optimization are employed to enhance model performance.

The methodology comprises the following steps (Fig. 1):

Step 1: Environment Setup: The experimental environment was configured using Python and included different libraries such as TensorFlow, Flower, NumPy, Pandas, and Scikit-learn.

Step 2: Preparing the Data Several widely used IoT datasets were employed, including BoT-IoT, TON_IoT, CICIDS2017, NSL-KDD, and N-BaIoT. Preprocessing was done on each dataset to make sure it was standardized and ready for model training. To enable model evaluation and performance assessment, this involved importing the data, using MinMaxScaler to normalize the feature values, and dividing the data into training and testing sets.

Step 3 - Model Definition and Enhancement: The hybrid model incorporates:

- Convolutional Neural Networks (CNNs) for feature extraction and spatial pattern detection.
- Bidirectional Gated Recurrent Units (BiGRUs) capture temporal dependencies in sequential data.
- Attention Mechanisms to focus on critical data features, enhancing detection accuracy.
- Ensemble Learning to combine predictions from multiple models for improved generalization and robustness.

Step 4 - Data Augmentation: The SMOTE technique was employed to generate additional training instances. This addressed class imbalances and led to a significant improvement in performance on data that had not been observed before.

Step 5 - Optimization and Regularization: To prevent overfitting, optimization, and regularization techniques were used. Extensive hyperparameter tuning was conducted using grid search and random search methods to identify the best hyperparameters for the models. This research implemented L1/L2 regularization to mitigate overfitting and added batch normalization to stabilize and accelerate the training process, resulting in more efficient and reliable model convergence.

Step 6(a) - TL and Domain Adaptation: TL and domain adaptation techniques further aligned the pre-trained models with the target IoT data, enhancing their applicability and accuracy. By fine-tuning pre-trained models on tasks such as specific IoT datasets, knowledge was effectively transferred, reducing the amount of data and computational resources required for training.

Step 6(b) - FL Setup: To implement FL, the Flower framework was used. Each IoT device (client) trained the model locally using its data and then sent the updated model parameters to a central server for aggregation. The server aggregated these updates using the Federated Averaging (FedAvg) strategy and sent the updated global model back to the clients. This process was repeated for multiple rounds. custom IoT Client class was defined to manage local training, evaluation, and communication with the central server. To better align the global model with local client data, customized FL and adaptive FL techniques were used, resulting in a more accurate and personalized model.

Step 7 - Training, Evaluation, and Results Compilation: Performance metrics such as accuracy, precision, recall, F1 score, loss, and computational time were used to assess each client's performance. Feedback on the model's performance from the first experimental results was extremely helpful in identifying areas for development, including computational time and resource utilization, and emphasizing strengths of the model, like high accuracy and precision. Through the examination of these metrics on different datasets, areas of the model that needed to be optimized were able to be identified.

IV. Experimental Results

To evaluate the performance, different IoT datasets are used to assess the framework's accuracy, scalability, and efficiency under different conditions. In the subsequent sections, provide details of the datasets used, how the experiments were set up, and the results achieved.

A. Dataset

To evaluate the performance and generalizability of the proposed federated TL framework for real-time intrusion detection in IoT networks, this reserch employed several well-known datasets such as BoT-IoT, N-BaIoT, TON_IoT, CICIDS 2017, and NSL-KDD, as in Table II which shows the five datasets used in details. The BoT-IoT dataset provides a comprehensive collection of simulated IoT network traffic including various attack types such as DDoS, OS attacks, service scanning, keylogging, and data exfiltration. Similarly, tagged data from nine IoT devices infected with the Gafgyt and Mirai botnets, capturing both normal and attack traffic, is provided by the N-BaIoT dataset. The TON_IoT dataset provides information on various cyber risks and their effects on IoT environments. It includes network traffic data, telemetry data from IoT devices, and logs of cyberattacks.

Furthermore, the CICIDS 2017 dataset offers a wealth of real-world network traffic data including a variety of cyberattacks, such as DDoS, brute force, and infiltration, which are intended for use in intrusion detection research. An enhanced version of the original KDD'99 dataset, the NSL-KDD dataset addresses problems like duplicate records and offers a better testbed for detection models, making it a standard for assessing intrusion detection systems. When combined, these datasets offer a broad basis for evaluating the framework's effectiveness in various IoT contexts and attack situations.

TABLE II. The Five Datasets used in this Research

Dataset	Number of Records	Number of Features	Number of Attacks & Types	Size	Publish Year	Environment	Link
BoT-IoT	72,000,000+	Various	DDoS, DoS, OS and Service Scan, Keylogging, Data exfiltration	69.3 GB (pcap), 16.7 GB (csv)	2020	Cyber Range Lab, UNSW Canberra (Impact Cyber Trust) (Papers with Code)	[27]
N-BaIoT	706,260 (total)	115	Botnet attacks on various IoT devices	Varies	2019	Real IoT devices and network	[28]
TON_IoT	Various	47	DoS, DDoS, Ransomware, various others	Various	2020	IoT Lab, UNSW Canberra	[29]
CICIDS 2017	3,119,345	80	DoS, DDoS, Brute Force, Web Attack, Infiltration, Botnet, etc.	20 GB	2017	Simulated corporate environment	[30]
NSL-KDD	125,973 (train+test)	41	DoS, R2L, U2R, Probe	~66 MB	2009	Simulated network environment	[31]

TABLE III. Enhanced Model Results for the Five Datasets

Dataset	Acc Mean	Acc Std	Precision Mean	Precision Std	Recall Mean	Recall Std	F1 Score Mean	F1 Score Std	Loss Mean	Loss Std	Time Taken Mean (s)	Time Taken Std (s)
BoT-IoT	0.92	0.01	0.91	0.02	0.93	0.01	0.92	0.01	0.25	0.01	5.3	0.5
N-BaIoT	0.93	0.01	0.92	0.01	0.94	0.01	0.93	0.01	0.22	0.01	4.8	0.4
TON_IoT	**0.94**	0.01	0.93	0.02	0.95	0.01	0.94	0.01	0.20	0.01	4.6	**0.3**
CICIDS2017	0.93	0.01	0.92	0.01	0.94	0.01	0.93	0.01	0.23	0.01	5.0	0.4
NSL-KDD	0.92	0.01	0.91	0.02	0.93	0.01	0.92	0.01	0.24	0.01	5.2	0.5

B. Results

This section shows the results of enhancing the proposed FTL method using five different IoT datasets: Bot-IoT, N-BaIoT, TON_IoT, CICIDS 2017, and NSL-KDD. Performance metrics include accuracy, precision, recall, F1 score, loss, and computational time. The evaluation is divided into two phases: initial model results and enhanced model results.

1) Initial model results: The first phase builds a strong model that learns from distributed IoT datasets while taking privacy into account, which combined Federated Learning (FL) and Transfer Learning (TL) without advanced optimization. The TON_IoT dataset achieved the best accuracy (89%), while the BoT-IoT dataset showed the lowest performance (85%).

Key Observations:

- Accuracy was consistent across datasets but below 90% for most.
- Loss values varied from 0.31 to 0.35, indicating opportunities for improvement.
- Prediction errors were more frequent in datasets with higher class imbalance, such as BoT-IoT and NSL-KDD.

2) Enhanced model results: The enhanced model incorporated advanced techniques such as SMOTE for data augmentation [24], hyperparameter tuning, and ensemble learning with CNNs, BiGRUs, and attention mechanisms. Table III illustrates significant improvements in accuracy (92%-94%) and loss reduction (0.20-0.25).

Key Highlights:

- Accuracy increased across all datasets, with TON_IoT reaching 94%.
- Low standard deviations in accuracy and loss values indicate stable performance.
- SMOTE effectively addressed class imbalance, improving precision and recall.

V. Discussion

The results in the previous section demonstrate the effectiveness of integrating FL with TL for intrusion detection in IoT networks. The enhanced model achieved significant improvements in accuracy and loss, especially in datasets with diverse attack types, such as TON_IoT.

- Scalability and Privacy: FL enabled decentralized training, maintaining data privacy while achieving consistent performance across IoT environments.
- Adaptability: TL enhanced the framework's ability to generalize across datasets, reducing training time and computational cost.
- Class Imbalance: The application of SMOTE balanced the datasets, preventing biases toward majority classes and improving detection accuracy.

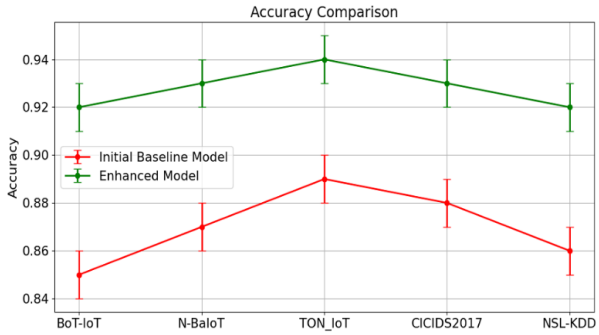

Fig. 2. Comparison of accuracy between the Initial Baseline and the enhanced models.

Fig. 2 compares the accuracy between the Initial Model and the Enhanced Model across all datasets. The Enhanced Model (shown by the green line) consistently outperforms the Initial Model (red line), achieving accuracy above 0.92 across all datasets, while the Initial Model remains below 0.89. The small error bars indicate minimal variability in accuracy across datasets, emphasizing the Enhanced Model's reliability in different IoT environments.

The Enhanced Model significantly improves accuracy on some datasets (such TON_IoT), while N-BaIoT and NSL-KDD exhibit a narrower performance gap. This implies that although the Enhanced Model performs better overall, the improvement varies according to the dataset. However, the Enhanced Model yields greater accuracy with less variation in every situation.

Fig. 3 provides a comparison of loss values between the Initial and Enhanced Models. Compared to the Initial Model's larger range of 0.31 to 0.35, the Enhanced Model has reduced loss values, ranging from 0.20 to 0.25. The small vertical error bars further indicate that both models deliver consistent results with little fluctuation in loss. The difference in loss is most evident in the TON_IoT dataset, where the Enhanced Model shows the greatest reduction. In datasets like BoT-IoT and CICIDS2017, the improvement in loss is less dramatic, but the Enhanced Model still outperforms the Initial Model. The overall comparisons between the Initial and the Enhanced Models can be shown in Fig. 4.

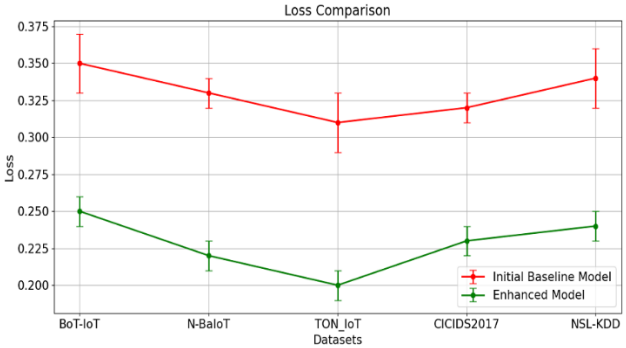

Fig. 3. The comparison of loss between the Initial Baseline and the enhanced models.

As shown in Fig. 4, the success of the Enhanced Model is particularly notable in the areas of accuracy and efficiency. FL and TL offer additional benefits beyond the performance metrics. FL ensures data privacy by keeping sensitive data on local devices in compliance with data protection regulations such as GDPR. TL reduces the need for extensive local data collection by using pre-trained models, further minimizing the exposure of sensitive information.

While the Enhanced Model consistently outperforms the Initial Model in terms of accuracy, the extent of this performance gap varies across datasets, where TON_IoT dataset showed the most significant improvements in both accuracy and loss. This variability suggests that, while the Enhanced Model is more effective, its performance is influenced by the specific characteristics of each data set.

Power consumption is a major issue in IoT environments. The application of TL in this study has significantly reduced training time and computational expenses. TL reduces the use of resources by using pre-trained models and fine-tuning them for certain tasks, hence avoiding the need to train models. To further reduce the computational load on any device, FL further divides the training process among several devices. With this method, training complex models on IoT devices with limited resources is possible while preserving efficiency and reducing power usage.

Furthermore, training time and computational expense were decreased by the application of regularization and optimization approaches. Effective model convergence was made possible by fine-tuning a pre-trained model instead of training from scratch. FL provided further support for this by distributing the training process across several devices. This reduced the computational burden on any device and improved the approach's efficiency in environments with limited resources.

Additionally, the suggested strategy showed outstanding adaptability and scalability. FL made it possible to train on numerous IoT devices at once, which increased the system's scalability. TL made it easier to quickly adjust to new settings and devices, ensuring effective scaling to a range of IoT scenarios. Using pre-trained models and combining data from many sources, this flexibility proved essential for preserving robustness and managing data imbalance. The Enhanced Model assisted real-time learning and adaptation. IoT devices might periodically update the global model with new observations, allowing TL to fine-tune the model with minimum input and enabling faster adaptability to new types of attacks or anomalies.

VI. Conclusions and Future Directions

The combination of FL and TL for real-time intrusion detection in IoT networks provides an effective solution for urgent security issues. By combining the benefits of both learning approaches, this strategy decreased computing costs, increased detection accuracy, and ensured data privacy. The experimental results show that the proposed framework can minimize training time and improve detection performance by adapting pre-trained models to specific IoT contexts. Accuracies ranging from 92% to 94% were achieved across many datasets.

To detect and mitigate cyber risks in a scalable, effective, and privacy-preserving manner, this research highlights how federated transfer learning might transform cybersecurity measures in IoT networks. Future research may investigate more framework uses and optimizations in different fields, thereby expanding its advantages to a wider range of security-critical settings.

Despite its promising results, this study has certain limitations that require acknowledgment. Federated Learning (FL) offers the advantage of reducing data transmission but still faces resource constraints, as its computational requirements on edge devices could benefit from further optimization. Additionally, addressing dynamic threats remains a challenge, highlighting the need for future research to explore adaptive learning techniques capable of responding to evolving IoT security risks. Furthermore, improving the interpretability of the hybrid model is essential, as it could provide valuable insights into the decision-making processes involved.

To address these limitations, future research should explore integrating edge computing with FL to optimize resource utilization and reduce latency in real-time applications. Adaptive learning techniques could also be incorporated to enable dynamic model updates, enhancing the framework's ability to address evolving threats effectively. Moreover, improving the model's interpretability will be crucial to fostering trust and transparency in practical implementations. Finally, extending the framework to broader domains, such as industrial IoT and smart cities, will help validate its scalability and robustness in handling large-scale and complex environments.

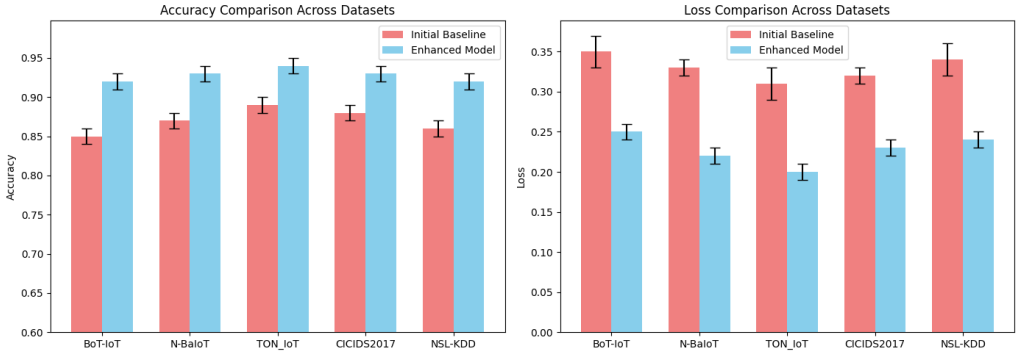

Fig. 4. Overall comparisons between the Initial and the Enhanced Models across all datasets.

REFERENCES

[1] Z. Almesleh, A. Gouissem and R. Hamila, "Federated Learning with Kalman Filter for Intrusion Detection in IoT Environment," 2024 IEEE 8th Energy Conference (ENERGYCON), Doha, Qatar, 2024, pp. 1-6, doi: 10.1109/ENERGYCON58629.2024.10488796.

[2] Salunkhe UR, Mali SN. Security enrichment in intrusion detection system using classifier ensemble. Journal of Electrical and Computer Engineering. 2017;2017(1):1794849.

[3] Vengatesan K, Kumar A, Naik R, Verma DK. Anomaly based novel intrusion detection system for network traffic reduction. In2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), 2018 2nd International Conference on 2018 Aug 30 (pp. 688-690). IEEE.

[4] T. Ohtani, R. Yamamoto and S. Ohzahata, "Detecting Zero-Day Attack with Federated Learning Using Autonomously Extracted Anomalies in IoT," 2024 IEEE 21st Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 2024, pp. 356-359, doi: 10.1109/CCNC51664.2024.10454669.

[5] Khraisat A, Gondal I, Vamplew P, Kamruzzaman J. Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity. 2019 Dec;2(1):1-22.

[6] Mohammad RM, Alsmadi MK, Almarashdeh I, Alzaqebah M. An improved rule induction based denial of service attacks classification model. Computers & Security. 2020 Dec 1;99:102008.

[7] Muneer S, Farooq U, Athar A, Ahsan Raza M, Ghazal TM, Sakib S. A Critical Review of Artificial Intelligence Based Approaches in Intrusion Detection: A Comprehensive Analysis. Journal of Engineering. 2024;2024(1):3909173.

[8] Saied M, Guirguis S, Madbouly M. Review of artificial intelligence for enhancing intrusion detection in the internet of things. Engineering Applications of Artificial Intelligence. 2024 Jan 1;127:107231.

[9] Alsmadi MK, Mohammad RM, Alzaqebah M, Jawarneh S, AlShaikh M, Al Smadi A, Alghamdi FA, Alqurni JS, Alfagham H. Intrusion Detection Using an Improved Cuckoo Search Optimization Algorithm.

[10] Latif S, Boulila W, Koubaa A, Zou Z, Ahmad J. Dtl-ids: An optimized intrusion detection framework using deep transfer learning and genetic algorithm. Journal of Network and Computer Applications. 2024 Jan 1;221:103784.

[11] Zhu J, Liu X. An integrated intrusion detection framework based on subspace clustering and ensemble learning. Computers and Electrical Engineering. 2024 Apr 1;115:109113.

[12] Zhang C, Xie Y, Bai H, Yu B, Li W, Gao Y. A survey on federated learning. Knowledge-Based Systems. 2021 Mar 15;216:106775.

[13] Li L, Fan Y, Tse M, Lin KY. A review of applications in federated learning. Computers & Industrial Engineering. 2020 Nov 1;149:106854.

[14] Khan LU, Saad W, Han Z, Hossain E, Hong CS. Federated learning for internet of things: Recent advances, taxonomy, and open challenges. IEEE Communications Surveys & Tutorials. 2021 Jun 18;23(3):1759-99.

[15] M. Umair, W. -H. Tan and Y. -L. Foo, "Dynamic Federated Learning Aggregation for Enhanced Intrusion Detection in IoT Attacks," 2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Osaka, Japan, 2024, pp. 524-529, doi: 10.1109/ICAIIC60209.2024.10463247.

[16] Iman M, Arabnia HR, Rasheed K. A review of deep transfer learning and recent advancements. Technologies. 2023 Mar 14;11(2):40.

[17] Weiss K, Khoshgoftaar TM, Wang D. A survey of transfer learning. Journal of Big data. 2016 Dec;3:1-40.

[18] A. U. Karimy and P. C. Reddy, "Analyzing Federated Learning as a novel approach for enhancing security and privacy in the Internet of Things (IoT)," 2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2024, pp. 1-7, doi: 10.1109/ICAECT60202.2024.10468686.

[19] Y. Luan, "Network Traffic Anomaly Detection Based on Federated Learning," 2024 4th International Conference on Neural Networks, Information and Communication Engineering (NNICE), Guangzhou, China, 2024, pp. 224-228, doi: 10.1109/NNICE61279.2024.10498908.

[20] M. H. Bhavsar, Y. B. Bekele, K. Roy, J. C. Kelly and D. Limbrick, "FL-IDS: Federated Learning-Based Intrusion Detection System Using Edge Devices for Transportation IoT," in IEEE Access, vol. 12, pp. 52215-52226, 2024, doi: 10.1109/ACCESS.2024.3386631.

[21] H. Babbar and S. Rani, "FRHIDS: Federated Learning Recommender Hybrid Intrusion Detection System Model in Software-Defined Networking for Consumer Devices," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 2492-2499, Feb. 2024, doi: 10.1109/TCE.2023.3329151.

[22] A. Raj, V. Sharma, S. Rani, A. K. Shanu and N. Kumar, "Strengthening the Security of IoT Devices Through Federated Learning: A Comprehensive Study," 2024 11th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2024, pp. 1-5, doi: 10.1109/ICRITO61523.2024.10522388.

[23] M. Al-Hawawreh and M. S. Hossain, "Federated Learning-Assisted Distributed Intrusion Detection Using Mesh Satellite Nets for Autonomous Vehicle Protection," in IEEE Transactions on Consumer Electronics, vol. 70, no. 1, pp. 854-862, Feb. 2024, doi: 10.1109/TCE.2023.3318727.

[24] Andresini G, Appice A, De Rose L, Malerba D. GAN augmentation to deal with imbalance in imaging-based intrusion detection. Future Generation Computer Systems. 2021 Oct 1;123:108-27.

[25] Elreedy, D.; Atiya, A.F. A comprehensive analysis of synthetic minority oversampling technique (SMOTE) for handling class imbalance. Inf. Sci. 2019, 505, 32–64.

[26] Mohammad R, Saeed F, Almazroi AA, Alsubaei FS, Almazroi AA. Enhancing Intrusion Detection Systems Using a Deep Learning and Data Augmentation Approach. Systems. 2024 Mar 1;12(3):79.

[27] BoT-IoT Dataset. (2018). Retrieved from https://research.unsw.edu.au/projects/bot-iot-dataset

[28] Kashif, M. (2019). N-BaIoT Dataset. Kaggle. Retrieved from https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset/code

[29] TON_IoT Dataset. (2020). Retrieved from https://research.unsw.edu.au/projects/ton-iot-datasets

[30] Huhn, C. (2017). CICIDS 2017. Kaggle. Retrieved from https://www.kaggle.com/datasets/chethuhn/network-intrusion-dataset

[31] Hassan, M. (2019). NSL-KDD Dataset. Kaggle. Retrieved from https://www.kaggle.com/datasets/hassan06/nslkdd