Dynamic Federated Learning Aggregation for Enhanced Intrusion Detection in IoT Attacks

Muhammad Umair
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
1221400084@student.mmu.edu.my

Wooi-Haw Tan
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
twhaw@mmu.edu.my

Yee-Loo Foo*
Faculty of Engineering
Multimedia University
Cyberjaya, Malaysia
ylfoo@mmu.edu.my

***Abstract*—The widespread integration of Internet of Things (IoT) devices has elevated security risks, specifically through vulnerabilities exploited by Botnet attacks. The emergence of these attacks highlights the necessity for resilient intrusion detection systems to detect such security threats. Prior methods employed a conventional centralized approach, involving the collection of data followed by training a machine learning or deep learning model. However, the conventional methods prove impractical for enterprise entities, as they are reluctant to share their sensitive data in a centralized environment due to privacy concerns. In this study, we utilize a decentralized Federated Learning (FL) approach to detect such Botnet attacks (i.e., Bruteforce, DDoS, DoS, Infiltration and Bot attacks). Our proposed FL method incorporates dynamic aggregation, facilitates the aggregation of updated models at local clients, thereby addressing the limitations associated with centralized data handling and privacy concerns. Our results show that the proposed method has achieved 87.98% accuracy for Botnet attacks and 100% accuracy for DoS and DDoS attacks.**

Keywords—federated learning, dynamic aggregation intrusion detection

I. Introduction

In the dynamic landscape of Artificial Intelligence (AI), the advent of decentralized paradigms has catalysed a paradigm shift, redefining the boundaries of collaborative model development [1, 2]. Among these groundbreaking approaches, Federated Learning (FL) has emerged as a pioneering framework, offering a transformative methodology for the collective advancement of machine learning (ML) and deep learning (DL) models. The concept of FL was introduced by researchers at Google [3], characterized as a distributed and decentralized framework designed to simultaneously accomplish data expansion and privacy protection.

By harnessing data from diverse and distributed sources, FL revolutionizes the conventional paradigm of centralized model training [3]. In contrast to traditional methodologies that require consolidating data into a centralized repository [4], FL enables devices to undertake localized model training, fostering a collaborative approach to model development [3]. This collaborative framework not only amplifies the scalability and efficiency of model training but also upholds the principles of data ownership and privacy, addressing critical concerns in the current data-centric landscape [4].

Within this decentralized framework, the local device often referred as client, seamlessly integrate with the overarching architecture of the DL model employed on the cloud centre server. This integration facilitates the local training of models on each respective device, ensuring a cohesive and synchronized approach to model development across the entire FL network [3].

Through an ongoing communication channel between the server and multiple client within a FL network, the client transmit the continually updated and trained file of the model. This continuous exchange of information ensures a seamless flow of insights and knowledge across the FL network, allowing each client to contribute to the collective enhancement of the global model [5].

Diverging from the conventional approach of centralized learning, FL only transmits the model updates during the communication between clients and the server, as opposed to transmitting the data of each individual client. This distinctive feature allows FL to indirectly augment the available data while mitigating the risk of original data leakage [6].

A simplified visualization of FL network has been given in Fig. 1, this figure represents the main components i.e., server which contains the global model architecture, clients which are considered as local devices in this case which contains local dataset in their memory, and the two-step methodology, i.e., 1 and 2, so the number 1 represents that server is sending some update to the network whereas, the number 2 represents that an update is being sent to the server.

Fig. 1, provides a simplified representation of the FL network, illustrating its primary components. The diagram includes the Server i.e., hosting the architecture of the global model, and client each maintaining a local dataset in their memory. And is a local model the architecture of the model is same as global model. Additionally, the figure delineates a two-step methodology, where step 1 signifies the server sending updates to the network, while step 2 represents the transmission of updates from the clients to the server.

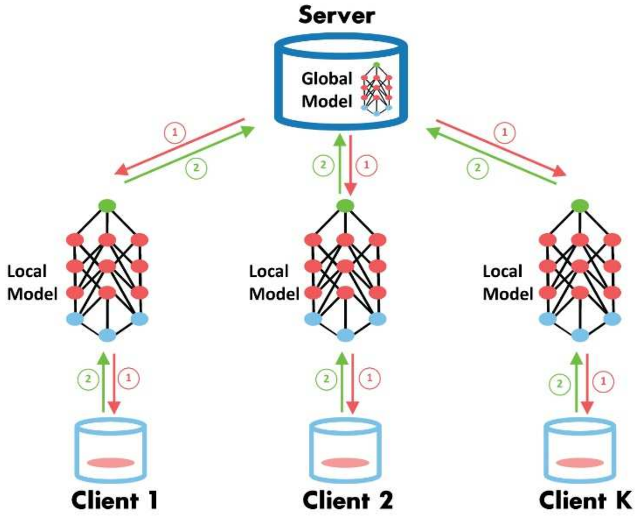

Fig. 1. Simplified representation of the federated learning network.

This paper delves into the application of FL in the context of intrusion detection. By harnessing the collective intelligence of distributed virtual clients, FL not only elevates the accuracy of intrusion detection model but does so while respecting the inherent privacy of individual data contributors. The paper explores how this decentralized approach mitigates concerns related to data security and able to classify these attacks which often occurs in IoT devices.

A. Focus of this study

In this study, our primary focus revolves around the implementation of a FL approach tailored for intrusion detection in Internet of Things (IoT) devices. To enhance the effectiveness of the FL framework, we employed a dynamic aggregation strategy, allowing for the collaborative training with a shared global model. The intricacies of intrusion detection were further addressed through the integration of a random forest model, serving as an importance feature extraction technique. This approach enabled the identification and prioritization of relevant features crucial for detecting attacks. For the overarching global model, we utilized a simplified architecture, containing convolutional neural network (CNN) layer in the model along with dense layers.

This paper comprises as follows: Section-II describes about the literature review, Section-III describes the data preparation and processing adobted for this study, Section-IV describes the proposed FL methodology. Section-V shows the results obtained and their discussion and Section-VI summarizes and concludes this study.

II. Literature review

The contemporary landscape of network security has witnessed a surge in the adoption of ML and DL methodologies for the formulation of robust detection models tailored for network intrusion detection [7, 8].,Traditional models necessitate the consolidation of extensive datasets in a central repository, a practice that raises concerns regarding data privacy, security, and ownership [9]. The inherently centralized nature of these approaches poses challenges in instances where data diversity is paramount, such as in the realm of FL for network intrusion detection.

The authors of reference [7] presented a study for intrusion detection system for software defined network using ML. They proposed a novel technique HFS-LGBM, the Hybrid Feature Selection (HFS) approach synergizes the benefits of correlation-based feature selection and random forest Recursive Feature Elimination. A static software network is constructed using Mininet, and the proposed system is evaluated using the NSL-KDD dataset. The experimental results demonstrate that HFS-LGBM outperforms other algorithms by achieving an accuracy of 98.72%.

The study of reference [10] introduces an innovative Intrusion Detection System (IDS) for IoT-Cloud environments, incorporating deep neural networks and swarm intelligence techniques. The approach utilizes conventional neural networks (CNNs) to extract important features from dataset, It employs the Capuchin Search Algorithm (CapSA) for efficient feature selection, enhancing classification accuracy.

In another work [11], the authors presented an innovative approach for Network Intrusion Detection Systems (NIDS) involving a weighted naive Bayes model. This model integrates particle swarm optimization and rough set theory to augment the system's detection capabilities. The collaborative application of these methodologies results in enhanced performance for identifying and mitigating intrusion events. Additionally, in [12], binary logistic regression statistical tools were employed for attack detection. This strategy involved extracting crucial data from the routing layer and scrutinizing sensor behaviour within the NIDS framework. The adoption of binary logistic regression aimed to elevate the accuracy and overall effectiveness of the detection mechanism.

In another research [13], the authors presented a NIDS by integrating a DL classification model with non-symmetric deep autoencoder and random forest classification algorithms. Their method demonstrated high accuracy when applied to the NSL-KDD and KDD Cup '99 datasets. Another study [14] proposed an NIDS model combining multivariate correlation analysis with Long Short-Term Memory (LSTM) networks. The multivariate correlation analysis algorithm was used for feature extraction, while the LSTM component handled intrusion classification. Moreover, in [15] presented an NIDS using adaptive composition oversampling technology and LightGBM. This innovative approach addressed unbalanced network intrusion data, leading to enhanced detection performance while reducing time complexity.

Presently, research endeavours are dedicated to integrating FL into the realm of NIDS. In [16], the authors introduced MT-DNN-FL, a multitask FL model aiming to tackle diverse tasks such as traffic recognition, traffic classification and abnormal traffic detection. This model emphasizes the utilization of FL for addressing various tasks within the context of NIDS. Furthermore, researchers in [17] provided an in-depth exploration of a federated deep learning model uniquely designed for industrial cyber-physical systems. The primary objective is to enhance the system's capability to detect potential cyber-attacks, showcasing the adaptability of FL in different domains of intrusion detection.

The authors of [18] introduced an adaptive NIDS that harnesses piecewise FL. This inventive method facilitates the training of multiple analogous networks within a shared global model, allowing numerous participants to collaborate and exchange various global models. In a separate investigation [19], researchers proposed an intelligent intrusion detection model incorporating LSTM within a FL framework. This sophisticated model demonstrated heightened precision and improved consistency in contrast to conventional approaches. Additionally, the developers of FedACNN, as highlighted in [20], introduced a method applied to network intrusion detection. This approach efficiently aggregates local models by assessing the Euclidean distance between the global and local models from the preceding round. Through this process, FedACNN significantly reduces the number of communication rounds by 50% compared to Fedavg, marking a promising advancement in the domain.

Meryem et al. [21] used CSE-CIC-IDS2018 [22] dataset and proposed Fed-ANIDS approach. They achieved remarkable accuracies of more than 90% results showcasing their effectiveness of approach. However, for an unseen dataset they achieved 68% of accuracy. Faisal et al. [23] used CSE-CIC-IDS2018 using FedAvg algorithm, they achieved 90% accuracy on unseen dataset. They used FedAvg algorithm to average the weights, however, their methodology got the limitations for dynamic aggregation of the process during on-going round.

III. Data preparation

A. Dataset

This study harnessed a meticulously structured dataset i.e., CSE-CIC-IDS2018 [22], the dataset is organized on a daily basis, capturing raw data such as network traffic in Pcap format and event logs from both Windows and Ubuntu systems for each day. Utilizing the CICFlowMeter-V3, they performed feature extraction, resulting in the derivation of over 80 distinctive traffic-related features. These features were then consolidated and stored in CSV files, each corresponding to a specific machine in the dataset. The dataset comprehensively encompasses seven distinct Botnet attacks scenarios: Bruteforce, Botnet, Denial of Service (DoS), Distributed Denial of Service (DDoS), Web attacks, and network infiltration. The attacking infrastructure involves 50 machines, while the victim organization is structured into five departments, housing a total of 420 machines and 30 servers. This dataset offers a thorough perspective, encompassing captured network traffic and system logs for each machine, along with the features extracted using the CICFlowMeter-V3. The data distribution with respect to each label has been shown in pie chart in Fig. 2, This figure shows that data was quite unbalanced, furthermore the samples for web attack labels are very less, so for this study we didn’t take the web attack samples.

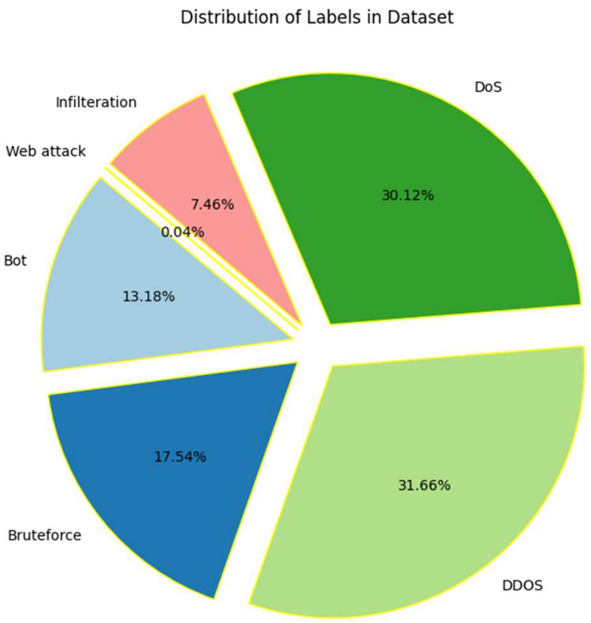

Fig. 2. Simplified representation of the federated learning network.

B. Data preprocessing

In preprocessing, the dataset underwent a meticulous partitioning into training and testing sets, adhering to an 80:20 ratio. To bolster the model's proficiency in processing categorical attack types, a judicious application of label encoding was implemented, effecting the conversion of such types into numerical representations. Furthermore, to ensure uniformity and standardization of the feature values, we employed feature scaling techniques, utilizing the min-max scaler. This meticulous approach contributed to the normalization of feature values and the establishment of a standardized range across the dataset.

C. Feature importance analysis

We used random forest model for feature importance analysis. This approach enables us to discern the most influential columns, providing valuable insights into the underlying patterns of network intrusions. Subsequently, we select the top 20 important columns, distributing them among clients for local model training purposes. Fig. 3, represents the bar chart and importance score of top columns that was done by random forest model.

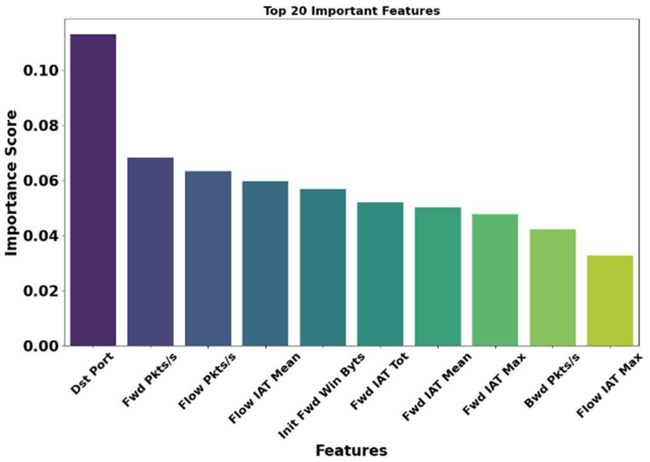

Fig. 3. Top 20 important features extracted from random forest model.

IV. Proposed federated learning approach

A. Overview of the proposed FL methodology

An illustrative representation of the adopted methodology is presented in Fig. 4, encompassing five pivotal stages. The initial stage involves Data Preprocessing, wherein data cleaning techniques are applied to address unnecessary and null values in the CSE-CIC-IDS2018 [22] dataset. Concurrently, data standardization is performed, and categorical labels are transformed into numerical values. The subsequent stage revolves around Feature Selection, a critical step given the original dataset 80 columns. To mitigate potential memory issues, the top 20 most important columns are selected using a random forest model. This curated dataset is then divided into training and test sets at an 80:20 ratio. The third stage focuses on Data Distribution, leveraging the IID technique to allocate data among three distinct clients.

Moving forward, the fourth stage introduces the implementation of the proposed dynamic aggregation FL methodology. This involves the utilization of a dynamic aggregator, responsible for aggregating local model updates upon receiving an updated file from the respective client. Finally, the last stage pertains to model evaluation, where the updated global model file from the dynamic aggregator is employed to assess the test set. Evaluation metrics such as f1-score, precision, recall, and accuracy are computed to gauge the performance of the dynamic FL approach.

B. Dynamic aggregator

The dynamic aggregator serves as a pivotal component in our proposed FL methodology, orchestrating the aggregation of local model updates from participating clients in a dynamic and adaptive manner. Unlike static aggregation methods, the dynamic aggregator responds promptly to incoming updates, ensuring real-time integration of valuable insights from the distributed clients. The agility of the dynamic aggregator enhances the convergence speed of the global model and

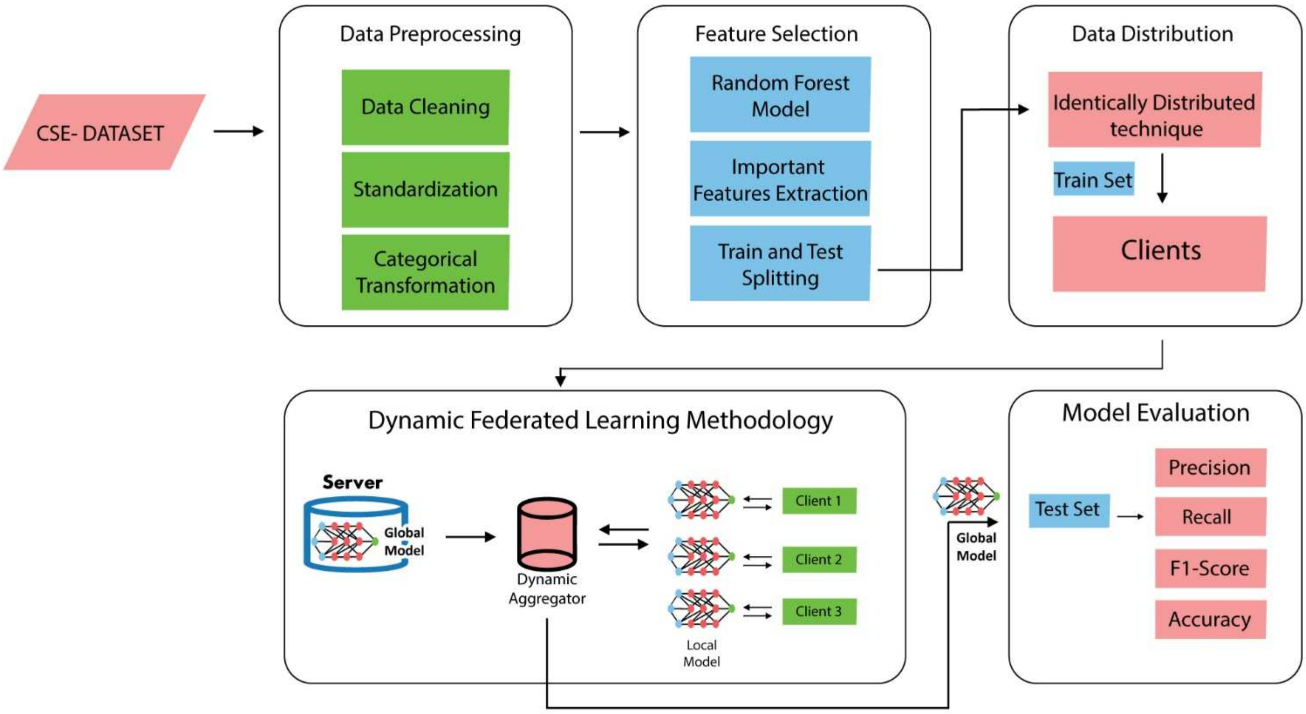

Fig. 4. Overview of utilized methodology for this study

adapts to the dynamic nature of the distributed learning environment. Mathematically, the dynamic aggregation process can be expressed as in (1):

$$\mathcal{G}_i = \alpha \times \mathcal{G}_{i-1} + (1-\alpha) \times \ell_i \qquad (1)$$

Where, $\mathcal{G}_i$ represents the global model at iteration i, ℓ_i denotes the local model update received from a client at iteration i. And α is the dynamic aggregation parameter, adapting to the characteristics of the incoming updates from local models.

The dynamic aggregator employs an adaptive mechanism for α to balance the contributions of the existing global model and the incoming local model update. This adaptability ensures that more weight is assigned to the local update when it contains crucial information, while also preserving the knowledge encoded in the current global model. The dynamic aggregator responsiveness and adaptability contribute to the efficiency and effectiveness of the FL process.

V. Experiment and Results

A. Experimental Settings

This study employed the Python programming language, specifically version 3.10.9. Additionally, we utilized the Tensorflow framework with the integration of Keras libraries. google colab notebook served as the platform for model training and coding activities.

B. Independent and identically distributed technique for data distribution among clients

To ensure the fair distribution of data among clients in a FL framework, the independent and identically distributed (IID) technique is employed. This technique assumes that the data samples are drawn independently from the same probability distribution. Mathematically in (2), let ∂ be the dataset, and ∂_i represents an individual data sample. The IID assumption implies that the probability of observing a specific data sample ∂_i is the same for all samples and follows a common probability distribution:

$$\mathcal{P}_{\partial_1,\partial_2,\dots,\partial_n} = \mathcal{P}_{\partial_1} \times \mathcal{P}_{\partial_2} \times \dots \times \mathcal{P}_{\partial_n} \qquad (2)$$

In the context of FL, this means that each client receives a subset of the data, and these subsets are representative of the overall data distribution. By adhering to the IID assumption, the training process on local data at each client contributes meaningfully to the development of a robust global model. This foundational principle helps maintain consistency and fairness in the learning process across diverse and decentralized data sources within the FL network.

The primary aim is to cultivate a cohesive global model by harnessing locally collected data from clients. This learning process unfolds at the individual client, involving the transmission of periodic intermediate model updates to the server. The main objective is to minimize the loss or error at each client during the model training phase, thereby optimizing the acquired model. An integral pursuit of FL is the attainment of the minimum loss i.e. ∂_a. This mathematical representation is given in (3) that encapsulates the core aspiration of FL, emphasizing the optimization of the learned model through the reduction of loss during the training process.

$$\min_a \partial, \; where \; \partial_a = \sum_{w=1}^{c} \rho_w \, \partial_a \qquad (3)$$

Where, c represents the participating clients in FL network, ∂_a represents the local loss function for respective client, $\rho_w \geq 0$ signifies the efficacy of the corresponding client.

C. Global model

The architecture of the global model employed for intrusion detection is outlined in Table I. The model follows a CNN design, consisting of Conv2D layers with rectified linear unit (ReLU) activation functions. The MaxPooling2D layer is incorporated to down-sample the spatial dimensions, followed by a Flatten layer to transform the data into a one-dimensional array. A densely connected layer with 64 units and a dropout layer for regularization precedes the final output layer, which consists of seven units corresponding to the different intrusion classes. The activation functions, such as ReLU and softmax, are strategically employed to introduce non-linearity and enable multi-class classification. The model total trainable parameters sum up to 3587, optimizing its capacity to learn patterns within the input data. This architecture is pivotal for the collaborative learning process in the FL framework. Table II., shows the total parameters and size of the model.

TABLE I. GLOBAL MODEL ARCHITECUTRE.

Layer	Output shape	Parameters
Conv 2D	(5 , 6 , 32)	672
Activation (ReLU)	(5 , 6 , 32)	0
Max Pooling 2D	(3 , 4 , 32)	0
Flatten	(384)	0
Dense	(64)	2460
Dropout	(64)	0
Dense	(6)	455
Activation (Softmax)	(6)	0

TABLE II. GLOBAL MODEL PARAMETERS.

	Parameters	Size in KB
Total Parameters	3587	100.65
Trainable parameters	3587	100.65
Non-trainable parameters	0	0

D. *Model evaluation*

In this study, we have employed key evaluation metrics, namely accuracy, precision, recall, and f1-score, to assess the performance of the proposed intrusion detection system. These metrics are defined as follows: accuracy is calculated as the ratio of correctly classified instances (true positives and true negatives) to the total number of instances. Precision is determined as the proportion of true positives to the total number of instances predicted as positive. Recall, also known as sensitivity or true positive rate, represents the ratio of true positives to the total number of actual positive instances. f1-score is the harmonic mean of precision and recall, providing a balanced measure between precision and recall. These metrics offer valuable insights into the effectiveness of the proposed intrusion detection system.

As illustrated in Fig. 5, the confusion matrix on the test set has been obtained using the trained global model file. The test serves as an evaluation phase, revealing instances where the model predicts inaccurate results, placing them in the wrong class. This observed discrepancy underscores the imperfections in classification, emphasizing the existence of misclassified samples. Nevertheless, it is crucial to note that in our proposed FL system, the data is intentionally maintained on the client end in a decentralized manner. This decentralized approach aligns with the core philosophy of FL, preserving data sovereignty and privacy on individual client devices, even when facing challenges associated with misclassifications during the evaluation process. Table III contains the classification report on global model.

TABLE III. CLASSIFICATION REPORT.

Label	Precision (%)	Recall	F1-score
Benign	91.09	89.45	0.90
Bot	1.00	1.00	1.00
Bruteforce	66.72	88.54	0.75
DDoS	1.00	1.00	1.00
DoS	80.24	63.82	0.63
Infilteration	89.34	90.10	0.90
Accuracy	87.98		

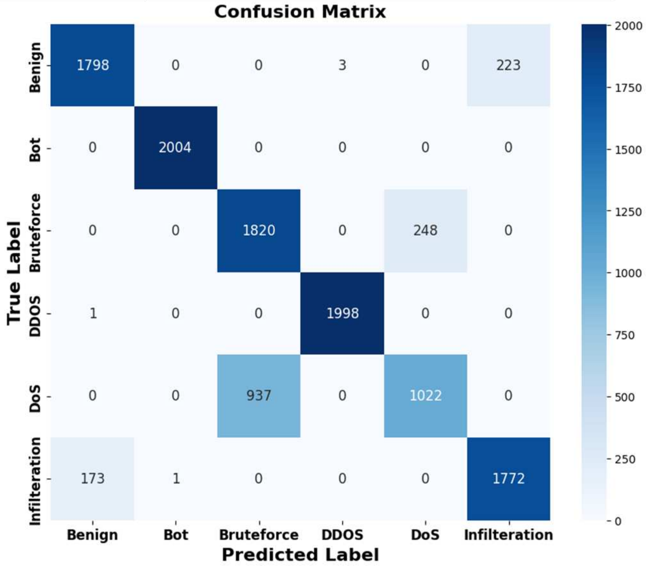

Fig. 5. Confusion metrix on test set using global model.

The results given in Table III, presents the classification report, demonstrate the robust performance of the proposed system across multiple intrusion categories. The precision values indicate high accuracy in positive predictions for each class, with particularly noteworthy precision scores for the 'Bot' and 'DDoS' classes, reaching 100%. The recall metrics highlight the correctly identified instances of each class, emphasizing its strength in categories such as 'Bot,' 'DDoS,' and 'Infiltration.' The f1-score, a harmonized measure of precision and recall, further confirms the balanced effectiveness of the system, with commendable scores across various intrusion types. The accuracy achieved 87%, showcasing its capability to correctly classify instances from the dataset. These results affirm the effectiveness of the proposed dynamic aggregation FL approach in intrusion detection for IoT devices, utilizing a feature extraction technique based on the random forest model and a global model architecture components. Furthermore, we done compare our proposed study with the previous studies those utilized same dataset and FL approaches, the data has been given in Table IV.

TABLE IV. GLOBAL MODEL PARAMETERS.

Pervious Studies	Dataset	Approach	Dynamic Aggregation	Accuracy
Meryem et al. [21]	CSE-CIC-IDS2018	Fed-ANIDS	No	Unseen dataset: 68.40%
Faisal et al. [23]	CSE-CIC-IDS2018	FedAvg	No	Accuracy 90%
Our Proposed	CSE-CIC-IDS2018	Dynamic FL appraoch	Yes	Accuracy 87.98%

VI. Conclusion

The focus of this study lies in utilizing FL for malware classification, specifically in the domain of intrusion detection for Internet of Things (IoT) devices. By utilizng DL model, we implemented a decentralized approach through the implementation of FL. The proposed dynamic aggregation FL approach incorporates importance feature extraction techniques using random forest model for local training and a global model architecture. The employed dataset encompasses diverse Botnet attacks, including Bruteforce, Bot attacks, DoS, DDoS and infiltration. Through data preprocessing, feature selection, and dynamic FL methodology implementation, we aimed to enhance the efficiency of intrusion detection. Furthermore, evaluation metrics such as accuracy, precision, recall, and f1-score were employed to assess the proposed system's performance. The classification report revealed promising results, with precision of 100% 100% for DoS and DDoS, and f1-score reflecting the model's effectiveness across different attack scenarios. Furthermore, we achieved an overall accuracy of 87.98% for the six different attacks. The findings underscore the potential of FL as a resilient and privacy-preserving paradigm for enhancing cybersecurity in the face of evolving threats.

Acknowledgment

The authors would like to acknowledge the support of the Fundamental Research Grant Scheme (FRGS) FRGS/1/2021/ICT08/MMU/03/1, Ministry of Higher Education, Malaysia.

References

[1] V. Subbiah, "The next generation of evidence-based medicine," *Nature Medicine,* vol. 29, no. 1, pp. 49-58, 2023.

[2] A. Rauniyar *et al.*, "Federated Learning for Medical Applications: A Taxonomy, Current Trends, Challenges, and Future Research Directions," *IEEE Internet of Things Journal,* pp. 1-1, 2023.

[3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," presented at the Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Proceedings of Machine Learning Research, 2017.

[4] S. Abdulrahman, H. Tout, H. Ould-Slimane, A. Mourad, C. Talhi, and M. Guizani, "A Survey on Federated Learning: The Journey From Centralized to Distributed On-Site Learning and Beyond," *IEEE Internet of Things Journal,* vol. 8, no. 7, pp. 5476-5497, 2021.

[5] M. Kamp *et al.*, "Efficient Decentralized Deep Learning by Dynamic Model Averaging," in *Machine Learning and Knowledge Discovery in Databases*, Cham, 2019, pp. 393-409: Springer International Publishing.

[6] W. Y. B. Lim *et al.*, "Federated Learning in Mobile Edge Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials,* vol. 22, no. 3, pp. 2031-2063, 2020.

[7] G. Logeswari, S. Bose, and T. Anitha, "An intrusion detection system for sdn using machine learning," *Intelligent Automation & Soft Computing,* vol. 35, no. 1, pp. 867-880, 2023.

[8] D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, "Intrusion Detection System Using Feature Extraction with Machine Learning Algorithms in IoT," vol. 12, no. 2, p. 29, 2023.

[9] G. Drainakis, P. Pantazopoulos, K. V. Katsaros, V. Sourlas, A. Amditis, and D. I. Kaklamani, "From centralized to Federated Learning: Exploring performance and end-to-end resource consumption," *Computer Networks,* vol. 225, p. 109657, 2023.

[10] M. Abd Elaziz, M. A. A. Al-qaness, A. Dahou, R. A. Ibrahim, and A. A. A. El-Latif, "Intrusion detection approach for cloud and IoT environments using deep learning and Capuchin Search Algorithm," *Advances in Engineering Software,* vol. 176, p. 103402, 2023.

[11] X. Ren, W. Jiao, and D. Zhou, "Intrusion detection model of weighted navie bayes based on particle swarm optimization algorithm," *Computer Engineering Applications,* vol. 52, no. 7, pp. 122-126, 2016.

[12] Y. Liu and D. Pi, "A Novel Kernel SVM Algorithm with Game Theory for Network Intrusion Detection," *KSII Transactions on Internet Information Systems,* vol. 11, no. 8, 2017.

[13] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," *IEEE Transactions on Emerging Topics in Computational Intelligence,* vol. 2, no. 1, pp. 41-50, 2018.

[14] R.-H. Dong, X.-Y. Li, Q.-Y. Zhang, and H. Yuan, "Network intrusion detection model based on multivariate correlation analysis – long short-time memory network," vol. 14, no. 2, pp. 166-174, 2020.

[15] J. Liu, Y. Gao, and F. Hu, "A fast network intrusion detection system using adaptive synthetic oversampling and LightGBM," *Computers & Security,* vol. 106, p. 102289, 2021.

[16] Y. Zhao, J. Chen, D. Wu, J. Teng, and S. Yu, "Multi-Task Network Anomaly Detection using Federated Learning," presented at the Proceedings of the 10th International Symposium on Information and Communication Technology, Hanoi, Ha Long Bay, Viet Nam, 2019. Available: https://doi.org/10.1145/3368926.3369705

[17] B. Li, Y. Wu, J. Song, R. Lu, T. Li, and L. Zhao, "DeepFed: Federated Deep Learning for Intrusion Detection in Industrial Cyber–Physical Systems," *IEEE Transactions on Industrial Informatics,* vol. 17, no. 8, pp. 5615-5624, 2021.

[18] Y. Sun, H. Esaki, and H. Ochiai, "Adaptive Intrusion Detection in the Networking of Large-Scale LANs With Segmented Federated Learning," *IEEE Open Journal of the Communications Society,* vol. 2, pp. 102-112, 2021.

[19] R. Zhao, Y. Yin, Y. Shi, and Z. Xue, "Intelligent intrusion detection based on federated learning aided long short-term memory," *Physical Communication,* vol. 42, p. 101157, 2020.

[20] D. Man, F. Zeng, W. Yang, M. Yu, J. Lv, and Y. Wang, "Intelligent Intrusion Detection Based on Federated Learning for Edge-Assisted Internet of Things," *Security and Communication Networks,* vol. 2021, p. 9361348, 2021.

[21] M. J. Idrissi *et al.*, "Fed-ANIDS: Federated learning for anomaly-based network intrusion detection systems," *Expert Systems with Applications,* vol. 234, p. 121000, 2023.

[22] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in *International Conference on Information Systems Security and Privacy*, 2018.

[23] F. Naeem, M. Ali, and G. Kaddoum, "Federated-Learning-Empowered Semi-Supervised Active Learning Framework for Intrusion Detection in ZSM," *IEEE Communications Magazine,* vol. 61, no. 2, pp. 88-94, 2023.