

Article

A Federated Learning-Based Network Intrusion Detection System for 5G and IoT Using Mixture of Experts

Loukas Ilias ^{*}, George Doukas , Vangelis Lamprou , Spiros Mouzakitis, Christos Ntanos 
and Dimitris Askounis

Decision Support Systems Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Iroon Polytechniou 9, 15780 Athens, Greece; gdoukas@epu.ntua.gr (G.D.); vlamprou@epu.ntua.gr (V.L.); smouzakitis@epu.ntua.gr (S.M.); cntanos@epu.ntua.gr (C.N.); askous@epu.ntua.gr (D.A.)

* Correspondence: lilias@epu.ntua.gr

Abstract

Fifth generation (5G) networks have significantly enhanced connectivity, speed, and reliability, transforming industries with faster and more efficient communication. The Internet of Things (IoT) has introduced unprecedented convenience and automation, revolutionizing sectors such as healthcare, finance, and smart infrastructure. However, both 5G networks and IoT environments are experiencing a high frequency of attacks. Intrusion detection systems (IDSs) built on federated learning (FL) are being proposed to boost data privacy and security. However, these IDSs are related with the inherent drawbacks of FL, namely the existence of non-independently and identically (non-IID) distributed features and the machine learning model complexity. To address these limitations, we present a study that integrates a Mixture of Experts (MoE) into an FL setting in the task of intrusion detection. Specifically, to mitigate the issues of model complexity within the FL setting, we use a sparsely gated MoE layer consisting of a router/gating network and a set of experts. Only a subset of experts is selected via applying noisy top- k gating. To alleviate the issue of non-IID data, we adopt the Label-based Dirichlet Partition method, utilizing Dirichlet sampling with a hyperparameter α to simulate a label-based non-IID data distribution. Four FL strategies are employed. We perform our experiments on the 5G-NIDD and BoT-IoT datasets. Findings show that the proposed approach achieves competitive performance across both datasets under heterogeneous federated settings.

Keywords: intrusion detection; 5G networks; Internet of Things; federated learning; non-IID data; mixture of experts



Academic Editor: Aryya Gangopadhyay

Received: 7 February 2026

Revised: 27 February 2026

Accepted: 1 March 2026

Published: 3 March 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The rapid advancement of communication technologies has led to the widespread adoption of fifth generation (5G) wireless networks across numerous domains, delivering substantial improvements in connectivity, speed, and reliability. While these enhancements offer transformative benefits, the exponential growth of interconnected devices—including those within the Internet of Things (IoT), critical infrastructure such as healthcare systems [1], and large-scale financial networks—has introduced significant security and privacy challenges [2]. Specifically, IoT has revolutionized daily life, bringing unprecedented levels of efficiency and convenience [3]. SonicWall's (<https://www.sonicwall.com/blog/sonicwall-2024-mid-year-cyber-threat-report->

[iot-madness-powershell-problems-and-more](#), accessed on 10 November 2025) mid-year report for 2024 underscores the increasing threat to IoT devices, revealing that they are now targeted by cyberattacks for an average of 52.8 h each week.

Designing a robust Intrusion Detection system (IDS) based on machine learning is of paramount importance [4]. Several research works have proposed IDS [5]. However, traditional IDSs are not able to capture the dynamic and complex characteristics of 5G networks and IoT environments, which are associated with the heterogeneity of devices, diverse communication protocols, and data formatting [6]. Additionally, low latency is required; thus, IDSs must be capable of identifying an attack as early as possible. However, traditional IDSs are not able to handle this vast magnitude of network traffic, while the diverse nature of the entities taking part in a 5G network or an IoT environment constitutes an additional barrier, since advanced mechanisms considering privacy concerns must be developed.

Integrating an IDS into a Federated learning (FL) framework is a promising solution to tackle the aforementioned challenges. Specifically, FL enables decentralized data processing and is essential in the 5G era, as it boosts data privacy and security by cutting down the need to transfer sensitive information and improving network operation and performance. Thus, data remains localized at each node during training, adaptability and detection capability of the whole system are improved, while communication cost is substantially reduced, since data are not transmitted through the network [7]. In an FL framework, aggregation strategies are employed to aggregate the local models from multiple clients and update the global model. An effective aggregation strategy enhances both the performance and the robustness of the global model. Despite the benefits of FL, the development of FL-based IDS in 5G and IoT environments is associated with challenges related to the existence of non-IID data among clients, head, and model complexity. Although FL-based approaches have been developed, existing studies largely overlook these challenges. For instance, although the work in [7] presents an FL-based approach utilizing the 5G-NIDD dataset for identifying attacks, it does not resolve the aforementioned challenges. Similarly, the authors in [8] divide the dataset manually into multiple subsets, where each subset is assigned a unique attack pattern. After that, each client trains the local model on its subset.

In this paper, we present a study utilizing Mixture of Experts (MoE) into an FL framework for recognizing attacks in 5G and IoT environments. Firstly, we utilize the Dirichlet label partitioning technique, which belongs to the data-partitioning strategies, in order to ensure that the effects of non-IID data in both 5G and IoT environments are alleviated. After that, we develop FL-based IDS, which integrate four different aggregation strategies, namely FedAvg [9], FedAvgM [10], FedProx [11], and FedTrimmedAvg [12]. To mitigate the issues of model complexity, a deep learning architecture is adopted that couples convolutional neural networks (CNNs) with a sparsely gated Mixture of Experts (MoE) layer. In terms of the MoE layer, a specific number of experts is selected based on the input data. Experiments are performed on both 5G and IoT environments utilizing the 5G-NIDD and BoT-IoT datasets, respectively.

Our main contributions are as follows:

- To the best of our knowledge, this is among the first studies to integrate MoE into an FL framework for identifying attacks into 5G and IoT environments.
- We employ the Dirichlet label partitioning technique to alleviate the impact of non-IID data.
- Our proposed approach mitigates the effects of the existence of non-IID data and model complexity.

2. Related Work

Several studies examine the significance and effectiveness of FL in cybersecurity, particularly in the context of intrusion detection.

A comprehensive survey [13] on the potential of FL for enhanced cybersecurity and trustworthiness in 5G and 6G networks has been published in [13]. Ref. [13] presents challenges in implementing FL in the context of 5G and 6G networks, which are related to communication bottlenecks, data heterogeneity, system heterogeneity, and ML model complexity.

A survey is presented in [14], where the authors describe in detail the benefits of FL for cybersecurity in the IoT environment. One of the challenges associated with FL for IoT [14] is related to the nonindependent and nonidentically distributed (non-IID) data. Specifically, in an IoT environment, data of a device might present differences due to changes in location, time, and users. Thus, the standard IID assumption is not adopted.

An additional challenge is pertinent to communication overhead and model complexity [15]. Similarly, the work in [16] describes the challenges associated with the development of FL-based intrusion detection mechanisms in IoT environments. Specifically, results of their study show that the existence of non-IID data distributions directly affects the effectiveness of security attack detection. The challenges involved in deploying FL-enabled IoT in IDS are related to the computational requirements of the deep learning approaches.

Moreover, the drawbacks and vulnerabilities of existing FL-based IDSs are discussed in [17]. Specifically, challenges are associated with the precise deployment of IDSs and the existence of non-IID data among others. With regard to the deployment of IDSs, the authors mention that deep learning models include an enormous number of parameters and hyperparameters.

The presence of non-IID data degrades classification performance [18] and impacts the federated training process by slowing down convergence [19]. To mitigate the effects of non-IID data, a feature fusion approach has been introduced in [20], namely FedFusion, where representations extracted from both the local and global models are aggregated.

FedLN [21] is also proposed as an approach for alleviating the effect of non-IID feature distributions. This is achieved by integrating a layer normalization into local models. Additionally, a review on the different techniques used to alleviate the issue of non-IID feature distributions is presented in [18]. Specifically, data-partitioning strategies are used for covering the typical non-IID data cases. The Dirichlet distribution [10,22] constitutes one of the techniques used.

In terms of the development of FL-based IDSs, which alleviate the issue of non-IID feature distributions, the related work is limited. Recently, the StatAvg method is proposed in [19] to mitigate the effects of non-IID data, where the authors present a method of producing global data statistics based on the local data statistics of FL clients. The authors perform their experiments on publicly available datasets in the IoT environment, namely TON-IoT (<https://research.unsw.edu.au/projects/toniot-datasets>, accessed on 10 November 2025) and CIC-IoT-2023 [23] datasets.

The work in [8] presents an FL framework utilizing IoT data. Three FL strategies are used, namely FedAvg, FedAvgM, and FedOpt. Three different models are compared, including artificial neural networks, LSTMs, and CNNs coupled with LSTMs. The authors experiment with different number of clients, ranging from 2 to 10 clients. The dataset is divided manually by the authors into multiple subsets, where each subset is assigned a unique attack pattern.

Recent IoT intrusion detection research has increasingly adopted adaptive neural architectures incorporating attention mechanisms to enhance feature representation. For example, the work in [24] proposes a TPE-based self-attention bidirectional LSTM model for IoT attack detection, where attention weights dynamically emphasize informative

temporal features. Such approaches reflect a broader trend toward input-dependent weighting mechanisms in Industrial IoT security. Although the proposed CNN-MoE framework differs architecturally, the gating network in sparsely gated MoE similarly performs input-conditioned weighting of expert outputs, enabling adaptive specialization across traffic patterns.

Our work differs from related work, since we present a study incorporating MoE into an FL setting in 5G and IoT environments. Our proposed approach alleviates the drawbacks of FL settings, which are related to (1) model complexity via MoE, where only a set of experts is activated, and (2) existence of non-IID data via the Dirichlet label partitioning technique.

3. Datasets

In this study, we have used the following datasets to perform our experiments:

- **5G-NIDD [25].** This dataset is created over a 5G wireless network environment of real mobile devices and consists of 1,215,890 unique data flows. The data are collected from two base stations, each containing an attacker node and multiple legitimate 5G users. The attacker nodes perform their attacks against a server located within the 5GTN MEC environment. The attack scenarios involve DoS/DDoS attacks and port scans. A description of the features used in this study is presented in Table A1 in Appendix A. Details about distribution of benign traffic and attacks are presented in Table 1.

Table 1. 5G-NIDD category distribution.

Category	Number of Instances
Benign	477,737
UDPFlood	457,340
HTTPFlood	140,812
SlowrateDoS	73,124
TCPConnectScan	20,052
SYNScan	20,043
UDPScan	15,906
SYNFlood	9721
ICMPFlood	1155

- **BoT-IoT [26,27].** This dataset contains botnet attack traffic generated with Internet of Things (IoT) networks. Specifically, the BoT-IoT dataset was generated in a testbed composed of virtual machines running different operating systems, along with network firewalls, network taps, the Node-red tool, and the Argus network security tool. A description of the features used in this study is described in Table A2 in Appendix B. The authors in [26] have released a subset of BoT-IoT, namely 5% of BoT-IoT. The full BoT-IoT dataset contains tens of millions of flows and requires substantial computational resources, particularly when training sparse Mixture-of-Experts architectures within a federated framework. For reproducibility and computational feasibility, we use the publicly released 5% subset, which is widely adopted in prior intrusion detection studies. Although the 5% subset reduces dataset size, it preserves the original traffic characteristics, allowing meaningful evaluation under heterogeneous federated conditions. Details about distribution of benign traffic and attacks are presented in Table 2.

Table 2. BoT-IoT category distribution.

Category	Subcategory	Number of Instances
Normal	Normal	477
DoS	TCP	615,800
	UDP	1,032,975
	HTTP	1485
DDoS	TCP	977,380
	UDP	948,255
	HTTP	989
Reconnaissance	OS Fingerprinting	17,914
	Service Scanning	73,168
Information Theft	Keylogging	73
	Data Exfiltration	6

4. Methodology

4.1. Preprocessing

Missing Values Imputation. In this study, missing numerical values are replaced with the mean of their corresponding class, whereas missing categorical values are imputed with the most frequent category within that class. In terms of the BoT-IoT dataset, we use only numerical features.

Features' Normalization. To normalize numerical features, Min-Max Scaling is applied, ensuring values fall between 0 and 1. Categorical features are transformed using one-hot encoding, excluding the first level to avoid redundancy.

4.2. Machine Learning Model

We adopt the machine learning model introduced in [28] with slight modifications. This machine learning model consists of CNN layers coupled with MoE.

4.2.1. CNN

- In terms of 5G-NIDD, we reshape the input features into a 2D matrix (6×13). This matrix is given as input to CNN layers. We design a convolutional neural network architecture to obtain a representation vector from input data. Specifically, we place 4 CNN cells in a row with 16, 32, 64, and 128 filters, respectively. Kernel size and padding are set equal to 3 and 1, respectively. Each CNN cell, except for the final one with 128 filters, consists of a 1D Convolutional, ReLU activation, and Max-Pooling layer. The final CNN cell does not include a max pooling layer. In terms of the max pooling layer, we set both kernel size and stride equal to 2.
- In terms of the BoT-IoT dataset, we reshape the input features into a 2D matrix (5×7). After that, we use three CNN cells. The first two CNN cells consist of a 1D Convolutional, a ReLU activation, and a Max-Pooling layer, while the third CNN cell consists of a 1D Convolutional and ReLU activation layer. All CNN layers have a padding of 1 and a kernel size of 3. The first CNN layer uses 16 filters, the second 32 filters, and the third 64 filters. In terms of the max pooling layer, we set both kernel size and stride equal to 2.

4.2.2. Mixture of Experts

Recent studies have demonstrated that sparsely gated Mixture-of-Experts (MoE) architectures can be efficiently trained in distributed environments using standard gradient-based optimization and parameter synchronization mechanisms [29]. In such settings, both

expert subnetworks and gating network parameters are updated locally and aggregated across compute nodes in a manner analogous to conventional neural network layers. This distributed compatibility makes MoE architectures suitable for integration within federated learning frameworks, where each client trains the full model locally prior to global aggregation. The output vector of the CNN layers is passed through a sparsely gated MoE layer, as described in [28,30]. This MoE layer is composed of a collection of experts and an associated gating network. The experts correspond to deep neural networks, while the gating network assigns a weight to each expert's output. Formally,

$$y_o = \sum_{i=1}^n g_i(x) y_i(x) \quad (1)$$

Noise and sparsity are also introduced, in order to ensure that the most important k experts are selected each time based on the input data. Formally,

$$\mathcal{G}(\mathbf{x}; \Theta)_i = \text{softmax}(\text{TopK}(g(\mathbf{x}; \Theta) + \mathcal{R}_{\text{noise}}, k))_i, \quad (2)$$

where $\mathcal{R}_{\text{noise}} = \text{StandardNormal}() \cdot \text{Softplus}((x \cdot W_{\text{noise}})_i)$

$$\text{TopK}(g(\mathbf{x}; \Theta), k)_i = \begin{cases} g(\mathbf{x}; \Theta)_i, & \text{condition,} \\ -\infty, & \text{otherwise.} \end{cases} \quad (3)$$

where

$$\text{condition: if } g(\mathbf{x}; \Theta)_i \text{ is in the top-}k \text{ elements of } g(\mathbf{x}; \Theta). \quad (4)$$

Two losses are utilized, namely the $L_{\text{importance}}$ and L_{load} . The $L_{\text{importance}}$ is responsible for balancing expert utilization, while the L_{load} facilitates balanced assignments of the training samples across all experts. Formally,

- $L_{\text{importance}}$

$$\text{Importance}(X) = \sum_{x \in X} G(x) \quad (5)$$

$$\mathcal{L}_{\text{importance}} = w_{\text{importance}} \cdot \text{CV}(\text{Importance}(X))^2 \quad (6)$$

- L_{load}

$$P(x, i) = \Phi \left(\frac{(x \cdot W_g)_i - \text{kth_excluding}(H(x), k, i)}{\text{Softplus}((x \cdot W_{\text{noise}})_i)} \right) \quad (7)$$

$$\text{Load}(X)_i = \sum_{x \in X} P(x, i) \quad (8)$$

$$\mathcal{L}_{\text{load}}(X) = w_{\text{load}} \cdot \text{CV}(\text{Load}(X))^2 \quad (9)$$

4.3. Dirichlet Label Partitioning Technique (Non-IID Client Distributions)

Similar to [10,31], we simulate a distributed environment with three clients by sampling from the Dirichlet(α) distribution on a per-label basis. Each sampled point determines the label proportion to be allocated to the clients, resulting in highly non-IID distributions when α is close to 0 and more iid-like distributions as α increases. Incorporating this approach maintains consistency with established benchmarks in FL research and offers the flexibility to create a realistic use-case scenario. In our experiments, we set $\alpha = 0.15$.

5G-NIDD. Training data labels allocations in terms of the 5G-NIDD dataset are presented in Figure 1.

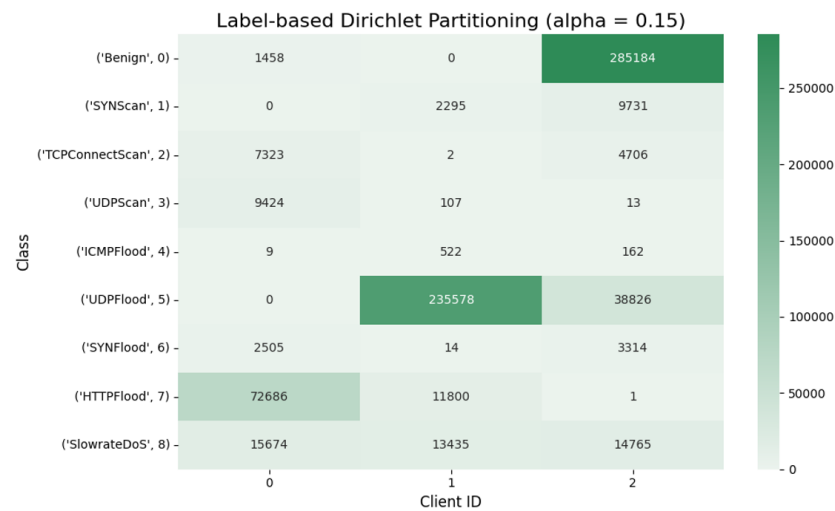


Figure 1. Training data labels allocations for Supervised Learning setup (5G-NIDD).

BoT-IoT. Training data labels allocations in terms of the BoT-IoT dataset are presented in Figure 2.

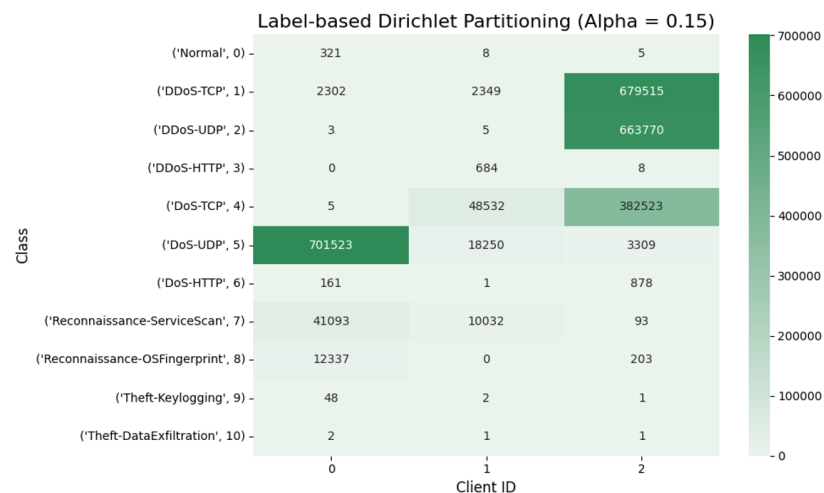


Figure 2. Training data labels allocations for Supervised Learning setup (BoT-IoT).

4.4. Federated Learning Strategies

In this section, we describe the FL strategies used.

FedAvg. This FL strategy is based on a simple averaging of the resultant local models across all clients. E epochs are applied to each client. The following equation is used:

$$w^{t+1} = \sum_{k=1}^N \frac{n_k}{n} w_k^t$$

where

- N denotes the number of clients;
- n denotes the total number of data samples across all participating clients;
- n_k refers to the number of data samples held by client k ;
- w_k^t denotes the k -th client's local update;
- w^{t+1} is the aggregated global model.

FedAvgM. This method adds momentum to the server by computing the following equations:

$$\Delta w = \sum_{k=1}^N \frac{n_k}{n} \Delta w_k$$

$$v \leftarrow \beta v + \Delta w$$

$$w \leftarrow w - v$$

where

- Δw_k denotes the weight update from the k -th client.

FedProx. This method introduces a proximal term to limit the effect of highly variable local updates. Thus, instead of minimizing the k -th client local function F_k , a μ -regularized objective is minimized instead. This term helps limit the divergence of client models from the global model, thereby leading to more stable convergence across non-IID data distributions.

$$w_k^{t+1} = \arg \min (F_k(w) + \frac{\mu}{2} \|w - w^t\|^2)$$

where

- w^t is the global model at round t ;
- μ is the proximal term;
- w_k^{t+1} is the locally optimized model for client k .

FedTrimmedAvg. This strategy trims a certain percentage of the highest and lowest updates before computing the final aggregation. The update is as follows:

$$w^{t+1} = \frac{1}{|S|} \sum_{k \in S} w_k^t$$

where

- S is the set of selected clients after removing the highest and lowest outliers;
- w_k^t is the local update for client k ;
- w^{t+1} is the aggregated global model.

5. Experiments

5.1. Experimental Setup

We perform a class-stratified train–test split of 60%/40% on both datasets, followed by the preprocessing steps described in Section 4.1. In the FL context, we consider $K = 3$ clients and implement the Dirichlet label partitioning technique for $\alpha = 0.15$ on the training data. This can be interpreted as a synthetic non-IID setting, commonly used in FL to induce label skew across clients. The training phase starts with a validation step, where 20% of each class is sampled per client to form a global validation set. This results in a training configuration of “batch size 1024”, “learning rate 0.0001”, “local epochs 5” and “federated rounds 30”, which is finally used to train the federated models from scratch on the entire (distributed) training set. Regarding FedProx, we use a proximal term accounting for 0.15. Recent analyses [32] of sparse MoE/Switch-MoE models report that when the number of experts is very small, overheads such as auxiliary load-balancing losses can overshadow the benefits of conditional computation, while employing very large numbers of experts yields diminishing returns. In terms of the 5G-NIDD dataset, we use 16 experts and select the 4 (value of k) most relevant ones. In terms of the Bot-IoT dataset, we use 64 experts and select the 16 (value of k) most relevant ones. In our setting, 5G-NIDD is smaller than Bot-IoT, so we adopt a moderate configuration (16 experts, top-4) that allows some specialization across attack types without inflating the global model size. For Bot-IoT, which is larger, we use a

higher-capacity MoE (64 experts, top-16) to better capture heterogeneous traffic profiles, still with a small fraction of experts active per input. We use Flower [33], which is an FL research framework. We conduct all experiments using PyTorch (<https://pytorch.org/>) [34].

5.2. Evaluation Metrics

Weighted precision, recall, and F1-score are used as metrics to evaluate the results of our proposed approach. We use weighted metrics to tackle the data imbalance.

Weighted Precision.

$$\text{Precision}_{\text{weighted}} = \sum_{i=1}^N \frac{w_i \cdot TP_i}{TP_i + FP_i}$$

where

$$w_i = \frac{TP_i + FN_i}{\sum_{j=1}^N (TP_j + FN_j)}$$

is the weight for class i .

Weighted Recall.

$$\text{Recall}_{\text{weighted}} = \sum_{i=1}^N \frac{w_i \cdot TP_i}{TP_i + FN_i}$$

Weighted F1-score.

$$\text{F1}_{\text{weighted}} = \sum_{i=1}^N w_i \cdot \frac{2 \cdot TP_i}{2TP_i + FP_i + FN_i}$$

Here,

- TP_i , FP_i , and FN_i are the true positives, false positives, and false negatives for class i ;
- N is the number of classes;
- w_i is the weight for each class, based on its support (i.e., the number of true instances for that class).

6. Results

6.1. Results on the 5G-NIDD Dataset

The results of our proposed approach on the 5G-NIDD dataset are reported in Table 3.

Among the four FL strategies, FedTrimmedAvg demonstrates the strongest overall performance, achieving the highest weighted precision (98.4401%) and weighted F1-score (98.1134%). This suggests that FedTrimmedAvg's robust aggregation approach is most effective at maintaining optimal precision-recall balance while mitigating the impact of outlier updates. FedAvg, the standard federated averaging method, performs nearly as well with a weighted F1-score of 98.0440%, but is slightly outperformed by FedTrimmedAvg in both precision (by 0.1985%) and F1-score (by 0.0694%). FedAvgM, which incorporates momentum into the averaging process, shows significantly lower performance across all metrics (weighted F1-score of 94.7341%), indicating that momentum updates may be detrimental in this context. FedProx exhibits the poorest performance (weighted F1-score of 73.7300%), suggesting its proximal term optimization fails to deliver benefits for this particular application.

Comparing the differences between strategies, FedTrimmedAvg surpasses FedAvg in weighted precision by 0.1985%, while FedAvg maintains a slight advantage in recall (by 0.0521%). The more substantial performance gaps become apparent when comparing these top performers with other approaches: FedTrimmedAvg outperforms FedAvgM by 3.3793% in weighted F1-score and dominates FedProx by an impressive 24.3834% in

F1-score. Similarly, FedAvg shows clear advantages over FedAvgM (by 3.3099% in F1-score) and FedProx (by 24.3140% in F1-score). These results demonstrate that while the difference between FedTrimmedAvg and FedAvg is relatively small, both significantly outperform alternative approaches, with FedTrimmedAvg emerging as the most balanced and effective strategy overall.

Overall, FedTrimmedAvg performs best overall, achieving the highest precision and F1-score, demonstrating robustness. FedAvg follows closely, making it a strong alternative. FedAvgM lags significantly, while FedProx performs worst, highlighting its limitations. These results show FedTrimmedAvg's effectiveness and the importance of strategic algorithm selection in FL.

Table 3. Comparison of FL strategies on the 5G-NIDD dataset.

FL Strategy	Weighted Prec. (%)	Weighted Rec. (%)	Weighted F1-Score (%)
FedAvg	98.2416	98.0087	98.0440
FedAvgM	95.7081	95.0491	94.7341
FedProx	84.3787	76.0369	73.7300
FedTrimmedAvg	98.4401	97.9566	98.1134

6.2. Results on the BoT-IoT Dataset

The results of our proposed approach on the BoT-IoT dataset are reported in Table 4.

Among the four strategies, FedAvgM demonstrates the best performance, achieving the highest weighted precision (95.5135%), weighted recall (95.2739%), and weighted F1-score (95.3295%). This suggests that FedAvgM is the most effective in maintaining both precision and recall balance, leading to superior overall model performance. FedAvg, the standard averaging method, performs well with a weighted F1-score of 90.2256%, but it is outperformed by FedAvgM in all metrics. FedProx, which modifies FedAvg by adding a proximal term, achieves slightly lower values (90.7212% weighted precision, 88.8838% weighted recall, and 88.1829% weighted F1-score), indicating a trade-off in performance. Finally, FedTrimmedAvg exhibits the lowest scores across all metrics, with a weighted F1-score of 83.7561%, suggesting that its robustness against outliers comes at the cost of reduced accuracy. Overall, FedAvgM emerges as the most effective strategy, significantly outperforming the others in all aspects.

Comparing the differences between strategies, FedAvg surpasses FedTrimmedAvg in weighted precision by 2.1535%, in weighted recall by 5.9059%, and in weighted F1-score by 6.4695%, indicating that FedTrimmedAvg's robustness mechanism may come at the cost of overall accuracy. Similarly, FedAvg outperforms FedProx by 0.5858% in weighted precision, 1.3974% in weighted recall, and 2.0427% in weighted F1-score, suggesting that FedProx introduces some performance trade-offs despite its modifications. On the other hand, FedAvgM surpasses FedAvg by 4.2065% in weighted precision, 4.9927% in weighted recall, and 5.1039% in weighted F1-score, highlighting the significant improvement provided by momentum-based updates.

Overall, FedAvgM emerges as the most effective strategy, outperforming all others by a substantial margin, while FedAvg maintains a balanced performance. In contrast, FedProx and FedTrimmedAvg exhibit lower scores, indicating trade-offs in robustness and optimization approaches.

Table 4. Comparison of FL strategies on the BoT-IoT dataset.

FL Strategy	Weighted Prec. (%)	Weighted Rec. (%)	Weighted F1-Score (%)
FedAvg	91.3070	90.2812	90.2256
FedAvgM	95.5135	95.2739	95.3295
FedProx	90.7212	88.8838	88.1829
FedTrimmedAvg	89.1535	84.3753	83.7561

7. Discussion

7.1. Interpreting Differences Between Aggregation Strategies

On 5G-NIDD, clients have a relatively diverse but only moderately imbalanced mix of traffic, so occasional strongly biased updates behave like outliers; in this case, the robust aggregation of FedTrimmedAvg, which downweights extreme updates, is beneficial. In contrast, BoT-IoT is extremely imbalanced with some rare attacks concentrated on specific clients, so trimming “extreme” updates can discard useful signal from those clients, while FedAvgM’s server-side momentum can accumulate their information over multiple rounds and thus performs best. Recent IoT security research has also explored graph-based modeling for detecting complex and stealthy threats. For example, the authors in [35] propose a Graph Neural Diffusion Network for sophisticated persistent threat hunting, leveraging relational structures and temporal encoding to capture multi-stage attack patterns. Although such approaches explicitly model inter-device dependencies, the proposed CNN–MoE framework focuses on flow-level representations within a privacy-preserving federated setting, addressing client heterogeneity through conditional computation. These directions are complementary, and integrating graph-aware representations into federated MoE-based intrusion detection is an interesting avenue for future work.

7.2. Impact of MoE Hyperparameters on Performance on the 5G-NIDD Dataset

To further assess the robustness of the proposed architecture, we conducted a small ablation study on the 5G-NIDD dataset, varying the number of experts (E) and the number of selected experts per input (top-k). Results are reported in Table 5. All experiments were performed under the same federated configuration, using FedTrimmedAvg, which demonstrated the strongest performance on this dataset.

Table 5. Impact of MoE hyperparameters on 5G-NIDD (weighted F1-score, %).

# Experts (E)	Top-k (k)	Weighted F1 (%)
8	2	97.65
8	4	97.82
16	2	97.95
16	4	98.11
16	8	98.02
32	4	98.07
32	8	98.05

7.3. Impact of Non-IID Configuration and Number of Clients

We intentionally select a relatively small Dirichlet concentration parameter $\alpha = 0.15$ with $K = 3$ clients to simulate a strongly non-IID scenario. This choice follows common practice in federated learning benchmarking, where small α values are used to emulate severe label skew and stress-test aggregation robustness. Larger α values yield per-client label distributions that progressively resemble the global distribution, effectively approach-

ing an almost IID setting in which standard federated algorithms typically perform well and performance differences between aggregation strategies tend to diminish.

It is important to clarify that $\alpha = 0.15$ does not aim to perfectly replicate real-world 5G or IoT traffic distributions. In practical deployments, heterogeneity arises due to factors such as geographic separation of base stations, device-type specialization (e.g., industrial versus consumer IoT), localized attack targeting, and temporal traffic variations. These factors may lead to significant imbalance across network segments, with certain sites predominantly observing specific attack categories. Although real operational distributions may not strictly follow a Dirichlet form, a small α provides a controlled and reproducible proxy for strong label concentration, enabling systematic evaluation under challenging heterogeneous conditions.

Our goal in this work is therefore to evaluate the proposed MoE-based IDS and the considered aggregation strategies under a difficult non-IID regime, rather than to optimize performance in an almost IID configuration obtained with larger α .

Next, we experiment with different values of the Dirichlet concentration parameter α in order to evaluate the sensitivity of the proposed federated MoE framework under varying levels of data heterogeneity. Results are reported in Table 6. Specifically, we consider $\alpha \in 0.10, 0.15, 1.50, 2.00$, where smaller values correspond to stronger label skew across clients and larger values approach more homogeneous (near-IID) distributions.

Table 6. Sensitivity analysis of Dirichlet hyperparameter α on 5G-NIDD (FedTrimmedAvg).

α	Weighted F1 (%)
0.10	97.94
0.15	98.11
1.50	98.03
2.00	98.01

8. Conclusions

In this paper, we present a study incorporating a sparsely gated MoE layer into an FL framework for identifying attacks in 5G and IoT networks. Specifically, as a machine learning model, we use CNNs coupled with MoE, where the most important experts are utilized according to the input data, thus alleviating the issue of model complexity. To mitigate issues of non-IID data, commonly referred to as data heterogeneity, we utilize the Dirichlet label partitioning technique. Next, four FL strategies are compared, namely FedAvg, FedAvgM, FedProx, and FedTrimmedAvg. Experiments are performed on two publicly available datasets, namely 5G-NIDD and BoT-IoT. Results show that our proposed method reaches a weighted F1-score of up to 98.1134% and 95.3295% on the 5G-NIDD and BoT-IoT datasets, respectively.

Below, we mention some ideas for future work:

- Explainable AI: Research on the impact of non-IIDness to the federated model. The analysis will extend in terms of both predictive capabilities and feature importance scores, fostering the transparency aspect of federated models as well.
- Further aggregation strategies will be tested to determine their impact on model convergence and generalization capabilities, including FedYogi, FedAdam, StatAvg, and FedAdaGrad strategies.
- Systematic exploration of (E, k) in federated 5G/IoT intrusion detection.
- Experiments on additional datasets: The models may also be tested on additional datasets (e.g., ToN-IoT, datasets developed in the context of the NANCY SNS JU project (<https://zenodo.org/records/14811122>, accessed on 10 June 2025) and iTrust6G SNS JU project (<https://www.sns-itrust6g.com/>, accessed on 10 June 2025)).

Author Contributions: Conceptualization, L.I. and V.L.; methodology, L.I., G.D. and V.L.; software, L.I. and V.L.; validation, L.I., G.D., V.L. and S.M.; formal analysis, L.I.; investigation, L.I.; resources, S.M., C.N. and D.A.; data curation, L.I.; writing—original draft preparation, L.I.; writing—review and editing, L.I., G.D., V.L., S.M., C.N. and D.A.; visualization, L.I. and V.L.; project administration, C.N. and D.A.; funding acquisition, C.N. and D.A. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union’s Horizon Europe research and innovation programme under Grant Agreement No 101139198, iTrust6G project. Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or SNS-JU. Neither the European Union nor the granting authority can be held responsible for them.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are publicly available.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

5G	Fifth generation
IoT	Internet of Things
FL	Federated Learning
IDS	Intrusion detection system
MoE	Mixture of Experts

Appendix A

Table A1. Description of features.

Features	Description	Type	No. of Features After Preprocessing
Seq	Sequence Number	Numerical	1
Dur	Duration	Numerical	1
RunTime	Run Time	Numerical	1
Mean	Mean Value	Numerical	1
Sum	Sum Value	Numerical	1
Min	Minimum Value	Numerical	1
Max	Maximum Value	Numerical	1
Proto	Protocol Type	Categorical	7
sTos	Type of Service values	Numerical	1
dTos		Numerical	1
sDSb	Source and Destination Behavioral Flags	Categorical	11
dDSb		Categorical	5
sTtl	Time-to-Live values for source and destination	Numerical	1
dTtl		Numerical	1
sHops	Number of hops taken by packets from source to destination	Numerical	1
dHops		Numerical	1
Cause	Cause code	Categorical	2

Table A1. Cont.

Features	Description	Type	No. of Features After Preprocessing
TotPkts	Total Packets	Numerical	1
SrcPkts	Source Packets	Numerical	1
DstPkts	Destination Packets	Numerical	1
TotBytes	Total Bytes	Numerical	1
SrcBytes	Source Bytes	Numerical	1
DstBytes	Destination Bytes	Numerical	1
Offset	Offset Value	Numerical	1
sMeanPktSz	Source Mean Packet Size	Numerical	1
dMeanPktSz	Destination Mean Packet Size	Numerical	1
Load	Load Value	Numerical	1
SrcLoad	Source Load	Numerical	1
DstLoad	Destination Load	Numerical	1
Loss	Total Loss	Numerical	1
SrcLoss	Source Loss	Numerical	1
DstLoss	Destination Loss	Numerical	1
pLoss	Packet Loss	Numerical	1
SrcGap	Time Gaps	Numerical	1
DstGap		Numerical	1
Rate		Numerical	1
SrcRate	Transmission Rates	Numerical	1
DstRate		Numerical	1
State	State Information	Categorical	10
SrcWin	TCP Window Sizes	Numerical	1
DstWin		Numerical	1
sVid	VLAN IDs	Numerical	1
dVid		Numerical	1
SrcTCPBase	TCP Base Values	Numerical	1
DstTCPBase		Numerical	1
TcpRtt	TCP Round Trip-Time	Numerical	1
SynAck	Specific Packet Types	Numerical	1
AckDat		Numerical	1
Total	-	-	78

Appendix B

Table A2. Feature descriptions.

Features	Description
Stime	Record start time
flgs number	Numerical representation of feature flags
proto number	Numerical representation of feature proto
pkts	Total count of packets in transaction
bytes	Total number of bytes in transaction

Table A2. Cont.

Features	Description
state number	Numerical representation of feature state
ltime	Record last time
seq	Argus sequence number
dur	Record total duration
mean	Average duration of aggregated records
stddev	Standard deviation of aggregated records
sum	Total duration of aggregated records
min	Minimum duration of aggregated records
max	Maximum duration of aggregated records
spkts	Source-to-destination packet count
dpkts	Destination-to-source packet count
sbytes	Source-to-destination byte count
dbytes	Destination-to-source byte count
rate	Total packets per second in transaction
srates	Source-to-destination packets per second
drates	Destination-to-source packets per second
TnBPSSrcIP	Total Number of bytes per source IP
TnBPDstIP	Total Number of bytes per Destination IP
TnP_PSrcIP	Total Number of packets per source IP
TnP_PDstIP	Total Number of packets per Destination IP
TnP PerProto	Total Number of packets per protocol
TnP_Per_Dport	Total Number of packets per dport
AR_P_Protocol_P_SrcIP	Mean rate per protocol per Source IP
AR_P_Protocol_P_DstIP	Mean rate per protocol per Destination IP
N_IN_Conn_P_SrcIP	Count of inbound connections per source IP
N_IN_Conn_P_DstIP	Count of inbound connections per destination IP
AR_P_Protocol_P_Sport	Average rate per protocol per sport
AR_P_Protocol_P_Dport	Average rate per protocol per dport
Pkts_P_State_P_Protocol_P_DstIP	Packet counts by state and protocol for each destination IP
Pkts_P_State_P_Protocol_P_SrcIP	Packet counts by state and protocol for each source IP

References

1. Radoglou-Grammatikis, P.; Sarigiannidis, P.; Efstathiopoulos, G.; Lagkas, T.; Fragulis, G.; Sarigiannidis, A. A Self-Learning Approach for Detecting Intrusions in Healthcare Systems. In *Proceedings of the ICC 2021—IEEE International Conference on Communications*; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6. [\[CrossRef\]](#)
2. Mwang'onda, A.N.; Phiri, M. Comprehensive survey study on fifth-generation wireless network and the internet of things. *EAI Endorsed Trans. Internet Things* **2023**, *9*, e3. [\[CrossRef\]](#)
3. Bovenzi, G.; Di Monda, D.; Montieri, A.; Persico, V.; Pescapè, A. Classifying attack traffic in IoT environments via few-shot learning. *J. Inf. Secur. Appl.* **2024**, *83*, 103762. [\[CrossRef\]](#)
4. Amponis, G.; Radoglou-Grammatikis, P.; Lagkas, T.; Ouzounidis, S.; Zevgara, M.; Moscholios, I.; Goudos, S.; Sarigiannidis, P. Towards Securing Next-Generation Networks: Attacking 5G Core/RAN Testbed. In *Proceedings of the 2022 Panhellenic Conference on Electronics & Telecommunications (PACET)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–4. [\[CrossRef\]](#)
5. Ahmad, J.; Latif, S.; Khan, I.U.; Alshehri, M.S.; Khan, M.S.; Alasbali, N.; Jiang, W. An interpretable deep learning framework for intrusion detection in industrial Internet of Things. *Internet Things* **2025**, *33*, 101681. [\[CrossRef\]](#)

6. Perifanis, V.; Pavlidis, N.; Koutsiamanis, R.A.; Efraimidis, P.S. Federated learning for 5G base station traffic forecasting. *Comput. Netw.* **2023**, *235*, 109950. [\[CrossRef\]](#)
7. Makris, I.; Ntampakis, N.; Lagkas, T.; Radoglou-Grammatikis, P.; Goudos, S.K.; Argyriou, V.; Fountoukidis, E.; Skarmeta, A.F.; Saura, P.F.; Sarigiannidis, P. Elevating 5G Network Security: A Profound Examination of Federated Learning Aggregation Strategies for Attack Detection. In *Proceedings of the 2023 IEEE Future Networks World Forum (FNWF)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–6. [\[CrossRef\]](#)
8. Sharma, H.; Yadav, G.S. Privacy-preserving intrusion detection with federated learning: Enhancing privacy and efficiency in IoT networks. *Int. J. Mach. Learn. Cybern.* **2025**, *16*, 5821–5839. [\[CrossRef\]](#)
9. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Arcas, B.A.y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics; Proceedings of Machine Learning Research*; Singh, A., Zhu, J., Eds.; PMLR: Cambridge, MA, USA, 2017; Volume 54, pp. 1273–1282.
10. Hsu, T.M.H.; Qi, H.; Brown, M. Measuring the effects of non-identical data distribution for federated visual classification. *arXiv* **2019**, arXiv:1909.06335. [\[CrossRef\]](#)
11. Li, T.; Sahu, A.K.; Zaheer, M.; Sanjabi, M.; Talwalkar, A.; Smith, V. Federated optimization in heterogeneous networks. *Proc. Mach. Learn. Syst.* **2020**, *2*, 429–450.
12. Yin, D.; Chen, Y.; Kannan, R.; Bartlett, P. Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates. In *Proceedings of the 35th International Conference on Machine Learning; Proceedings of Machine Learning Research*; Dy, J., Krause, A., Eds.; PMLR: Cambridge, MA, USA, 2018; Volume 80, pp. 5650–5659.
13. Blika, A.; Palmos, S.; Doukas, G.; Lamprou, V.; Pelekis, S.; Kontoulis, M.; Ntanos, C.; Askounis, D. Federated Learning For Enhanced Cybersecurity And Trustworthiness In 5G and 6G Networks: A Comprehensive Survey. *IEEE Open J. Commun. Soc.* **2024**, *6*, 3094–3130. [\[CrossRef\]](#)
14. Ghimire, B.; Rawat, D.B. Recent Advances on Federated Learning for Cybersecurity and Cybersecurity for Federated Learning for Internet of Things. *IEEE Internet Things J.* **2022**, *9*, 8229–8249. [\[CrossRef\]](#)
15. Bhanbhro, J.; Nisticò, S.; Palopoli, L. Issues in federated learning: Some experiments and preliminary results. *Sci. Rep.* **2024**, *14*, 29881. [\[CrossRef\]](#)
16. Campos, E.M.; Saura, P.F.; González-Vidal, A.; Hernández-Ramos, J.L.; Bernabé, J.B.; Baldini, G.; Skarmeta, A. Evaluating Federated Learning for intrusion detection in Internet of Things: Review and challenges. *Comput. Netw.* **2022**, *203*, 108661. [\[CrossRef\]](#)
17. Agrawal, S.; Sarkar, S.; Aouedi, O.; Yenduri, G.; Piamrat, K.; Alazab, M.; Bhattacharya, S.; Maddikunta, P.K.R.; Gadekallu, T.R. Federated Learning for intrusion detection system: Concepts, challenges and future directions. *Comput. Commun.* **2022**, *195*, 346–361. [\[CrossRef\]](#)
18. Iyer, V.N. A review on different techniques used to combat the non-IID and heterogeneous nature of data in FL. *arXiv* **2024**, arXiv:2401.00809. [\[CrossRef\]](#)
19. Bouzinis, P.S.; Radoglou-Grammatikis, P.; Makris, I.; Lagkas, T.; Argyriou, V.; Papadopoulos, G.T.; Sarigiannidis, P.; Karagiannidis, G.K. StatAvg: Mitigating Data Heterogeneity in Federated Learning for Intrusion Detection Systems. *arXiv* **2024**, arXiv:2405.13062. [\[CrossRef\]](#)
20. Yao, X.; Huang, T.; Wu, C.; Zhang, R.; Sun, L. Towards Faster and Better Federated Learning: A Feature Fusion Approach. In *Proceedings of the 2019 IEEE International Conference on Image Processing (ICIP)*; IEEE: Piscataway, NJ, USA, 2019; pp. 175–179. [\[CrossRef\]](#)
21. Du, Z.; Sun, J.; Li, A.; Chen, P.Y.; Zhang, J.; Li, H.H.; Chen, Y. Rethinking normalization methods in federated learning. In *DistributedML '22: Proceedings of the 3rd International Workshop on Distributed Machine Learning*; Association for Computing Machinery: New York, NY, USA, 2022; pp. 16–22. [\[CrossRef\]](#)
22. Blei, D.M.; Ng, A.Y.; Jordan, M.I. Latent dirichlet allocation. *J. Mach. Learn. Res.* **2003**, *3*, 993–1022.
23. Neto, E.C.P.; Dadkhah, S.; Ferreira, R.; Zohourian, A.; Lu, R.; Ghorbani, A.A. CICIOT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. *Sensors* **2023**, *23*, 5941. [\[CrossRef\]](#)
24. Dontu, S.; Addula, S.R.; Kumar Pareek, P.; Vallabhaneni, R.; Adnan, M.M. Attack Detection from Internet of Things using TPE based Self-Attention based Bidirectional Long-Short Term Memory. In *Proceedings of the 2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [\[CrossRef\]](#)
25. Samarakoon, S.; Siriwardhana, Y.; Porambage, P.; Liyanage, M.; Chang, S.Y.; Kim, J.; Kim, J.; Ylianttila, M. 5g-nidd: A comprehensive network intrusion detection dataset generated over 5g wireless network. *arXiv* **2022**, arXiv:2212.01298. [\[CrossRef\]](#)
26. Koroniotis, N.; Moustafa, N.; Sitnikova, E.; Turnbull, B. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Gener. Comput. Syst.* **2019**, *100*, 779–796. [\[CrossRef\]](#)
27. Peterson, J.M.; Leevy, J.L.; Khoshgoftaar, T.M. A Review and Analysis of the Bot-IoT Dataset. In *Proceedings of the 2021 IEEE International Conference on Service-Oriented System Engineering (SOSE)*; IEEE: Piscataway, NJ, USA, 2021; pp. 20–27. [\[CrossRef\]](#)

28. Ilias, L.; Doukas, G.; Lamprou, V.; Ntanos, C.; Askounis, D. Convolutional Neural Networks and Mixture of Experts for Intrusion Detection in 5G Networks and beyond. *arXiv* **2024**, arXiv:2412.03483. [[CrossRef](#)]
29. Sievers, J.; Blank, T.; Simon, F. Advancing Accuracy in Energy Forecasting using Mixture-of-Experts and Federated Learning. In *e-Energy '24: Proceedings of the 15th ACM International Conference on Future and Sustainable Energy Systems*; Association for Computing Machinery: New York, NY, USA, 2024; pp. 65–83. [[CrossRef](#)]
30. Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; Dean, J. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. *arXiv* **2017**, arXiv:1701.06538.
31. Tang, Z.; Hu, Z.; Shi, S.; Cheung, Y.m.; Jin, Y.; Ren, Z.; Chu, X. Data resampling for federated learning with non-iid labels. In *Proceedings of the International Workshop on Federated and Transfer Learning for Data Sparsity and Confidentiality in Conjunction with IJCAI*, Montreal, QC, Canada, 19–27 August 2021.
32. Ludziejewski, J.; Pióro, M.; Krajewski, J.; Stefaniak, M.; Krutul, M.; Małaśnicki, J.; Cygan, M.; Sankowski, P.; Adamczewski, K.; Miłoś, P.; et al. Joint MoE Scaling Laws: Mixture of Experts Can Be Memory Efficient. In *Proceedings of the 42nd International Conference on Machine Learning*; *Proceedings of Machine Learning Research*; Singh, A., Fazel, M., Hsu, D., Lacoste-Julien, S., Berkenkamp, F., Maharaj, T., Wagstaff, K., Zhu, J., Eds.; PMLR: Cambridge, MA, USA, 2025; Volume 267, pp. 41056–41073.
33. Beutel, D.J.; Topal, T.; Mathur, A.; Qiu, X.; Fernandez-Marques, J.; Gao, Y.; Sani, L.; Kwing, H.L.; Parcollet, T.; Gusmão, P.P.d.; et al. Flower: A Friendly Federated Learning Research Framework. *arXiv* **2020**, arXiv:2007.14390.
34. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2019.
35. Yenugula, M.; Konda, B.; Kasula, V.K.; Yadulla, A.R.; Tumma, C.; Rakki, S.B. A Graph Neural Diffusion Network for Sophisticated Persistent Threat Hunting in IoT Environments. In *Proceedings of the 2025 IEEE Conference on Computer Applications (ICCA)*; IEEE: Piscataway, NJ, USA, 2025; pp. 1–6. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.