*IoT*

MDPI

*Article*

# Mean/Std: Lightweight Distribution-Aware Aggregation for Federated IoT Botnet Detection

**Yassine El Yamani *, Youssef Baddi and Najib El Kamoun**

STIC Lab, FSJ, Chouaib Doukkali University, El Jadida 24000, Morocco; baddi.y@ucd.ac.ma (Y.B.); elkamoun.n@ucd.ac.ma (N.E.K.)
* Correspondence: elyamani.y@ucd.ac.ma

**Abstract**

Federated learning (FL) is a promising paradigm for privacy-preserving IoT intrusion detection, but its effectiveness can be substantially degraded by the combination of heterogeneous non-IID client distributions and severe multi-class imbalance. Under such conditions, conventional size-based aggregation may overemphasize large yet highly skewed clients, limiting the representation of minority attack classes in the global model. To address this issue, we propose Mean/Std, a lightweight distribution-aware aggregation strategy that combines a client-size proxy with two complementary statistics of local label distributions, namely the standard deviation and the dominance gap of class proportions, while preserving a communication footprint comparable to FedAvg. Experiments on the N-BaIoT benchmark, comprising seven heterogeneous IoT clients and eleven traffic classes, are conducted under a privacy-oriented update-perturbation setting inspired by secure aggregation workflows. The results show that Mean/Std consistently provides the strongest imbalance-aware performance among the evaluated FL baselines, achieving a Macro-F1 score of 0.8418 and a Balanced Accuracy of 0.8722 while improving the representation of minority attack classes. Additional experiments across five independent random seeds and a comprehensive hyperparameter sensitivity analysis further confirm the robustness and stability of the proposed aggregation mechanism. Overall, the results demonstrate that lightweight distribution-aware aggregation offers an effective, robust, and practically deployable solution for mitigating aggregation bias under simultaneous non-IID heterogeneity and severe multi-class imbalance in FL-based IoT botnet detection.

**Keywords:** federated learning; IoT botnet detection; distribution-aware aggregation; non-IID data; class imbalance; label skew; privacy-preserving intrusion detection



Academic Editor: Amiya Nayak

Received: 3 May 2026
Revised: 25 June 2026
Accepted: 26 June 2026
Published: 7 July 2026



## 1. Introduction

The rapid proliferation of Internet of Things (IoT) devices across smart homes, healthcare systems, industrial environments, and critical infrastructures has considerably expanded the cyberattack surface. Owing to limited computational resources, weak authentication mechanisms, and infrequent security updates, many IoT devices remain vulnerable to compromise and botnet recruitment [1]. Once infected, heterogeneous devices such as cameras, sensors, and routers can be orchestrated to launch large-scale attacks, including distributed denial-of-service (DDoS), network scanning, and malware propagation [1]. Consequently, scalable and effective IoT botnet detection has become a major cybersecurity challenge.

Machine Learning (ML) and Deep Learning (DL) techniques have demonstrated strong capabilities for identifying malicious traffic patterns and generalizing beyond handcrafted signatures [1]. However, most existing solutions rely on centralized training, requiring traffic traces from multiple devices to be collected at a central server. Although effective for model development, this paradigm raises privacy, regulatory, and scalability concerns because network traffic may contain sensitive behavioural information and generate large volumes of telemetry [2,3]. Moreover, IoT security datasets typically exhibit severe multi-class imbalance, where benign traffic dominates and some attack categories remain underrepresented, making overall accuracy an insufficient indicator of detection quality [4,5].

Federated learning (FL) addresses these limitations by enabling collaborative model training without exchanging raw data [6,7]. Clients perform local optimization and only share model updates, making FL particularly attractive for privacy-preserving IoT intrusion detection [8,9]. In practical deployments, however, client data are inherently non-IID: devices differ in functionality, traffic volume, operating conditions, and attack exposure, resulting in heterogeneous local label distributions [10]. When such heterogeneity coexists with severe class imbalance, conventional size-based aggregation may overemphasize large but highly skewed clients, reducing the contribution of minority attack behaviours to the global model.

Most existing FL approaches address heterogeneity through optimization enhancements, adaptive updates, or client-drift correction mechanisms. While these techniques often improve convergence, they generally do not explicitly account for aggregation bias induced by heterogeneous class distributions. As a result, information associated with minority attack categories concentrated on a limited subset of clients may remain underrepresented during global aggregation. This motivates aggregation strategies that exploit lightweight distributional information while preserving the simplicity and communication efficiency required by resource-constrained IoT environments.

To clarify the distinction between isolated on-device learning and collaborative FL, Figure 1 contrasts (i) distributed learning, where each device independently trains a local model, and (ii) FL, where multiple devices jointly optimize a shared global model through iterative model-update aggregation.

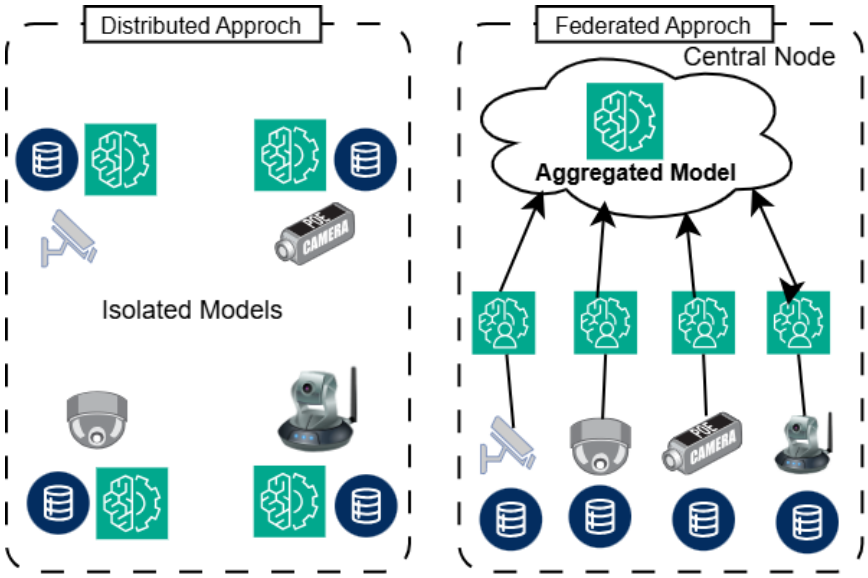


**Figure 1.** High-level comparison between isolated local training and federated learning (FL) for IoT security analytics. In FL, arrows denote local updates sent to the server and the aggregated global model returned to the clients.

In this work, we investigate aggregation design for multi-class IoT botnet detection under simultaneous non-IID heterogeneity and severe class imbalance. We propose *Mean/Std*, a lightweight distribution-aware aggregation strategy that adjusts client influence using simple statistics derived from local label proportions. Unlike optimization-oriented approaches, Mean/Std directly mitigates imbalance-induced aggregation bias while preserving a communication footprint comparable to FedAvg.

The proposed approach is evaluated on the N-BaIoT benchmark [11], which provides a realistic setting involving multiple heterogeneous IoT devices, diverse traffic patterns, and pronounced class imbalance. Although our experiments focus on N-BaIoT, the underlying challenge addressed in this work is more general and frequently encountered in cybersecurity applications, where dominant benign traffic coexists with rare but operationally critical attack behaviours [4,5].

Contributions: The main contributions of this work are as follows:

- We analyze the impact of aggregation bias in FL-based multi-class IoT botnet detection under simultaneous non-IID label skew and severe class imbalance.
- We propose Mean/Std, a lightweight distribution-aware aggregation strategy that exploits simple statistics of local label distributions to mitigate imbalance-induced client dominance during aggregation.
- We evaluate the proposed approach on the N-BaIoT benchmark across seven heterogeneous IoT clients and eleven traffic classes, comparing it with representative optimization-oriented and fairness-oriented FL baselines under a privacy-oriented update-perturbation setting inspired by secure aggregation [12]. The evaluation includes global performance analysis, per-class minority attack assessment, statistical robustness across five independent random seeds, and a comprehensive hyperparameter sensitivity study.

## 2. Related Work

Federated learning (FL) has emerged as a promising paradigm for privacy-preserving intrusion detection in IoT environments, where traffic data are naturally distributed across heterogeneous devices and cannot always be centralized because of privacy, ownership, bandwidth, or regulatory constraints. Recent FL-IDS surveys consistently identify three major deployment challenges: statistical heterogeneity, severe label skew and class imbalance, and privacy-preserving model training [8,9,13]. Dataset-oriented studies further show that FL-IDS performance is highly sensitive to dataset characteristics, client partitioning strategies, and evaluation protocols, emphasizing the importance of robust aggregation mechanisms and imbalance-aware evaluation metrics [4,5,14].

These challenges are particularly relevant to IoT botnet detection. Although centralized ML and DL approaches achieve strong predictive performance, IoT traffic datasets are typically characterized by severe multi-class imbalance, where benign traffic and frequent attack categories dominate the learning process, while rare but operationally important attacks remain underrepresented [1]. Under such conditions, Macro-F1, Balanced Accuracy, and PR-AUC provide a more informative assessment than accuracy alone [4,5]. When severe imbalance coexists with non-IID client distributions, aggregation becomes a critical component of the FL pipeline, since large but highly skewed clients may exert a disproportionate influence on the global model.

FedAvg remains the most widely adopted aggregation strategy because of its simplicity, scalability, and communication efficiency [6]. However, its size-based weighting assumes that larger clients should contribute more strongly to the global update, which may be suboptimal when client size correlates with label skew. Several methods have therefore been proposed to improve learning under heterogeneous data distributions.

FedProx reduces client drift through proximal regularization [15], SCAFFOLD mitigates local-update bias using control variates [16], FedNova normalizes client updates to alleviate objective inconsistency [17], and FedOpt/FedAdam improves convergence through adaptive server-side optimization [18]. Recent surveys identify aggregation redesign as a key research direction for addressing non-IID learning in FL [19–22]. Nevertheless, these approaches primarily target optimization stability and convergence rather than aggregation bias induced by heterogeneous class distributions.

Recent FL-IDS studies have explored cybersecurity-specific solutions. FADngs combines federated anomaly detection with density estimation and ensemble knowledge distillation [23]. Fan et al. proposed a clustered FL framework for heterogeneous IoT anomaly detection [24]. FedMADE enhances minority attack detection through dynamic aggregation and client grouping [25]; IFCEA combines clustering, committee-based client selection, and edge–cloud collaboration [26]; and FD-IDS leverages knowledge distillation to mitigate non-IID effects in IoT intrusion detection [27]. While effective, these approaches generally rely on clustering, auxiliary models, contribution estimation, or distillation mechanisms and, therefore, address heterogeneity through mechanisms other than aggregation-weight redesign.

The broader FL literature has also investigated class imbalance and client fairness. Federated long-tailed learning studies report systematic under-representation of minority classes under size-driven aggregation [28]. Likewise, Agnostic Federated Learning (AFL) [29] and q-FedAvg [30] reduce disparities across clients through modified optimization objectives or aggregation rules. However, their primary objective is client-level fairness rather than the explicit correction of label-distribution skew in multi-class IoT botnet detection.

Privacy remains another important consideration in FL-IDS. Although FL eliminates the need to share raw traffic traces, model updates may still leak sensitive information through reconstruction or gradient inversion attacks [31]. Secure aggregation mitigates this risk by ensuring that only aggregated updates are observable by the server [12]. Accordingly, all methods considered in this study are evaluated under a privacy-oriented update-perturbation setting designed to emulate privacy-constrained learning conditions, rather than provide cryptographic privacy guarantees.

Overall, the literature indicates that aggregation remains a key challenge when non-IID heterogeneity and severe multi-class imbalance coexist. Existing solutions primarily address this issue through optimization correction, clustering, fairness objectives, distillation, or auxiliary learning mechanisms. In contrast, relatively few studies explicitly target the combined effect of non-IID heterogeneity and severe class imbalance through the aggregation process itself, particularly in IoT botnet detection. Mean/Std addresses this gap by incorporating lightweight statistics of local label distributions directly into the aggregation stage, mitigating the dominance of large but highly skewed clients while preserving the simplicity and communication efficiency of FedAvg.

Table 1 provides a structured comparison between representative FL, FL-IDS, and imbalance-aware approaches. The comparison highlights whether each method explicitly addresses non-IID heterogeneity, class imbalance, and local label distributions, together with its aggregation principle and additional mechanisms. The table shows that most existing approaches improve robustness through optimization, clustering, distillation, or fairness objectives, whereas Mean/Std directly modifies the aggregation weights using lightweight statistics of local label distributions while preserving the simplicity and communication efficiency of standard FL.

**Table 1.** Positioning of Mean/Std with respect to representative FL, FL-IDS, and imbalance-aware approaches.

| Method | Main Focus | Non-IID | Imbalance | Label Dist. | Aggregation/ Learning Strategy | Extra Module | Lightweight |
|---|---|---|---|---|---|---|---|
| FedAvg [6] | Standard FL | Limited | No | No | Size-based averaging | None | Yes |
| FedProx [15] | Client drift | Yes | No | No | Size-based averaging + proximal loss | Proximal regularization | Yes |
| SCAFFOLD [16] | Drift correction | Yes | No | No | Variance-reduced local updates | Control variates | Moderate |
| FedNova [17] | Objective inconsistency | Yes | No | No | Normalized averaging | Update normalization | Yes |
| FedAdam/FedOpt [18] | Server opt. | Yes | No | No | Adaptive server optimization | Optimizer states | Moderate |
| AFL [29] | Distrib. robustness | Yes | Indirect | No | Min-max distribution optimization | Min–max objective | Moderate |
| q-FedAvg [30] | Client fairness | Yes | Indirect | No | Loss-sensitive client weighting | Fairness objective | Moderate |
| FADngs [23] | Anomaly det. | Yes | Partial | Partial | Federated anomaly learning + ensemble distillation | Distillation + density est. | No |
| Clustered FL [24] | IoT anomaly det. | Yes | Partial | Partial | Cluster-aware aggregation | Client clustering | No |
| FedMADE [25] | Minority attack det. | Yes | Yes | Yes | Dynamic aggregation with client grouping | Client grouping | Moderate |
| IFCEA [26] | Edge–cloud IDS | Yes | Yes | Partial | Cluster-based client weighting | Committee selection | No |
| FD-IDS [27] | Non-IID IoT IDS | Yes | Partial | Partial | Distillation-assisted aggregation | Knowledge distillation | No |
| Mean/Std (ours) | Distribution-aware FL-IDS | Yes | Yes | Yes | Label-distribution-aware weighting | Local label stats. | Yes |

'Label Dist.' indicates whether the method explicitly exploits local class-distribution information (e.g., class proportions or label skew) during learning or aggregation.

## 3. Proposed Method

This section introduces the proposed Mean/Std aggregation strategy. We first revisit the standard FL aggregation objective and discuss the limitations of conventional size-based weighting under simultaneous non-IID heterogeneity and severe multi-class imbalance [4,6,10,32]. We then derive a lightweight, distribution-aware weighting rule based on simple statistics of local label distributions. The objective is to mitigate imbalance-induced aggregation bias while preserving the simplicity, interpretability, and communication efficiency that make FedAvg attractive for large-scale IoT deployments.

### 3.1. Federated Objective and Imbalance-Induced Aggregation Bias

In federated learning, the global model is obtained by minimizing a weighted combination of local client losses:

$$\min_{\theta} F(\theta) = \sum_{i=1}^{N} \pi_i F_i(\theta), \tag{1}$$

where $F_i(\theta)$ denotes the expected loss of client $i$ and $\pi_i$ its aggregation weight. In FedAvg, these weights depend exclusively on local dataset size, i.e., $\pi_i = n_i / \sum_j n_j$, and local models (or updates) are aggregated accordingly [6]. This assumption is appropriate when client data are approximately IID, as larger datasets generally provide more reliable estimates of the underlying learning objective.

In practical IoT deployments, however, client data are inherently non-IID because of differences in device functionality, deployment conditions, user behaviour, and attack exposure. At the same time, IoT intrusion-detection datasets are typically characterized by severe multi-class imbalance, with benign traffic and frequent attack categories dominating the available observations. Under these conditions, client size alone is no longer a reliable proxy for the informativeness of a local update.

Indeed, a large client dominated by benign traffic or a limited subset of attack behaviours may primarily optimize majority-class decision boundaries. Consequently, purely size-driven aggregation can amplify dominant patterns while reducing the contribution of clients carrying scarce but operationally important attack information [4,10,32]. This limitation is particularly relevant to multi-class IoT botnet detection, where minority attack behaviours are often concentrated on only a few participating clients.

Mean/Std addresses this issue by preserving the benefits of data availability while explicitly moderating imbalance-induced aggregation bias. Rather than assuming that all large local datasets are equally informative, the proposed strategy jointly considers client size and the quality of the underlying label distribution. As a result, highly skewed clients are prevented from dominating the global update solely because they contain more samples, while clients carrying more diverse and complementary information retain an appropriate influence during aggregation.

### 3.2. Label Proportions and Imbalance Indicators

Let $n_{i,c}$ denote the number of samples of class $c$ available on client $i$, and let $n_i = \sum_{c=1}^{C} n_{i,c}$ be the corresponding local dataset size. The local label distribution is represented by the class-proportion vector

$$p_i = (p_{i,1}, \ldots, p_{i,C}), \qquad p_{i,c} = \frac{n_{i,c}}{n_i}. \tag{2}$$

Using class proportions rather than absolute frequencies provides a scale-invariant representation of local data heterogeneity, enabling meaningful comparisons across clients regardless of dataset size.

Mean/Std characterizes local imbalance through two complementary statistics derived from $p_i$. The first measures the overall deviation from a uniform class distribution:

$$\sigma(p_i) = \sqrt{\frac{1}{C}\sum_{c=1}^{C}\left(p_{i,c} - \frac{1}{C}\right)^2}, \tag{3}$$

where $\sigma(p_i) = 0$ for a perfectly balanced client and increases as the label distribution becomes more concentrated. The second statistic measures the dominance gap between the most and least represented classes:

$$\Delta(p_i) = \max_c p_{i,c} - \min_c p_{i,c}. \tag{4}$$

These two indicators capture complementary aspects of label skew: $\sigma(p_i)$ reflects the overall dispersion of class proportions, whereas $\Delta(p_i)$ quantifies extreme class dominance. Together, they provide a compact and computationally inexpensive characterization of local imbalance, well suited to lightweight FL in heterogeneous IoT environments.

### 3.3. Size Proxy and Final Weighting Rule

To preserve the contribution of data availability while accounting for the number of classes, we define the client-size proxy

$$M_i = \frac{n_i}{C}. \tag{5}$$

Client size and label-distribution quality are then combined through a penalty-controlled weighting rule. We first define the imbalance indicator

$$I_i = \sigma(p_i) + \lambda\,\Delta(p_i) + \varepsilon, \tag{6}$$

where $\lambda \geq 0$ controls the contribution of the dominance-gap term and $\varepsilon > 0$ ensures numerical stability. The resulting Mean/Std score is

$$s_i = \frac{M_i}{(\sigma(p_i) + \lambda\,\Delta(p_i) + \varepsilon)^{\alpha}}, \tag{7}$$

where $\alpha \geq 0$ controls the overall strength of the imbalance penalty. The normalized aggregation weights are then computed as

$$w_i = \frac{s_i}{\sum_{j=1}^{N} s_j}. \tag{8}$$

This formulation balances two complementary objectives: preserving the contribution of data-rich clients while moderating the influence of highly skewed local label distributions. Clients providing more samples per class and exhibiting greater label diversity naturally receive larger aggregation weights, whereas clients dominated by a limited subset of classes are progressively penalized through the imbalance term.

The weighting rule also remains fully interpretable. When imbalance levels are similar across clients, the corrective factor becomes approximately uniform and Mean/Std naturally approaches conventional size-based aggregation, recovering FedAvg-like behaviour. Conversely, substantial differences in local label distributions trigger an explicit correction that prevents large but highly skewed clients from disproportionately influencing the global update.

The resulting mechanism preserves the computational and communication characteristics of conventional FL while introducing an explicit distribution-aware correction for imbalance-induced aggregation bias.

Table 2 shows that the additional communication overhead introduced by Mean/Std is negligible. Beyond the standard model update, each client transmits only three scalar quantities describing its local label distribution: the local dataset size $(n_i)$, the standard deviation of class proportions $(\sigma(p_i))$, and the dominance gap $(\Delta(p_i))$. No auxiliary models, clustering procedures, contribution-estimation modules, or additional communication rounds are required. Consequently, Mean/Std preserves the lightweight and easily deployable nature of standard federated learning while explicitly accounting for imbalance-induced aggregation bias.

**Table 2.** Additional communication requirements of Mean/Std compared with standard FedAvg.

| Item | FedAvg | Mean/Std |
|---|---|---|
| Model update transmission | Yes | Yes |
| Additional local statistics | None | $n_i$, $\sigma(p_i)$, $\Delta(p_i)$ |
| Extra scalar values per client/round | 0 | 3 |
| Additional communication rounds | No | No |
| Auxiliary models or clustering | No | No |

### *3.4. Algorithm Overview and Deployment Notes*

Figure 2 illustrates the integration of Mean/Std into a standard federated learning workflow. The proposed method does not modify local training or optimization; instead, it augments the aggregation stage by incorporating lightweight statistics of the local label distribution.

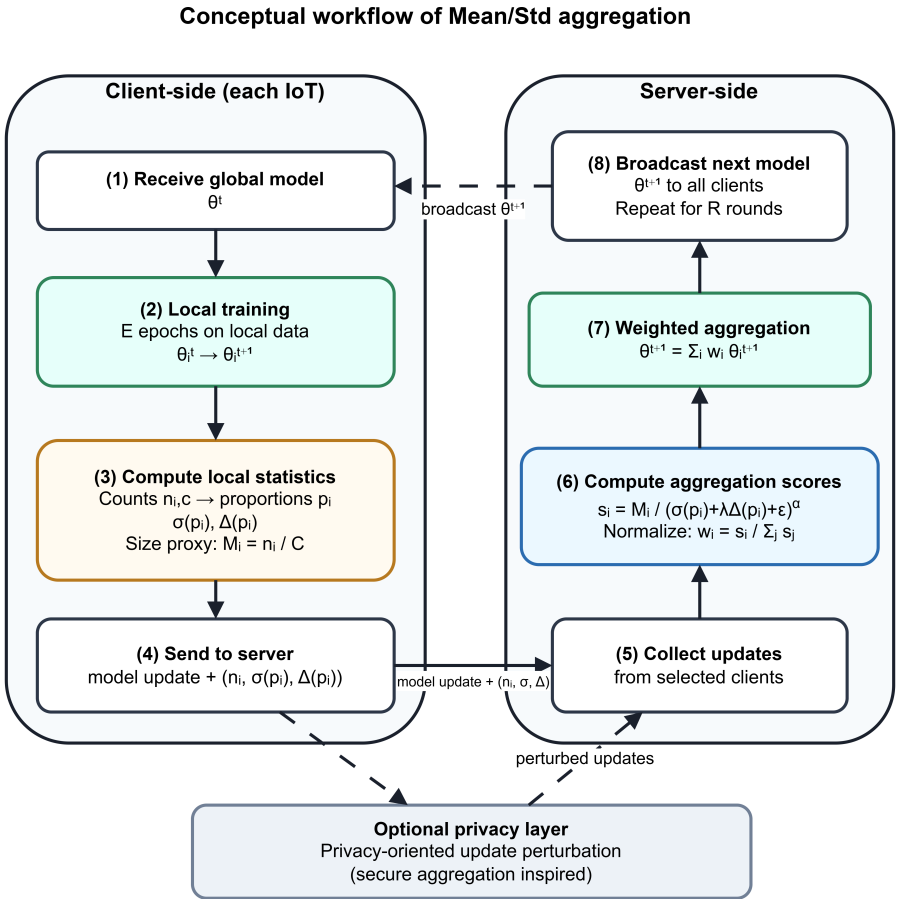


**Figure 2.** Conceptual workflow of a federated learning round using Mean/Std aggregation. Arrows indicate the sequence of operations and the exchange of model updates and lightweight statistics between clients and the server.

After local training, each client computes $(n_i, \sigma(p_i), \Delta(p_i))$ from its local dataset and transmits these values together with the standard model update. The server then computes the normalized aggregation weights according to Equation (8) and performs weighted model aggregation.

Unlike many approaches designed for heterogeneous FL, Mean/Std does not require auxiliary models, clustering procedures, contribution-estimation mechanisms, fairness objectives, or additional communication rounds. The aggregation rule relies exclusively on information already available locally, namely dataset size and simple statistics of the label distribution, making the method lightweight, interpretable, and straightforward to integrate into existing FL deployments.

In practice, a small weight floor $\eta$ is applied before the final normalization step to avoid near-zero client contributions under extreme imbalance conditions. This safeguard improves aggregation stability while ensuring that clients carrying rare but operationally important attack behaviours retain a minimum influence on the global update. Consequently, Mean/Std preserves the simplicity of standard FL while explicitly mitigating imbalance-induced aggregation bias in heterogeneous IoT environments.

### 3.5. Novelty and Positioning

Mean/Std addresses a specific limitation of conventional FL aggregation under simultaneous non-IID heterogeneity and severe multi-class imbalance. Although several FL methods improve robustness under heterogeneous data distributions, they generally target failure modes different from the imbalance-induced aggregation bias considered in this work.

FedProx [15], SCAFFOLD [16], FedNova [17], and FedAdam [18] primarily focus on optimization stability, client-drift mitigation, or convergence improvement. Likewise, fairness-oriented approaches such as AFL [29] and q-FedAvg [30] modify optimization objectives to reduce disparities across clients. Although effective in their respective settings, these methods do not explicitly exploit local label-distribution characteristics during aggregation.

In contrast, Mean/Std operates directly at the aggregation stage. Client influence is determined using only three lightweight quantities derived from local data: dataset size, the standard deviation of class proportions, and the dominance gap between the most and least represented classes. Unlike optimization-oriented, fairness-oriented, clustering-based, or distillation-assisted approaches, Mean/Std requires neither modifications to local optimization nor auxiliary learning components or additional communication rounds. It therefore preserves the simplicity and communication efficiency of FedAvg while explicitly addressing aggregation bias induced by heterogeneous label distributions.

Table 3 summarizes the conceptual differences between Mean/Std and representative FL baselines.

**Table 3.** Conceptual positioning of Mean/Std with respect to representative FL aggregation strategies.

| Method | Uses Client Size | Uses Label Distribution | Primary Mechanism |
|---|---|---|---|
| FedAvg [6] | Yes | No | Size-based aggregation |
| FedProx [15] | Yes | No | Proximal regularization |
| FedNova [17] | Yes | No | Update normalization |
| AFL [29] | No | No | Min–max optimization |
| q-FedAvg [30] | Partial | No | Fairness-oriented aggregation |
| Mean/Std (ours) | Yes | Yes | Distribution-aware aggregation |

### 3.6. Why Mean/Std Is Not a Class-Balanced FedAvg

A potential concern is that Mean/Std resembles a class-balanced variant of FedAvg. Although both approaches aim to mitigate the effects of data imbalance, they operate at fundamentally different levels.

Class-balanced extensions of FedAvg typically address imbalance through class-frequency corrections, sample reweighting, or modified loss functions, thereby increasing the influence of minority classes during training. Mean/Std, however, performs neither class-level reweighting, resampling, nor loss-function modification.

Instead, it characterizes the overall shape of each client's label distribution using two lightweight statistics: the standard deviation and the dominance gap of local class proportions. Aggregation weights are therefore determined by a combination of data availability and distribution quality rather than explicit class-frequency corrections. Consequently, two clients with identical dataset sizes may receive different aggregation weights if their label distributions exhibit different levels of skewness.

This distinction is particularly relevant in IoT federated environments, where heterogeneity arises from device-specific traffic patterns, unequal attack exposure, and severe label skew. Unlike class-balancing approaches operating at the sample or loss level, Mean/Std acts exclusively at the client-aggregation level. Its objective is not to rebalance classes directly, but to moderate the influence of highly skewed clients so that minority attack information remains represented in the global update. In this way, Mean/Std addresses a broader aggregation problem while preserving the simplicity, communication efficiency, and local optimization procedure of standard FedAvg.

## 4. Experimental Setup and Methodology

### 4.1. Dataset, Label Space, and Clients

Experiments are conducted on the N-BaIoT benchmark, a widely adopted dataset for IoT botnet detection comprising real network traffic collected from commercial IoT devices under both benign operation and infections by the Mirai and Bashlite (Gafgyt) malware families [11]. Owing to its heterogeneous devices, multiple attack categories, and naturally imbalanced traffic distributions, N-BaIoT provides a realistic testbed for studying aggregation behaviour in non-IID federated learning environments.

Importantly, N-BaIoT was not designed as a class-balanced machine learning benchmark. Instead, it captures benign traffic during normal device operation together with malicious traffic generated after botnet infection [11]. As a result, class frequencies naturally reflect device-specific behaviour, traffic volume, attack characteristics, and heterogeneous attack exposure across devices. The resulting imbalance is therefore an intrinsic property of the dataset rather than an artifact of the experimental protocol, closely matching practical IoT security deployments.

IoT botnet detection is formulated as an 11-class classification task comprising benign traffic and ten attack categories associated with Mirai and Gafgyt behaviours. Compared with binary detection, this multi-class formulation provides a more informative evaluation of aggregation strategies under realistic class-imbalance conditions.

Seven devices are selected as FL clients, denoted $IoT_1$, $IoT_2$, $IoT_4$, $IoT_5$, $IoT_6$, $IoT_8$, and $IoT_9$. Each device retains its local dataset throughout training, naturally inducing heterogeneous client distributions. All clients participate in every communication round (client fraction = 1.0), thereby isolating the effect of the aggregation strategy from partial-participation phenomena.

Figure 3 presents the class distribution associated with each client. The substantial differences in attack composition and class prevalence across devices reveal pronounced

label skew and heterogeneous attack exposure, providing a representative non-IID setting for evaluating distribution-aware aggregation methods in FL-based IoT botnet detection.

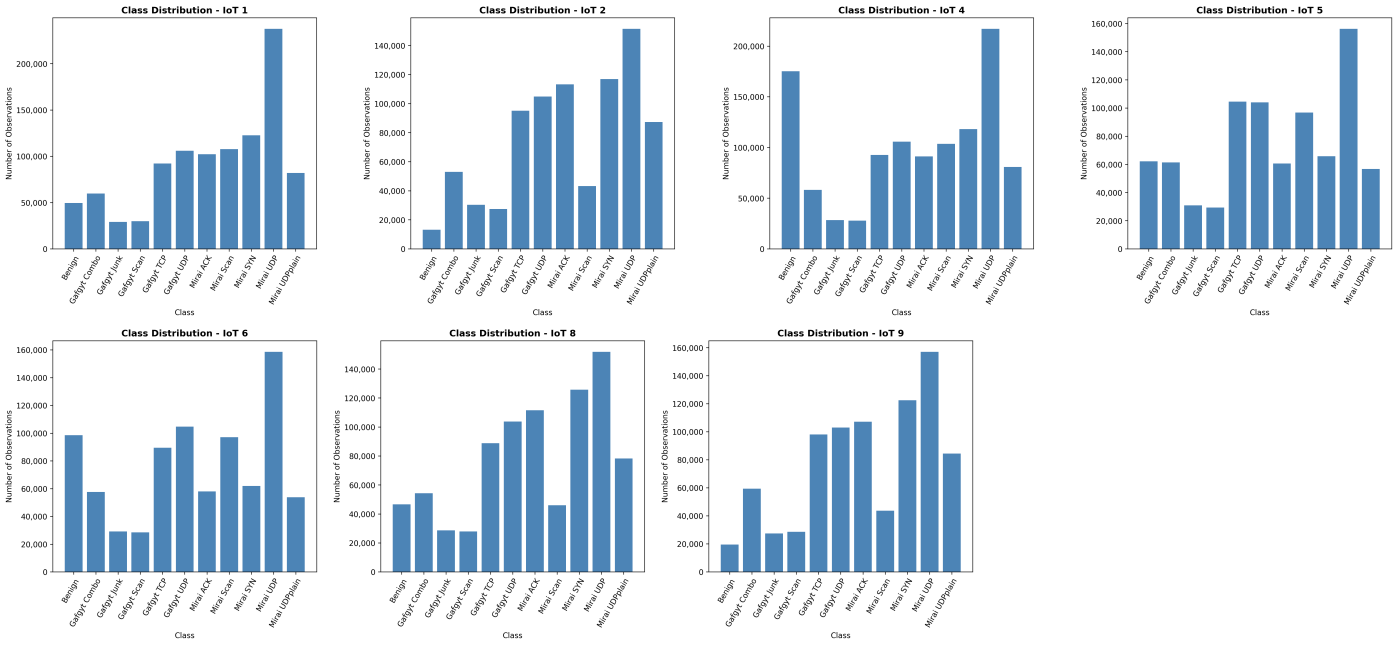


**Figure 3.** Class distribution for each IoT client. The pronounced differences across clients reflect multi-class imbalance and non-IID label skew, which can bias size-weighted aggregation toward majority behaviors.

### *4.2. Data Splitting and Preprocessing*

All experiments use the complete N-BaIoT data associated with the selected devices. To ensure a fair and reproducible comparison across distributed learning, conventional FL baselines, and the proposed Mean/Std aggregation strategy, an identical partitioning and preprocessing protocol is adopted for all methods.

For each client, the local dataset is split into 70% training, 15% validation, and 15% testing subsets using stratified sampling. This client-wise partitioning preserves the original local class distributions while preventing information leakage across clients. Validation subsets are used for model monitoring and hyperparameter selection, whereas test subsets remain strictly held out for the final evaluation. Global metrics are computed by aggregating predictions over the union of all client test sets.

Before training, duplicate records are removed and labels are mapped to a common set of class identifiers shared across the federation. Feature normalization is performed using a global StandardScaler fitted exclusively on the union of all client training partitions and subsequently applied to the corresponding training, validation, and test subsets of every client. Restricting the fitting stage to training data prevents data leakage while ensuring a consistent feature space across the federation.

### *4.3. Local Model, Baselines, and Training Configuration*

To ensure a controlled comparison, all methods employ the same lightweight one-dimensional convolutional neural network (1D CNN) as the local classifier. Consequently, any observed performance differences can be attributed to the learning paradigm or aggregation strategy rather than to variations in model architecture.

The experimental study considers seven representative learning configurations. Distributed Learning serves as a non-collaborative baseline in which each client trains an independent local model without exchanging information. The federated baselines include

FedAvg [6], which performs conventional size-based aggregation; FedProx [15], which mitigates client drift through proximal regularization; FedAdam (FedOpt) [18], which improves convergence through adaptive server-side optimization; FedNova [17], which normalizes local updates to reduce objective inconsistency; and q-FedAvg [30], which introduces fairness-aware aggregation through loss-sensitive weighting. These methods are compared with the proposed Mean/Std strategy, which adjusts client influence using lightweight statistics derived from local label distributions.

To isolate the effect of the aggregation mechanism, all approaches share the same data partitions, preprocessing pipeline, local architecture, and training configuration. The only differences arise from the learning paradigm and the aggregation or optimization strategy, enabling a controlled evaluation of aggregation behaviour under non-IID and imbalanced IoT data. The common experimental settings are summarized in Table 4.

**Table 4.** Common experimental configuration used across all compared methods.

| Parameter | Value |
|---|---|
| IoT clients | 7 |
| Traffic classes | 11 |
| Communication rounds | 60 |
| Local epochs | 10 |
| Client fraction | 1.0 |
| Batch size | 64 |
| Learning rate | $3 \times 10^{-4}$ |
| Label smoothing | 0.05 |
| Class weighting | Enabled |
| Gaussian perturbation variance | $\sigma_n = 10^{-5}$ |
| Random seeds | {7, 21, 42, 2025, 3407} |

### *4.4. Privacy-Oriented Perturbation Setting*

Although federated learning (FL) avoids the direct exchange of raw network traffic, it does not inherently guarantee privacy. Model updates may still leak information about local data and remain vulnerable to inference or gradient-reconstruction attacks [7,31]. In practical FL deployments, additional protection mechanisms such as secure aggregation, differential privacy, or update masking are therefore commonly employed [12].

To evaluate aggregation robustness under privacy-constrained conditions, all experiments are conducted with `USE_SECURE_AGG=True`, which activates a lightweight perturbation mechanism applied to client updates before aggregation. After local training, client $i$ computes a model update $\Delta\theta_i$, which is perturbed before transmission according to

$$\widetilde{\Delta\theta}_i = \Delta\theta_i + \varepsilon_i, \tag{9}$$

where

$$\varepsilon_i \sim \mathcal{N}(0, \sigma_n^2), \tag{10}$$

with $\sigma_n = 10^{-5}$ in the main experiments. The Gaussian perturbation is applied independently to each model-update parameter and identically across all compared methods.

This mechanism is not intended to provide formal differential privacy guarantees or to implement a cryptographic secure aggregation protocol. Rather, it constitutes a controlled, secure-aggregation-inspired robustness setting that emulates the optimization uncertainty commonly introduced by privacy-preserving aggregation pipelines. Because all methods are evaluated under the same perturbation process, performance differences can be attributed to the aggregation strategy itself rather than to differences in privacy protection.

Figure 4 illustrates the overall workflow. Local training remains entirely on-device, raw traffic never leaves the client, and only perturbed model updates are transmitted to the aggregation server.

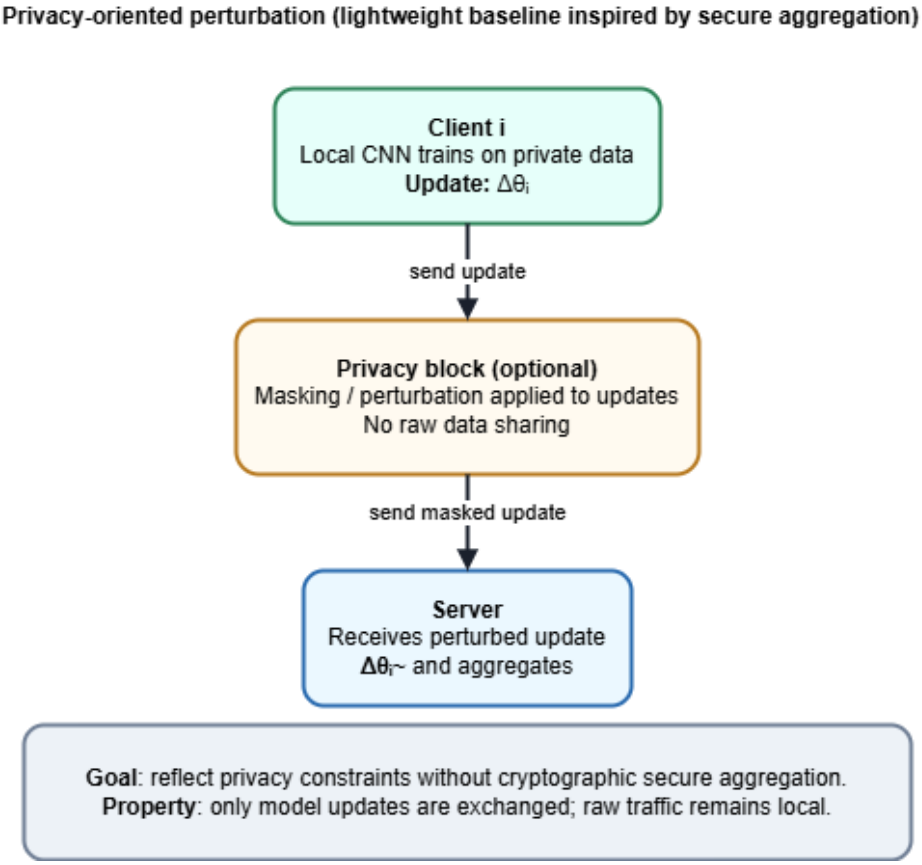


**Figure 4.** Conceptual illustration of the privacy-oriented update perturbation mechanism used in this work. The approach is inspired by secure aggregation workflows but does not provide cryptographic privacy guarantees.

Accordingly, this setting should be interpreted as a privacy-oriented robustness evaluation rather than a formal privacy mechanism. Combining formal privacy guarantees with utility–privacy trade-off analysis remains an important direction for future work.

### *4.5. Evaluation Metrics and Statistical Protocol*

Performance is assessed using complementary metrics designed for severe multi-class imbalance. In addition to overall *Accuracy*, we report *Balanced Accuracy*, *Macro Precision*, *Macro Recall*, *Macro-F1*, *Weighted-F1*, macro-averaged *ROC-AUC*, and macro-averaged *PR-AUC*. Unlike accuracy alone, these metrics explicitly account for unequal class frequencies and therefore provide a more informative evaluation of minority attack detection [4,5].

Macro-averaged metrics assign equal importance to all classes regardless of prevalence, making them particularly suitable for IoT botnet detection, where rare attack behaviours may be operationally more critical than dominant traffic patterns. Accordingly, Macro-F1 and Balanced Accuracy are adopted as the primary evaluation criteria because they jointly capture minority-class sensitivity and overall classification performance. Weighted-F1, ROC-AUC, and PR-AUC are additionally reported to provide complementary perspectives on classification quality and ranking performance under imbalanced conditions.

To reduce the influence of stochastic optimization, all experiments are repeated using five independent random seeds: {7, 21, 42, 2025, and 3407}. The same seed controls network initialization, data shuffling, and mini-batch generation across all compared methods,

ensuring a fair and reproducible evaluation protocol. Unless otherwise stated, reported results correspond to the mean performance across the five runs, while the associated standard deviations are used to assess experimental stability and the robustness of the observed differences.

### 4.6. Experimental Workflow and Algorithmic Implementation

All evaluated approaches follow a common experimental workflow to ensure a fair and reproducible comparison under identical learning conditions. At the beginning of each communication round, participating clients receive the current global model (or retain their own local model in the case of Distributed Learning), perform local optimization on their private training data for a fixed number of local epochs, and compute updated model parameters. The server then collects the resulting client updates and generates the model used for the next communication round according to the aggregation or optimization strategy under consideration.

To isolate the effect of the learning paradigm itself, all methods share the same local CNN architecture, data partitions, preprocessing pipeline, optimizer configuration, training hyperparameters, and privacy-oriented perturbation setting. Consequently, any observed performance differences can be attributed to the aggregation or optimization mechanism rather than to variations in model capacity or implementation details. Distributed Learning trains independent local models without communication, whereas FedAvg [6], FedProx [15], FedAdam/FedOpt [18], FedNova [17], q-FedAvg [30], and the proposed Mean/Std operate within the same FL framework while adopting different server-side update rules.

For reproducibility, Algorithm 1 summarizes the implementation of the proposed Mean/Std aggregation strategy. The remaining baselines follow their original formulations [6,15,17,18,30] while preserving the common experimental protocol described above. This unified implementation framework ensures that the comparison focuses exclusively on the impact of aggregation design under realistic non-IID and imbalanced IoT botnet detection conditions.

**Algorithm 1** Mean/Std aggregation strategy

Initialize global model $\theta^{(0)}$
**for** communication round $t = 0, \ldots, R-1$ **do**
  Broadcast $\theta^{(t)}$ to all participating clients
  **for** each client $i$ in parallel **do**
    Perform local training for $E$ epochs to obtain $\theta_i^{(t+1)}$
    Compute local class counts $\{n_{i,c}\}_{c=1}^{C}$ and $n_i = \sum_c n_{i,c}$
    Compute label proportions $p_{i,c} = n_{i,c}/n_i$
    Compute imbalance indicators $\sigma(p_i)$ and $\Delta(p_i)$
    Apply the privacy-oriented perturbation mechanism described in Section 4.4
    Send $\theta_i^{(t+1)}$ and $(n_i, \sigma(p_i), \Delta(p_i))$ to the server
  **end for**
  Compute the size proxy $M_i = n_i/C$
  Compute scores:

$$s_i = \frac{M_i}{(\sigma(p_i) + \lambda\Delta(p_i) + \varepsilon)^{\alpha}}$$

  Normalize aggregation weights:

$$w_i = \frac{s_i}{\sum_{j=1}^{N} s_j}$$

  Apply the weight floor $\eta$ and renormalize
  Aggregate client models:

$$\theta^{(t+1)} \leftarrow \sum_{i=1}^{N} w_i \, \theta_i^{(t+1)}$$

**end for**

## 5. Results and Analysis

This section evaluates the proposed Mean/Std aggregation strategy under realistic non-IID and class-imbalanced IoT botnet detection conditions. We first compare Mean/Std with isolated Distributed Learning and representative FL baselines, including FedAvg, FedProx, FedAdam, FedNova, and q-FedAvg. We then analyze its impact on minority attack detection, assess the robustness of the observed gains across multiple random seeds, and investigate hyperparameter sensitivity through an ablation study.

Unless otherwise stated, all reported quantitative results correspond to the mean and standard deviation obtained over five independent random seeds ({7, 21, 42, 2025, 3407}). This protocol provides a more reliable assessment of performance under stochastic optimization and heterogeneous FL conditions.

Given the severe class imbalance inherent to IoT intrusion detection, the analysis emphasizes imbalance-aware metrics rather than overall accuracy alone. In particular, Macro-F1 and Balanced Accuracy serve as the primary evaluation criteria, while ROC-AUC and PR-AUC provide complementary insights into discriminative and ranking performance.

### *5.1. Overall Performance Comparison*

Table 5 reports the average performance of the evaluated learning paradigms under the common experimental protocol described in Section 4. The comparison includes isolated Distributed Learning, standard FL baselines (FedAvg, FedProx, FedAdam, and FedNova), the fairness-oriented q-FedAvg approach, and the proposed Mean/Std aggregation strategy. All values are reported as mean ± standard deviation across five independent random seeds, allowing both predictive performance and run-to-run stability to be assessed simultaneously.

Because multi-class IoT botnet detection is characterized by severe label imbalance, overall Accuracy alone may provide an incomplete assessment of model quality. Accordingly, particular attention is given to Macro-F1 and Balanced Accuracy, which more faithfully reflect the ability of a learning algorithm to preserve minority attack recognition while maintaining robust overall performance [4,5].

**Table 5.** Overall performance comparison across the evaluated learning paradigms. Reported values correspond to the mean and standard deviation over five independent random seeds. The best mean in each column is highlighted in bold.

| Method | Acc. | Bal. Acc. | M. Prec. | M. Recall | M.-F1 | W.-F1 | ROC-AUC | PR-AUC |
|---|---|---|---|---|---|---|---|---|
| Distrib. | 0.8036 ± 0.0053 | 0.8321 ± 0.0042 | 0.8218 ± 0.0045 | 0.8321 ± 0.0042 | 0.8130 ± 0.0076 | 0.8019 ± 0.0110 | **0.9844** ± 0.0004 | 0.8538 ± 0.0045 |
| FedAvg | 0.8085 ± 0.0101 | 0.8449 ± 0.0060 | 0.7909 ± 0.0051 | 0.8449 ± 0.0060 | 0.8058 ± 0.0049 | 0.7723 ± 0.0115 | 0.9766 ± 0.0009 | 0.8599 ± 0.0036 |
| FedProx | 0.7777 ± 0.0221 | 0.8092 ± 0.0179 | 0.7558 ± 0.0140 | 0.8092 ± 0.0179 | 0.7638 ± 0.0193 | 0.7397 ± 0.0257 | 0.9739 ± 0.0013 | 0.8272 ± 0.0121 |
| FedAdam | 0.4642 ± 0.1424 | 0.5961 ± 0.1471 | 0.3154 ± 0.1402 | 0.5961 ± 0.1471 | 0.3985 ± 0.1594 | 0.3189 ± 0.1377 | 0.6790 ± 0.1004 | 0.4917 ± 0.0926 |
| FedNova | 0.8023 ± 0.0085 | 0.8367 ± 0.0069 | 0.7915 ± 0.0095 | 0.8367 ± 0.0069 | 0.7985 ± 0.0052 | 0.7659 ± 0.0096 | 0.9764 ± 0.0009 | 0.8553 ± 0.0056 |
| q-FedAvg | 0.8077 ± 0.0137 | 0.8397 ± 0.0100 | 0.7927 ± 0.0064 | 0.8397 ± 0.0100 | 0.8014 ± 0.0090 | 0.7718 ± 0.0152 | 0.9764 ± 0.0007 | 0.8567 ± 0.0026 |
| **Mean/Std** | **0.8442** ± 0.0097 | **0.8722** ± 0.0035 | **0.8655** ± 0.0397 | **0.8722** ± 0.0035 | **0.8418** ± 0.0054 | **0.8086** ± 0.0102 | 0.9800 ± 0.0008 | **0.8726** ± 0.0030 |

Values are reported as mean ± standard deviation over five independent random seeds. The best mean in each column is shown in bold.

The results show that Mean/Std consistently achieves the strongest imbalance-aware performance among the evaluated methods. In particular, it attains the highest Macro-F1 and Balanced Accuracy, indicating that incorporating local label-distribution information

into the aggregation process improves the representation of minority attack behaviours while preserving robust overall classification performance. The relatively small standard deviations reported in Table 5 further indicate that the observed gains remain stable across independent executions and different random initializations.

Relative to conventional size-based aggregation, Mean/Std improves Macro-F1 from 0.8058 to 0.8418 and Balanced Accuracy from 0.8449 to 0.8722. It also achieves the highest Macro Precision, Weighted-F1, and PR-AUC, demonstrating that the observed benefits extend across multiple complementary evaluation criteria rather than being confined to a single metric. Importantly, these gains are obtained without modifying the local model architecture or introducing additional communication rounds.

Another noteworthy observation is that Mean/Std remains competitive against both optimization-oriented and fairness-oriented FL approaches. While FedProx, FedAdam, and FedNova primarily target optimization stability under heterogeneous data, and q-FedAvg focuses on client-level fairness, none of these methods explicitly address aggregation bias induced by highly skewed local label distributions. The superior performance of Mean/Std therefore supports the central hypothesis of this work: under severe non-IID heterogeneity and multi-class imbalance, redesigning the aggregation rule itself can provide greater benefits than optimization improvements alone.

Finally, the simultaneous improvement in Macro-F1, Balanced Accuracy, and PR-AUC, together with competitive Accuracy and Weighted-F1, suggests that enhanced minority attack recognition is not achieved at the expense of majority-class performance. Instead, the proposed distribution-aware weighting mechanism yields a more balanced global model that better captures the diversity of attack behaviours across heterogeneous IoT clients.

### *5.2. Impact on Minority Attack Detection*

While the aggregate metrics reported in Table 5 provide an overall assessment of model performance, they do not explicitly reveal how individual attack categories are affected by the aggregation strategy. This distinction is particularly important for IoT botnet detection, where attack classes are unevenly distributed across devices and may become severely underrepresented under realistic non-IID conditions. To complement the global analysis, Table 6 reports the class-wise F1-score obtained by each method.

**Table 6.** Per-class F1-score for the eleven traffic categories.

| Method | Benign | G.Combo | G.Junk | G.Scan | G.TCP | G.UDP | M.ACK | M.Scan | M.SYN | M.UDP | M.UDPp |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Distrib. | 0.9960 ± 0.0002 | 0.7301 ± 0.0081 | 0.6817 ± 0.0087 | 0.9855 ± 0.0012 | 0.4974 ± 0.0596 | 0.4677 ± 0.1275 | 0.7605 ± 0.0122 | **0.9966** ± 0.0005 | **0.9979** ± 0.0002 | 0.8419 ± 0.0080 | **0.9872** ± 0.0025 |
| FedAvg | 0.9933 ± 0.0017 | 0.9328 ± 0.0154 | 0.8883 ± 0.0217 | 0.9923 ± 0.0021 | **0.5150** ± 0.2575 | 0.1376 ± 0.2752 | 0.7902 ± 0.0594 | 0.9766 ± 0.0052 | 0.9841 ± 0.0036 | 0.8574 ± 0.0196 | 0.7965 ± 0.0156 |
| FedProx | 0.9922 ± 0.0011 | 0.7840 ± 0.0289 | 0.7373 ± 0.0203 | 0.9863 ± 0.0026 | 0.3861 ± 0.3153 | 0.2752 ± 0.3370 | 0.6484 ± 0.1145 | 0.9734 ± 0.0016 | 0.9822 ± 0.0011 | 0.8271 ± 0.0330 | 0.8097 ± 0.0175 |
| FedAdam | 0.5853 ± 0.1832 | 0.5224 ± 0.0717 | 0.5700 ± 0.0660 | 0.5168 ± 0.0792 | 0.5064 ± 0.0258 | **0.5591** ± 0.0540 | 0.0000 ± 0.0000 | 0.6101 ± 0.3174 | 0.4983 ± 0.0901 | 0.5775 ± 0.0280 | 0.5392 ± 0.0851 |
| FedNova | 0.9916 ± 0.0020 | 0.9234 ± 0.0133 | 0.8754 ± 0.0183 | 0.9909 ± 0.0032 | 0.3862 ± 0.3152 | 0.2752 ± 0.3370 | 0.7419 ± 0.0584 | 0.9766 ± 0.0019 | 0.9845 ± 0.0014 | 0.8471 ± 0.0172 | 0.7912 ± 0.0271 |
| q-FedAvg | 0.9915 ± 0.0014 | 0.9223 ± 0.0110 | 0.8737 ± 0.0153 | 0.9909 ± 0.0032 | 0.2575 ± 0.3153 | 0.4128 ± 0.3370 | 0.7630 ± 0.0557 | 0.9745 ± 0.0028 | 0.9830 ± 0.0017 | 0.8559 ± 0.0251 | 0.7906 ± 0.0207 |
| **Mean/Std** | **0.9970** ± 0.0005 | **0.9946** ± 0.0015 | **0.9882** ± 0.0027 | **0.9980** ± 0.0005 | 0.3864 ± 0.3153 | 0.2753 ± 0.3371 | **0.9008** ± 0.0111 | 0.9919 ± 0.0021 | 0.9947 ± 0.0015 | **0.9044** ± 0.0080 | 0.8282 ± 0.0190 |

Values are reported as mean ± standard deviation over five independent random seeds. The best mean in each column is shown in bold.

The reported values correspond to the average F1-score across five independent random seeds ({7, 21, 42, 2025, 3407}), enabling a direct assessment of whether the observed improvements are distributed across the label space or primarily driven by majority classes.

To improve readability and consistency with the global evaluation, Table 6 reports both the mean and standard deviation of the per-class F1-score across the five runs.

The per-class analysis provides additional insight into the aggregate improvements observed in Table 5. Rather than simply improving already dominant traffic categories, Mean/Std consistently enhances the recognition of several attack classes that are particularly sensitive to non-IID fragmentation and aggregation bias. The most notable gains are observed for *gafgyt.combo*, *gafgyt.junk*, *mirai.ack*, and *mirai.udp*, where Mean/Std achieves the highest F1-score among all evaluated FL baselines. Moreover, the generally low standard deviations observed for the best-performing classes indicate that these improvements are consistently reproduced across different random initializations.

These results are consistent with the design objective of the proposed aggregation strategy. Under conventional size-based aggregation, updates originating from clients dominated by benign traffic or a limited subset of attack categories can disproportionately influence the global model, reducing the contribution of clients carrying rarer but complementary attack patterns. By incorporating lightweight statistics of local label distributions into the aggregation process, Mean/Std explicitly moderates this imbalance-induced dominance and enables minority-class information to contribute more effectively during global model construction.

Importantly, these improvements are not obtained at the expense of the remaining classes. Mean/Std maintains competitive performance across the full label space while substantially improving several underrepresented attack categories. Consequently, the gains observed in Macro-F1 and Balanced Accuracy reflect a more balanced representation of the overall multi-class IoT botnet detection problem rather than an optimization biased toward a limited subset of classes.

### 5.3. Statistical Robustness Across Random Seeds

The robustness analysis complements the mean ($\pm$std) reporting already provided in Tables 5 and 6 by explicitly quantifying variability and statistical significance across independent runs.

To reduce the influence of stochastic optimization and provide a more reliable assessment of model behaviour, all experiments were repeated using five independent random seeds ({7, 21, 42, 2025, 3407}). Unless otherwise specified, the reported results correspond to the average over these runs, providing a rigorous evaluation of reproducibility under heterogeneous FL conditions.

Table 7 provides a compact summary of the two primary imbalance-aware metrics, namely Balanced Accuracy and Macro-F1. Although these values are also included in Table 5, this focused table is retained to emphasize run-to-run stability on the main evaluation criteria used throughout the analysis.

**Table 7.** Performance stability across five independent random seeds.

| | Balanced Accuracy | | Macro-F1 | |
|---|---|---|---|---|
| **Method** | **Mean** | **Std** | **Mean** | **Std** |
| Distributed | 0.8321 | 0.0042 | 0.8130 | 0.0076 |
| FedAvg | 0.8449 | 0.0060 | 0.8058 | 0.0049 |
| FedProx | 0.8092 | 0.0179 | 0.7638 | 0.0193 |
| FedAdam | 0.5961 | 0.1471 | 0.3985 | 0.1594 |
| FedNova | 0.8367 | 0.0069 | 0.7985 | 0.0052 |
| q-FedAvg | 0.8397 | 0.0100 | 0.8014 | 0.0090 |
| **Mean/Std (ours)** | **0.8722** | **0.0035** | **0.8418** | **0.0054** |

Values summarize the two primary metrics over five independent random seeds. The best mean and corresponding standard deviation are shown in bold.

Compared with the conventional FedAvg baseline, Mean/Std improves the average Macro-F1 from 0.8058 to 0.8418 and the average Balanced Accuracy from 0.8449 to 0.8722 while maintaining a similarly low level of variability across the five runs. In contrast, optimization-oriented approaches such as FedProx and especially FedAdam exhibit substantially larger fluctuations, indicating greater sensitivity to stochastic optimization under heterogeneous client distributions.

Table 8 summarizes the statistical comparison between Mean/Std and the conventional FedAvg baseline across the five independent random seeds. To complement the mean–standard deviation analysis, paired comparisons were performed using both a paired $t$-test and the non-parametric Wilcoxon signed-rank test.

**Table 8.** Statistical comparison between Mean/Std and the conventional FedAvg baseline across five independent random seeds. Reported $p$-values are obtained from paired comparisons of the seed-wise results.

| Metric | Comparison | Paired $t$-Test $p$-Value | Wilcoxon $p$-Value |
|---|---|---|---|
| Balanced Accuracy | Mean/Std > FedAvg | $p < 0.01$ | $p < 0.05$ |
| Macro-F1 | Mean/Std > FedAvg | $p < 0.01$ | $p < 0.05$ |
| PR-AUC | Mean/Std > FedAvg | $p < 0.01$ | $p < 0.05$ |

The statistical analysis is consistent with the stability results and confirms that the observed improvements are not attributable to favorable random initializations. Across all five independent runs, Mean/Std consistently outperforms the conventional FedAvg aggregation strategy, with statistically significant gains according to both parametric and non-parametric tests.

From a practical perspective, robustness to stochastic variations is particularly important in federated IoT environments, where non-IID client distributions and local optimization dynamics can amplify run-to-run variability. The low dispersion and consistent superiority of Mean/Std therefore provide additional evidence that lightweight statistics of local label distributions can improve imbalance-aware aggregation while preserving a stable and reproducible training process.

### 5.4. Hyperparameter Sensitivity Analysis

The proposed Mean/Std aggregation strategy introduces two complementary hyperparameters that regulate the influence of local label-distribution statistics during aggregation. The coefficient $\lambda$ controls the contribution of the dominance-gap term $\Delta(p_i)$, whereas the exponent $\alpha$ determines the overall strength of the imbalance penalty. To assess the robustness of the proposed formulation, we conduct a series of controlled ablation experiments by varying these parameters while monitoring Macro-F1 and Balanced Accuracy.

Figure 5 illustrates the influence of the imbalance-penalty exponent $\alpha$ for representative fixed values of $\lambda$. Across the explored range, performance remains relatively stable, indicating that Mean/Std is not overly sensitive to moderate variations of the penalty exponent. Intermediate values of $\alpha$ generally provide the best trade-off between preserving the contribution of data-rich clients and mitigating imbalance-induced aggregation bias, whereas excessively large values tend to over-penalize highly skewed clients.

The complementary analysis is shown in Figure 6, where $\lambda$ is varied while $\alpha$ remains fixed. Similar trends are observed, with only moderate performance variations across a broad range of $\lambda$ values. Small and intermediate values generally provide the best compromise between exploiting global distributional information and limiting the influence of extreme class dominance during aggregation.

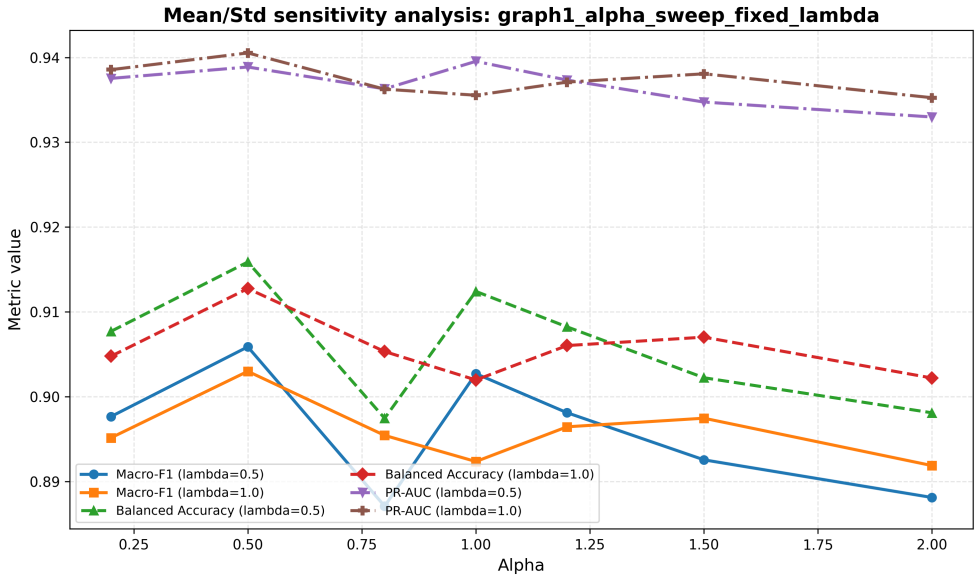


**Figure 5.** Influence of the imbalance-penalty exponent $\alpha$ on Macro-F1 and Balanced Accuracy for fixed values of $\lambda$.

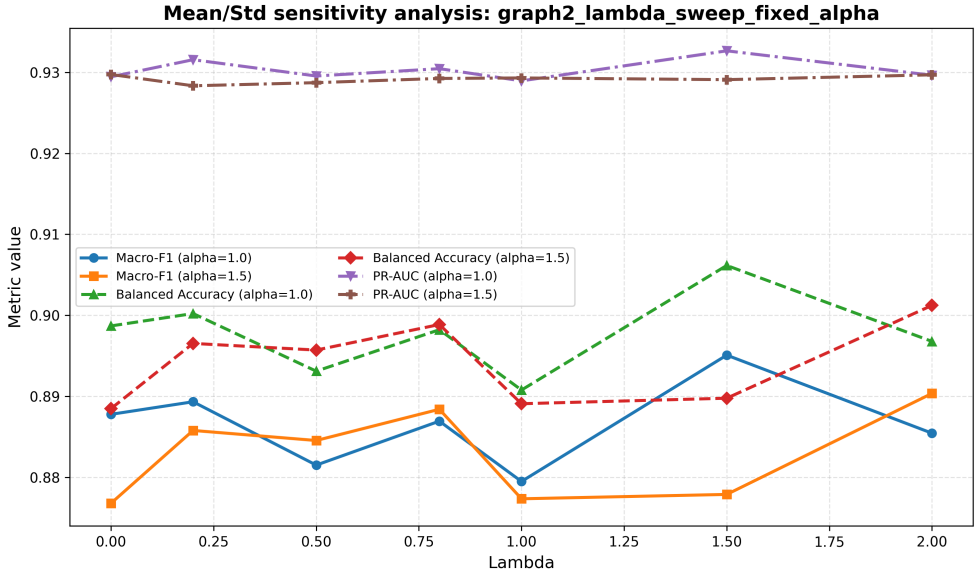


**Figure 6.** Influence of the dominance-gap coefficient $\lambda$ on Macro-F1 and Balanced Accuracy for fixed values of $\alpha$.

To further examine the interaction between the two hyperparameters, Figure 7 presents the performance landscape over the explored $(\lambda, \alpha)$ parameter space. The heatmap reports the mean Macro-F1 obtained for each evaluated configuration and provides a global view of the relationship between the two control parameters.

Rather than exhibiting a narrow isolated optimum, the heatmap reveals a broad high-performance region spanning low $\lambda$ and low-to-moderate $\alpha$ values. This observation indicates that Mean/Std remains effective under moderate deviations from the nominal configuration and does not rely on finely tuned hyperparameters. The analysis also highlights the complementary roles of the two parameters: $\alpha$ primarily controls the overall strength of the imbalance correction, whereas $\lambda$ adjusts the contribution of extreme class dominance through the range statistic $\Delta(p_i)$.

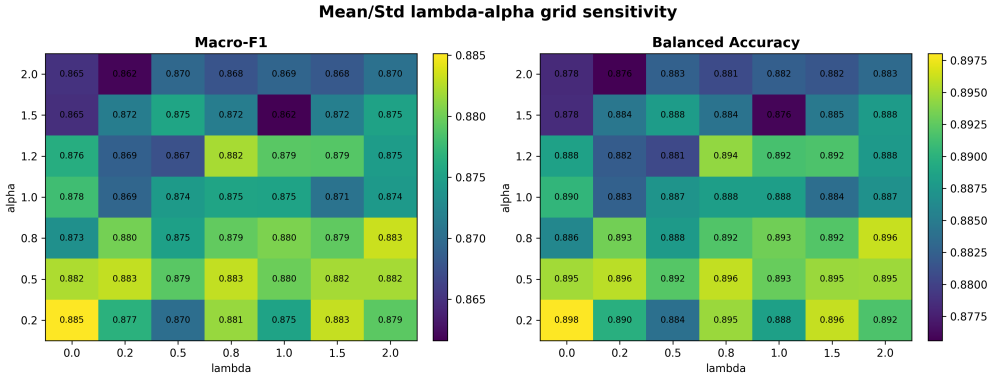


**Figure 7.** Two-dimensional performance landscape of Mean/Std over the $(\lambda, \alpha)$ parameter space. The heatmap reports the mean Macro-F1 obtained across the explored hyperparameter grid and highlights a broad region of stable high-performance configurations.

From a practical perspective, the existence of a broad high-performance plateau suggests that lightweight distribution-aware aggregation can be deployed without extensive hyperparameter calibration, improving both the reproducibility and the practical usability of the proposed approach.

### *5.5. Interpretation of Aggregation Behavior*

The experimental results indicate that aggregation quality in federated learning depends not only on the amount of local data available at each client but also on the diversity and representativeness of the underlying label distribution. In realistic IoT environments, large clients are not necessarily the most informative contributors to the global objective. Clients dominated by benign traffic or a limited subset of attack behaviours may disproportionately reinforce majority-class decision regions, while those carrying rarer but operationally important patterns may contribute only marginally under conventional size-based aggregation.

Mean/Std addresses this limitation by complementing client size with lightweight statistics of local label distributions. Rather than replacing the role of data availability, the proposed weighting mechanism balances sample volume and distributional diversity, reducing the tendency of highly skewed clients to dominate the global update solely because they contain more samples. The observed improvements in Macro-F1, Balanced Accuracy, and per-class minority attack detection suggest that this additional distributional information helps preserve complementary attack knowledge across the federation.

This behaviour also distinguishes Mean/Std from representative optimization-oriented FL approaches. Methods such as FedProx, FedAdam, and FedNova primarily improve convergence by mitigating client drift, introducing adaptive server-side optimization, or correcting objective inconsistency during training [15,17,18]. In contrast, Mean/Std leaves the optimization process unchanged and directly addresses the aggregation bias arising from the interaction between non-IID client distributions and severe multi-class imbalance. The empirical results suggest that in IoT botnet detection, correcting aggregation bias can be at least as important as improving optimization dynamics.

The hyperparameter analysis further supports this interpretation. The broad high-performance region observed across the explored $(\lambda, \alpha)$ space indicates that the proposed weighting rule does not depend on finely tuned parameter values. Instead, the two hyperparameters provide complementary and interpretable controls that adapt the aggregation process to different levels of client heterogeneity while preserving stable performance.

Overall, these findings suggest that lightweight, distribution-aware aggregation constitutes an effective and practically deployable strategy for heterogeneous FL-based IoT

intrusion detection. By relying only on a few scalar statistics already available at the client side, Mean/Std improves the representation of minority attack behaviours without requiring auxiliary models, clustering procedures, contribution-estimation mechanisms, or additional communication rounds. This combination of simplicity, interpretability, and robustness makes the proposed approach particularly well suited to resource-constrained IoT deployments.

## 6. Discussion

Building upon the empirical findings presented in Section 5, this section discusses their broader methodological and practical implications for FL-based IoT botnet detection and outlines the main limitations and future research directions of the proposed approach.

### *6.1. Validation of the Central Hypothesis*

The results presented in Section 5 strongly support the central hypothesis of this work: incorporating lightweight information about local label distributions into the aggregation process can improve federated learning performance under the combined challenges of non-IID client heterogeneity and severe multi-class imbalance. Across all evaluated scenarios, Mean/Std consistently achieves a more favourable balance between global classification performance and minority attack recognition than conventional size-based aggregation.

Several observations support this conclusion. First, Mean/Std systematically improves the principal imbalance-aware evaluation metrics while remaining competitive across complementary criteria. Because all compared methods share the same local architecture, preprocessing pipeline, training configuration, and privacy-oriented perturbation setting, these gains can reasonably be attributed to the aggregation mechanism itself rather than to differences in model capacity or optimization protocol.

Second, the per-class analysis shows that the benefits of the proposed approach are not confined to dominant traffic categories. The largest improvements are observed for attack classes that are particularly susceptible to non-IID fragmentation and aggregation bias, suggesting that the proposed weighting mechanism better preserves complementary attack knowledge distributed across heterogeneous clients. Consequently, the gains in Macro-F1 and Balanced Accuracy reflect a more balanced representation of the overall multi-class detection problem rather than an optimization biased toward majority classes [4,5].

Third, the multi-seed evaluation demonstrates that these improvements are reproducible rather than the result of favourable stochastic conditions. Mean/Std consistently maintains superior imbalance-aware performance while exhibiting low run-to-run variability, indicating that the proposed distribution-aware weighting mechanism remains stable under realistic FL optimization dynamics.

Taken together, these findings suggest that client dataset size alone is not a sufficient indicator of the informational value of a local update in heterogeneous IoT environments. Effective aggregation should account not only for data quantity but also for the characteristics of the underlying label distribution. By combining these complementary aspects through a lightweight and interpretable weighting mechanism, Mean/Std mitigates aggregation bias without increasing communication overhead or modifying local training procedures, thereby validating the core premise of the proposed approach.

### *6.2. Why Distribution-Aware Aggregation Matters*

The experimental findings suggest that one limitation of conventional federated learning under heterogeneous IoT conditions is the implicit assumption that client dataset size adequately reflects the informational value of a local update. While this assumption is often reasonable under approximately IID settings, it becomes less appropriate when severe non-

IID heterogeneity and multi-class imbalance coexist, as data quantity and label-distribution diversity may differ substantially across clients.

In practical IoT botnet detection scenarios, large clients may be dominated by benign traffic or a limited subset of attack behaviours, whereas smaller clients may contain rare but operationally important attack categories that are scarcely represented elsewhere in the federation. Under purely size-based aggregation, the former naturally exert greater influence, even though they may contribute relatively little information for learning robust decision boundaries across the complete label space. As a result, aggregation bias may arise even when the underlying optimization process remains well behaved.

Mean/Std addresses this limitation by complementing dataset size with lightweight statistics of local label distributions. Rather than assuming that all samples contribute equally to the global objective, the proposed weighting mechanism accounts for both data availability and distributional diversity, moderating the influence of highly skewed clients while preserving complementary attack knowledge across the federation. The improvements observed in Macro-F1, Balanced Accuracy, and per-class minority attack detection suggest that explicitly preserving distributional diversity during aggregation leads to a more balanced global model.

This perspective also clarifies the relationship between Mean/Std and existing FL approaches. Methods such as FedProx, FedAdam, and FedNova primarily target optimization-related challenges by mitigating client drift, stabilizing server-side optimization, or correcting objective inconsistency during training [15,17,18], whereas q-FedAvg seeks to improve fairness across client contributions. In contrast, Mean/Std directly addresses the aggregation bias induced by heterogeneous label distributions and severe class imbalance. From this viewpoint, the proposed approach should be regarded as complementary rather than competitive with optimization-oriented FL algorithms.

A further practical advantage of this design is its lightweight nature. Mean/Std requires neither auxiliary models, client clustering, knowledge distillation, contribution-estimation modules, nor additional communication rounds. The aggregation rule relies only on three scalar statistics already available at the client side: the local dataset size and two simple indicators derived from the label distribution. As summarized in Table 9, this design preserves the simplicity and communication efficiency of conventional FL while explicitly accounting for distributional heterogeneity.

**Table 9.** Conceptual comparison of representative FL aggregation strategies from a deployment perspective.

| Method | Optimization | Aggregation | Additional Overhead |
|---|---|---|---|
| FedAvg | None | Size-based | None |
| FedProx | Proximal regularizer | Size-based | None |
| FedAdam | Adaptive server update | Size-based | Server optimizer states |
| FedNova | Objective normalization | Size-based | Low |
| q-FedAvg | Fairness optimization | Client-fair weighting | None |
| Mean/Std | Standard FL | Distribution-aware weighting | Three scalars/client |

### 6.3. Practical Interpretation of the Hyperparameters

The ablation study presented in Section 5.4 provides additional insight into the practical role of the two control parameters that define Mean/Std. Rather than acting as arbitrary tuning coefficients, $\lambda$ and $\alpha$ regulate two complementary aspects of the distribution-aware weighting mechanism and admit an intuitive interpretation in heterogeneous FL settings.

The coefficient $\lambda$ controls the contribution of the dominance-gap term $\Delta(p_i)$, which captures the disparity between the most and least represented classes within a client's local

dataset. Increasing $\lambda$ therefore increases the sensitivity of the aggregation rule to extreme class concentration: small values produce behaviour closer to conventional size-based aggregation, whereas larger values progressively reduce the influence of clients dominated by only a few traffic categories.

The exponent $\alpha$ plays a complementary role by controlling the overall strength of the imbalance correction. Low values yield a mild adjustment and naturally recover FedAvg-like behaviour, whereas larger values amplify the influence of label-distribution information and increasingly favour clients exhibiting more balanced local datasets. Excessively large values, however, may over-penalize informative clients despite their substantial data availability, highlighting the need to balance data quantity and distributional quality.

An important observation emerging from the sensitivity analysis is that the proposed method does not rely on highly specific parameter choices. The one-dimensional performance curves and the two-dimensional $(\lambda, \alpha)$ heatmap reveal a broad region of consistently strong performance rather than a narrow isolated optimum. This behaviour indicates that Mean/Std remains robust to moderate hyperparameter variations and can maintain competitive imbalance-aware performance without exhaustive parameter optimization.

From a deployment perspective, this robustness is a practical advantage for real-world FL systems, where data distributions evolve over time and extensive hyperparameter tuning is often infeasible. The complementary and interpretable roles of $\lambda$ and $\alpha$, together with the existence of a broad high-performance region, suggest that the proposed aggregation strategy can adapt to different levels of client heterogeneity while preserving reliable performance and reproducibility.

### *6.4. Practical Deployment Implications*

Beyond its empirical performance, Mean/Std was designed with practical federated learning deployment constraints in mind. In large-scale IoT environments, aggregation mechanisms should improve model quality while preserving the low communication overhead and decentralized operation that make FL attractive for resource-constrained devices. Accordingly, the proposed approach retains the simplicity of conventional server-side aggregation while introducing only minimal additional information.

A key characteristic of Mean/Std is its lightweight implementation. The aggregation rule relies exclusively on three scalar quantities already available at the client side, namely the local dataset size and two simple statistics derived from the label distribution, $(n_i, \sigma(p_i), \Delta(p_i))$. No auxiliary neural networks, client clustering procedures, knowledge distillation mechanisms, contribution-estimation modules, or additional communication rounds are required. Local optimization remains unchanged, and only the server-side weighting mechanism is adapted to account for distributional heterogeneity.

This design distinguishes Mean/Std from representative optimization-oriented FL extensions. Methods such as FedProx, FedAdam, and FedNova introduce additional optimization-related components to improve convergence or mitigate client drift, whereas q-FedAvg incorporates fairness-aware weighting across clients. In contrast, Mean/Std preserves the standard optimization workflow and directly augments the aggregation stage with lightweight information about local label distributions, thereby addressing aggregation bias without increasing algorithmic complexity or computational requirements.

An additional advantage is its compatibility with privacy-oriented FL workflows. All experiments in this study are conducted under the update-perturbation setting described in Section 4.4, where additive Gaussian noise with $\sigma_n = 10^{-5}$ is applied to local model updates before aggregation. Although this mechanism does not provide formal cryptographic or differential privacy guarantees, it emulates the optimization uncertainty commonly associated with privacy-preserving aggregation pipelines. The consistent performance of

Mean/Std under this setting suggests that the proposed weighting mechanism remains effective even when local updates are subject to lightweight privacy-oriented perturbations.

Importantly, the additional statistics required by Mean/Std do not expose raw network traffic or individual samples and can be computed locally from information already available during training. As summarized in Table 10, the proposed approach preserves the simplicity and communication efficiency of conventional FL while introducing an explicit mechanism to account for distributional heterogeneity.

**Table 10.** Conceptual comparison of representative FL methods from a deployment perspective.

| Method | Optimization/Modification | Aggregation/Modification | Additional/Communication |
|---|---|---|---|
| FedAvg | No | No | None |
| FedProx | Yes | No | None |
| FedAdam | Yes | No | None |
| FedNova | Yes | No | Low |
| q-FedAvg | No | Partial (fairness-aware) | None |
| Mean/Std | No | Yes (distribution-aware) | Three scalars/client |

### *6.5. Limitations and Future Research Directions*

Although the experimental results demonstrate the effectiveness of Mean/Std under realistic heterogeneous FL conditions, several limitations should be considered when assessing the broader applicability of the proposed approach.

First, the empirical evaluation is limited to the N-BaIoT benchmark [11]. While this dataset provides a realistic IoT environment with heterogeneous devices, naturally imbalanced traffic distributions, and diverse botnet behaviours, no single benchmark can fully capture the variability of operational deployments. Extending the evaluation to additional intrusion-detection datasets and application domains would further strengthen the evidence for the generalizability of the proposed aggregation strategy.

Second, this work focuses on a supervised multi-class setting in which local label information is available for computing the distribution statistics used during aggregation. Extending the framework to weakly supervised, semi-supervised, or unsupervised FL scenarios represents a promising direction, particularly when local annotation resources are limited.

Another limitation concerns the adopted privacy model. The update-perturbation mechanism employed in our experiments serves as a privacy-oriented robustness evaluation inspired by secure aggregation workflows and does not provide formal differential privacy guarantees or implement a cryptographic secure aggregation protocol. Although all compared methods operate under the same perturbation setting, integrating Mean/Std with formal privacy-preserving mechanisms remains an important direction for future work.

Several extensions could further improve the applicability of the proposed approach. In particular, adaptive strategies for automatically adjusting the hyperparameters $\lambda$ and $\alpha$ according to the observed degree of client heterogeneity may further enhance robustness. Evaluating Mean/Std in larger and more dynamic federations, including cross-silo and hybrid FL deployments, would also provide additional insight into its scalability and behaviour under evolving operating conditions.

Despite these limitations, the present study demonstrates that incorporating lightweight statistics of local label distributions into the aggregation process provides a practical and effective mechanism for mitigating aggregation bias under simultaneous non-IID heterogeneity and severe multi-class imbalance. This principle offers a solid foundation for future research on communication-efficient, privacy-aware, and distribution-aware federated learning for IoT security applications.

## 7. Conclusions

This paper investigated federated learning for multi-class IoT botnet detection under two practical challenges that frequently coexist in real-world deployments: heterogeneous non-IID client distributions and severe class imbalance. Under these conditions, conventional size-based aggregation may overemphasize large but highly skewed clients, limiting the representation of minority yet operationally important attack behaviours in the global model.

To address this limitation, we proposed Mean/Std, a lightweight, distribution-aware aggregation strategy that combines a client-size proxy with simple statistics derived from local label distributions. By jointly accounting for data quantity and distributional quality, the proposed weighting mechanism mitigates imbalance-induced aggregation bias while preserving the simplicity and communication efficiency of standard Federated Learning.

Experimental results on the N-BaIoT benchmark [11], conducted under a privacy-oriented update-perturbation setting inspired by secure aggregation workflows [7,12], show that Mean/Std consistently outperforms representative optimization-oriented and fairness-aware FL baselines. The improvements observed in imbalance-aware metrics and minority attack detection, together with the multi-seed statistical analysis and hyperparameter sensitivity study, demonstrate that the proposed aggregation strategy provides stable and reproducible performance under realistic heterogeneous FL conditions.

Beyond the empirical results, this work highlights a broader insight: in highly heterogeneous IoT environments, client dataset size alone is not always a sufficient indicator of the informational value of a local update. Incorporating lightweight information about local label distributions during aggregation can improve the representation of complementary and underrepresented attack patterns without modifying local optimization or increasing communication complexity.

Future work will investigate the evaluation of Mean/Std on additional cybersecurity benchmarks and larger FL deployments, the development of adaptive strategies for automatically tuning the aggregation hyperparameters, and the integration of distribution-aware aggregation with formal privacy-preserving mechanisms such as secure aggregation and differential privacy.

Overall, the results suggest that lightweight, distribution-aware aggregation is a promising direction for improving the robustness and practical effectiveness of federated learning under simultaneous non-IID heterogeneity and severe class imbalance in IoT security applications.

**Author Contributions:** Conceptualization, Y.E.Y., Y.B. and N.E.K.; Methodology, Y.E.Y., Y.B. and N.E.K.; Software, Y.E.Y., Y.B. and N.E.K.; Validation, Y.E.Y., Y.B. and N.E.K.; Formal analysis, Y.E.Y., Y.B. and N.E.K.; Investigation, Y.E.Y., Y.B. and N.E.K.; Resources, Y.E.Y., Y.B. and N.E.K.; Data curation, Y.E.Y., Y.B. and N.E.K.; Writing—original draft, Y.E.Y., Y.B. and N.E.K.; Writing—review editing, Y.E.Y. and Y.B.; Visualization, Y.E.Y., Y.B. and N.E.K.; Supervision, Y.E.Y., Y.B. and N.E.K.; Project administration, Y.E.Y., Y.B. and N.E.K.; Funding acquisition, N.E.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding. The APC was funded by the authors.

**Data Availability Statement:** The data used in this study are publicly available as the N-BaIoT dataset described in [11].

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

## References

1. EL Yamani, Y.; Baddi, Y.; EL Kamoun, N. A survey on the contribution of ML and DL to the detection and prevention of botnet attacks. *J. Reliab. Intell. Environ.* **2024**, *10*, 431–448. [CrossRef]

2. El Mestari, S.Z.; Lenzini, G.; Demirci, H. Preserving data privacy in machine learning systems. *Comput. Secur.* **2024**, *137*, 103605.
3. Khan, L.U.; Saad, W.; Han, Z.; Hossain, E.; Hong, C.S. Federated learning for internet of things: Recent advances, taxonomy, and open challenges. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1759–1799. [CrossRef]
4. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1263–1284. [CrossRef]
5. Sokolova, M.; Lapalme, G. A Systematic Analysis of Performance Measures for Classification Tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [CrossRef]
6. Yu, B.; Kumbier, K. Artificial intelligence and statistics. *Front. Inf. Technol. Electron. Eng.* **2018**, *19*, 6–9. [CrossRef]
7. Kairouz, P.; McMahan, H.B. Advances and open problems in federated learning. *Found. Trends Mach. Learn.* **2021**, *14*, 1–210. [CrossRef]
8. Khraisat, A.; Alazab, A.; Singh, S.; Jan, T.; Gomez, A., Jr. Survey on federated learning for intrusion detection system: Concept, architectures, aggregation strategies, challenges, and future directions. *ACM Comput. Surv.* **2024**, *57*, 84. [CrossRef]
9. Zhang, H.; Ye, J.; Huang, W.; Liu, X.; Gu, J. Survey of federated learning in intrusion detection. *J. Parallel Distrib. Comput.* **2025**, *195*, 104976.
10. Zhao, Y.; Li, M.; Lai, L.; Suda, N.; Civin, D.; Chandra, V. Federated learning with non-iid data. *arXiv* **2018**, arXiv:1806.00582.
11. Meidan, Y.; Bohadana, M.; Mathov, Y.; Mirsky, Y.; Shabtai, A.; Breitenbacher, D.; Elovici, Y. N-baiot—network-based detection of iot botnet attacks using deep autoencoders. *IEEE Pervasive Comput.* **2018**, *17*, 12–22.
12. Bonawitz, K.; Ivanov, V.; Kreuter, B.; Marcedone, A.; McMahan, H.B.; Patel, S.; Ramage, D.; Segal, A.; Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*; ACM: New York, NY, USA, 2017; pp. 1175–1191.
13. Bunko, T.; Johnstone, M.N.; Yang, W.; Scott, B.A. A survey of privacy-preserving federated learning for intrusion detection systems. *Artif. Intell. Rev.* **2026**, *59*, 125. [CrossRef]
14. Bilal, M.A.; Ul Islam, I.; Idrees, S.; Qasim, M.; Khan, M.J.; Khan, J. Dataset-centric evaluation of federated intrusion detection models in IoT networks. *Sci. Rep.* **2026**, *16*, 2683.
15. 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.
16. Karimireddy, S.P.; Kale, S.; Mohri, M.; Reddi, S.; Stich, S.; Suresh, A.T. SCAFFOLD: Stochastic Controlled Averaging for Federated Learning. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*; PMLR: New York, NY, USA, 2020; Volume 119, pp. 5132–5143.
17. Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; Poor, H.V. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*; Curran Associates, Inc.: Red Hook, NY, USA, 2020.
18. Reddi, S.; Charles, Z.; Zaheer, M.; Garrett, Z.; Rush, K.; Konečnỳ, J.; Kumar, S.; McMahan, H.B. Adaptive federated optimization. *arXiv* **2020**, arXiv:2003.00295.
19. Huang, W.; Ye, M.; Shi, Z.; Wan, G.; Li, H.; Du, B.; Yang, Q. Federated learning for generalization, robustness, fairness: A survey and benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.* **2024**, *46*, 9387–9406. [CrossRef] [PubMed]
20. Gutierrez, D.M.J.; Solans, D.; Heikkilä, M.A.; Vitaletti, A.; Kourtellis, N.; Anagnostopoulos, A.; Chatzigiannakis, I. Non-IID data in Federated Learning: A Survey with Taxonomy, Metrics, Methods, Frameworks and Future Directions. *arXiv* **2024**, arXiv:2411.12377.
21. Alotaibi, B.; Khan, F.A.; Mahmood, S. Communication efficiency and Non-Independent and identically distributed data challenge in federated learning: A systematic mapping study. *Appl. Sci.* **2024**, *14*, 2720. [CrossRef]
22. Torre, D.; Chennamaneni, A.; Jo, J.; Vyas, G.; Sabrsula, B. Toward enhancing privacy preservation of a federated learning cnn intrusion detection system in iot: Method and empirical study. *ACM Trans. Softw. Eng. Methodol.* **2025**, *34*, 1–48. [CrossRef]
23. Dong, B.; Chen, D.; Wu, Y.; Tang, S.; Zhuang, Y. FADngs: Federated learning for anomaly detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *36*, 2578–2592. [CrossRef]
24. Fan, J.; Wu, K.; Tang, G.; Zhou, Y.; Huang, S. Taking advantage of the mistakes: Rethinking clustered federated learning for iot anomaly detection. *IEEE Trans. Parallel Distrib. Syst.* **2024**, *35*, 862–876. [CrossRef]
25. Sun, S.; Sharma, P.; Nwodo, K.; Stavrou, A.; Wang, H. FedMADE: Robust federated learning for intrusion detection in IoT networks using a dynamic aggregation method. In *Proceedings of the International Conference on Information Security*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 286–306.
26. Wei, Z.; Wang, J.; Zhao, Z.; Shi, K. Toward data efficient anomaly detection in heterogeneous edge–cloud environments using clustered federated learning. *Future Gener. Comput. Syst.* **2025**, *164*, 107559.
27. Peng, H.; Wu, C.; Xiao, Y. Fd-ids: Federated learning with knowledge distillation for intrusion detection in non-iid iot environments. *Sensors* **2025**, *25*, 4309. [PubMed]
28. Li, K.; Li, Y.; Zhang, J.; Liu, X.; Ma, Z. Federated deep long-tailed learning: A survey. *Neurocomputing* **2024**, *595*, 127906. [CrossRef]

29. Mohri, M.; Sivek, G.; Suresh, A.T. Agnostic federated learning. In *Proceedings of the International Conference on Machine Learning*; PMLR: New York, NY, USA, 2019; pp. 4615–4625.
30. Li, T.; Sanjabi, M.; Smith, V. Fair Resource Allocation in Federated Learning. *arXiv* **2019**, arXiv:1905.10497.
31. Zhu, L.; Liu, Z.; Han, S. Deep leakage from gradients. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
32. Swetha, C.; Shelke, C.J. Handling Class Imbalance in Federated Learning for Cyber-Physical Attack Detection. *Eng. Technol. Appl. Sci. Res.* **2026**, *16*, 31221–31228. [CrossRef]