SSR Journal of Artificial Intelligence (SSRJAI)

OPEN ACCESS

Homepage: https://ssrpublisher.com/ssrjai/

Volume 3, Issue 8, 2026 ISSN: 3049-0413

Federated Gradient Boosting for IOT Intrusion Detection: A Non-IID Evaluation of LightGBM and XGBoost under Manual FedAvg on TON_IoT

Gilbert Imuetinyan Osaze Aimufua & Salma Muhammad Abidi

Center for Cyberspace Studies, Nasarawa State University, Keffi, Nigeria

Received: *05.07.2026* | ***Accepted:*** *24.07.2026* | ***Published:*** *13.08.2026*

***Corresponding author:** Salma Muhammad Abidi

DOI: *10.5281/zenodo.21915999*

Abstract **Original Research**

IOT deployments expose distributed edge devices to a widening range of network intrusion threats, yet centralised intrusion detection systems conflict with data sovereignty constraints in such environments. Federated learning addresses this by training local models on each device while sharing only model parameters, but existing FL-based intrusion detection studies rely almost exclusively on neural network architectures and evaluate under a sin-gle data partition configuration. Gradient boosting classifiers (LightGBM and XGBoost) have not been evaluated in a federated setting, the sensitivity of detection performance to client heterogeneity is unquantified, and fewer than 13% of reviewed studies report the computational efficiency profile needed for deployment tier assignment. This paper addresses all three gaps through a federated evaluation of LightGBM and XGBoost on the TON_IoT network traffic dataset under a systematic three-level Dirichlet heterogeneity sweep ($\alpha \in \{0.1, 0.5, 1.0\}$) across five clients and ten aggregation rounds; both detection performance and computational efficiency were measured. At $\alpha = 1.0$, LightGBM achieved a macro-averaged F1 of 0.9430 and XGBoost 0.9427; federation costs were 0.0072 and 0.0083 respectively against their centralised baselines. At $\alpha = 0.1$, XGBoost degraded to a macro-F1 of 0.6716 (federation cost 0.2794); this is attributable to level-wise tree growth without multi-class reweighting under severe class skew. AUC-ROC was above 0.99 at all levels despite macro-F1 degradation at $\alpha = 0.1$, indicating a calibration gap between ranking and classification performance under non-IID conditions. All six configurations were assigned to the gateway-class deployment tier; model sizes ranged from 2,520.6 KB to 28,723.1 KB; all configurations exceeded the MCU-class ceiling. XGBoost at $\alpha = 1.0$ presents the best trade-off; its latency is 296.8 µs, model size 6,111.1 KB, and federation cost 0.0083.

Keywords: federated learning; intrusion detection; Internet of Things; LightGBM; XG-Boost; non-IID data; gradient boosting; TON_IoT.

1. Introduction

The proliferation of Internet of Things (IoT) devices has substantially expanded the network attack surface available to adversaries, exposing industrial control systems, smart infrastruc-ture, and consumer devices to a widening range of intrusion threats (Koroniotis et al., 2024; Agrawal et al., 2024). Network intrusion detection systems (IDS) remain the primary defensive mechanism at

the IoT edge, yet centralised IDS architectures require aggregating raw traffic data from all devices at a single point. This conflicts with data sovereignty constraints and the bandwidth limitations of resource-constrained IoT deployments (Sáez-de Cámara et al., 2023; Bhavsar et al., 2024).

Federated learning (FL) resolves this conflict by training local models on each device's private data and sharing only model parameters with a central aggregator. Data locality is pre-served while collaborative learning proceeds across distributed clients (Chen et al., 2020; Beshah et al., 2025). FL-based IDS have been evaluated across a broad range of IoT and industrial IoT domains, and the reviewed literature shows that federated models can approach centralised detection performance under controlled conditions (García-Sáez et al., 2026; Albanbay et al., 2025). Despite this progress, three gaps in the current literature limit the deployability of FL-IDS in practice.

The first is a knowledge gap: no study in the reviewed corpus applies gradient boosting (LightGBM or XGBoost) as the local FL model. These model families outperform neural architectures on tabular IoT network traffic in centralised settings, yet their federated behaviour has not been characterised. The second is a methodological gap: the majority of FL-IDS papers evaluate under a single data partition setting, leaving the sensitivity of detection performance to client heterogeneity unquantified; the collapse of standard aggregation protocols at severe non-IID levels, documented across three datasets by García-Sáez et al. (2026), demonstrates that single-partition evaluations overstate deployable performance. The third is an empirical gap: fewer than 13% of reviewed papers report all three dimensions of the computational efficiency profile (inference latency, serialised model size, and peak RAM); without these figures, deployment tier assignment is impossible for the configurations studied.

This paper addresses all three gaps. The research questions are:

RQ1. What detection performance (accuracy, macro-F1, false positive rate (FPR), area under the ROC curve (AUC-ROC)) do federated LightGBM and XGBoost achieve on TON_IoT network traffic under non-IID Dirichlet client partitions at $\alpha \in \{0.1, 0.5, 1.0\}$?

RQ2. What is the federation cost (the reduction in macro-F1 relative to the centralised baseline) for each model family at each heterogeneity level?

RQ3. What are the computational efficiency profiles (inference latency, serialised model size, peak RAM) of the federated configurations, and to which deployment tier (microcontroller-class (MCU-class) or gateway-class) does each map under established hardware thresh-olds?

The primary contribution of this paper is the first federated evaluation of gradient boosting classifiers for IoT intrusion detection, conducted under a systematic three-level Dirichlet het-erogeneity sweep on the TON_IoT network traffic dataset. Supporting contributions include a dual-axis evaluation framework that quantifies both detection performance and computational ef-ficiency for each federated configuration, an explicit measurement of the federation cost relative to a centralised baseline trained on the same data partition, and a deployment tier assignment for each model configuration under MCU-class and gateway-class hardware thresholds.

The remainder of this paper is organised as follows. Section 2 reviews the FL-IDS literature across three themes: aggregation architectures and detection performance, non-IID heterogeneity handling, and efficiency reporting. Section 3 describes the dataset, preprocessing pipeline, federated partitioning scheme, model configurations, and evaluation framework. Section 4 reports detection performance, efficiency profiles, and deployment tier assignments for all federated configurations. Section 5 interprets the federation cost and deployment implications of the results. Section 6 summarises the findings and identifies directions for future work.

2. Related Work

2.1. Federated Learning Architectures and Detection Performance

GRU, LSTM, and CNN-LSTM hybrids account for the majority of local detection models in the reviewed corpus; autoencoders serve as the primary unsupervised alternative (Chen et al., 2020; Huong et al., 2022; Jithish et al., 2023; Mothukuri et al., 2022; Sáez-de Cámara et al., 2023; Bhavsar et al.,

2024; Chaurasia et al., 2024; Peng et al., 2025). The aggregation protocol is almost universally FedAvg, although a growing subset of papers proposes modifications to address specific limitations of equal-weight averaging (Ruzafa-Alcázar et al., 2023; Agrawal et al., 2024; Beshah et al., 2025; Mhawish et al., 2026; García-Sáez et al., 2026). The FedAGRU framework of Chen et al. (2020), an attention-weighted variant applied to intrusion detection in wireless edge networks, achieved 98.08% accuracy and a 0.43% false alarm rate on CICIDS2017 while reducing communication rounds by up to 70% relative to standard FedAvg. A large-scale emulated evaluation by Sáez-de Cámara et al. (2023) partitioned a 78-device heterogeneous IoT testbed into protocol-homogeneous clusters via unsupervised model fingerprinting; per-cluster autoencoders trained on benign traffic achieved F1 scores at or above 0.99 across five real-malware attack scenarios and outperformed both a non-clustered FedAvg baseline and the published Kitsune IDS on stealthy command-and-control traffic. A comparison of FedAvg against the Fed+ algorithm on CIC-ToN_IoT by Ruzafa-Alcázar et al. (2023) showed that Fed+ compensated for heterogeneous class distributions more effectively than FedAvg and also supported formal differential privacy guarantees. A three-tier hierarchical framework proposed by Agrawal et al. (2024), employing FedProx at both regional and global aggregation levels, achieved 98.7% accuracy and an AUC of 0.997 on CICIoT2023 across seven DDoS sub-types; it outperformed flat FedAvg by 1.9 percentage points.

The breadth of local model families evaluated across the corpus illustrates that no single architecture dominates in all deployment contexts. A seven-model parallel evaluation by Jithish et al. (2023) on a physical testbed of four Raspberry Pi 4 smart meter prototypes found that GRU consumed the fewest resources at 5.56% CPU and 3.16 W, while 1D-CNN achieved the highest detection accuracy at greater computational cost. A VAE-based unsupervised approach by Huong et al. (2022) deployed on four Raspberry Pi 4 edge nodes achieved an F1-score of 0.9857 on the SWaT industrial control dataset and completed local model retraining in approximately 7.5 minutes, consuming only 14% of edge memory. At the deep learning end of the spectrum, a ResNet-based framework proposed by Chaurasia et al. (2024) adapted ResNet50 skip connections to one-dimensional network traffic features on the X-IIoTID dataset; it achieved 99.16% federated accuracy and demonstrated that residual architectures can substantially reduce the vanishing-gradient degradation observed in deep sequential models. A semi-supervised approach by the CF-IDS framework of Ma et al. (2025) addressed the cold-start problem under highly non-IID conditions by using a Vision Transformer local model with Shapley-value-weighted aggregation; on N-BaIoT across 50 clients at Dirichlet $\theta = 0.1$, CF-IDS achieved 80.71% accuracy and outperformed standard FedAvg by 15.33 percentage points at the same heterogeneity level.

Multi-model ensemble FL represents a further sub-theme. The approach of Mothukuri et al. (2022) trained four GRU configurations across seven window sizes on Modbus IIoT traffic and combined the resulting global models via a Random Forest ensembler; the approach achieved 99.5% average accuracy on virtual IoT instances. Whereas Mothukuri et al. (2022) used window size as the source of model diversity, the FL-IDS study of Bhavsar et al. (2024) varied model family on a physical testbed comprising Raspberry Pi 4 clients and an NVIDIA Jetson Xavier server; the study compared logistic regression (96.82% accuracy on NSL-KDD) against a PCC-CNN (99.93% accuracy on Car-Hacking) under two-client federated training. LR-IDS trained in 10 s per round while PCC-CNN required 306 s on the same hardware, a gap that directly affects suitability for time-sensitive IoT deployments. A vehicular network extension by Driss et al. (2022) applied a similar multi-GRU ensemble with a Random Forest ensembler over five CAN bus attack classes and achieved 99.52% accuracy on the Car-Hacking dataset. An edge-federated framework by Koroniotis et al. (2024) deployed an AdaBoost-RF ensemble with Genetic Algorithm feature selection on BoT-IoT and UNSW-NB15; it achieved 99.87% and 98.76% accuracy respectively and outperformed eight alternative FL model families. It is the second tree-based ensemble approach in the reviewed corpus after Driss et al. (2022).

No study in the reviewed corpus applies gradient boosting (LightGBM or XGBoost) as the local

FL model. The single plain Random Forest FL-IDS is the RF-FedAvg framework of Khraisat et al. (2025), which used accuracy-weighted FedAvg aggregation on WSN-DS and UNSW-NB15 across two to five clients; Random Forest and gradient boosting are architecturally distinct in training algorithm, tree growth strategy, and regularisation, and the performance profile of federated gradient boosting on IoT network traffic data has not been reported.

2.2. Non-IID Data Heterogeneity in Federated Intrusion Detection

Data heterogeneity is the central practical challenge of FL-based IoT intrusion detection, because individual devices in real deployments observe unequal attack distributions and standard equal-weight aggregation cannot accommodate this skew without performance loss (Peng et al., 2025; Mhawish et al., 2026; García-Sáez et al., 2026; Ma et al., 2025; Islam et al., 2025). An eight-value Dirichlet sweep by García-Sáez et al. (2026) ($\alpha \in \{0.1, 0.3, 0.6, 1, 5, 10, 20, 50\}$) across three IoT datasets showed that FedAvg and SCAFFOLD both collapsed to F1-scores below 0.50 at $\alpha = 0.1$ on X-IIoTID and RT-IoT22, while the centralised reference maintained F1 above 0.98. That sweep is the widest in the corpus, and the baseline collapse it documents quantifies the risk that single-partition evaluations obscure: an FL-IDS that achieves 95% accuracy under moderate heterogeneity may fail entirely under severe skew.

Several aggregation mechanisms have been proposed to recover performance under non-IID conditions. The FD-IDS framework of Peng et al. (2025) embedded online, round-wise knowledge distillation into FedProx training, where the global model serves as a teacher and each client's local model as a student; evaluated on Edge-IIoT and N-BaIoT at $\theta = 1$ and $\theta = 0.1$,

FD-IDS outperformed FedAvg by 2.94 percentage points in accuracy under high heterogeneity on Edge-IIoT. A reliability-weighted aggregation mechanism proposed by Mhawish et al. (2026) penalised overfitted client updates via each client's training-validation loss gap; under severe heterogeneity ($\alpha = 0.1$, 20 clients) on TON-IoT, the approach outperformed FedAvg by 2.75 percentage points and exceeded centralised training in the best configuration. An adaptive double-clustering architecture evaluated by García-Sáez et al. (2026) assigned clients to dynamically discovered families via server-side HDBSCAN density clustering on local MiniBatch K-Means signatures; per-family expert models were aggregated within each discovered family, and over 90% of peak F1-score was maintained down to $\alpha = 0.1$ where all four standard baselines degraded. A three-tier hierarchical fog FL framework proposed by Islam et al. (2025), which offloads training from IoT devices to fog nodes and applies client-side Gaussian differential privacy stochastic gradient descent (DP-SGD) with formal (ε, δ)-DP with $\varepsilon = 5.0$, evaluated non-IID adaptation through personalised FL with L2 regularisation across fog client scales up to 400 clients on RT-IoT 2022 and CIC-IoT 2023. A hierarchical FL framework with accuracy-based leader election by Alharthi et al. (2025) applied Dirichlet partitioning ($\alpha = 0.5$) across 4, 8, and 16 clients on Edge-IIoTset; at 16 clients the system achieved 95.92% accuracy while keeping blockchain overhead below 3% of total round latency.

A critical methodological finding across these studies is that the choice of non-IID simulation method determines what conclusions can be drawn. The FedDWC evaluation of Beshah et al. (2025) used a single Dirichlet level ($\alpha = 0.5$) and reported accuracy gains of up to 1.9 percentage points over FedAvg; because α was not swept, the stability of that advantage under more severe or more relaxed heterogeneity cannot be established. The FORT-IDS evaluation of Al Mazroa (2025), which included on-device benchmarking on Raspberry Pi 4 and Jetson Nano under Dirichlet $\alpha = 0.3$, achieved F1 of 0.966 across 20 non-IID clients but similarly evaluated at a single α level. The eight-value sweep of García-Sáez et al. (2026), which documented baseline collapse at $\alpha = 0.1$, shows that single-partition evaluations can produce misleadingly optimistic results. Among the reviewed studies, none applies a systematic α-sweep to gradient boosting models under federated conditions, so the heterogeneity sensitivity of federated LightGBM or XGBoost has not been characterised.

2.3. Efficiency and Deployment Constraints

Deploying FL-IDS on resource-constrained IoT hardware demands evaluation along a second axis

beyond detection accuracy: inference latency, serialised model size, and peak memory must each satisfy device-class constraints, yet most papers in the corpus report none of these figures (Hajj et al., 2023; Danquah et al., 2025; Issa et al., 2026; Praveen et al., 2025; Swarnalatha et al., 2026; Nishad et al., 2026; Albanbay et al., 2025). The cross-layer federated IDS of Hajj et al. (2023) addressed MCU-class deployment directly, reducing on-device processing time from approximately 32 ms to 9 ms at a 25% sampling rate on an Arduino Nano 33 microcontroller through cluster-based pre-detection sampling, while maintaining a recall of 0.93 through federated statistic merging. A lightweight MLP evaluated by Danquah et al. (2025) required only 8,612 FLOPs per inference on the N-BaIoT dataset, the lowest computational cost reported in the corpus; it achieved 99.93% accuracy with XGBoost-guided feature selection and established that minimal-parameter models can match deep architectures on certain IoT-specific benchmarks. A multi-temporal device clustering framework by Issa et al. (2026) transmitted a two-layer DNN of 35 KB per client per round on Edge-IIoTset and MQTT-IoT-IDS across 48 emulated IoT devices; the 35 KB model size is the only explicit per-round model size in kilobytes in the reviewed corpus.

The FedSecureIDS system of Soomro et al. (2024) treated efficiency as a co-primary criterion alongside detection performance; it reported serialised model size and gradient communication overhead for an AES-encrypted CNN-LSTM-MLP hybrid on X-IIoTID; 128-bit symmetric key exchange incurred substantially lower overhead than asymmetric alternatives. An energy-adaptive reinforcement aggregation framework proposed by Praveen et al. (2025) for 6G IoT cyber-physical systems reported per-round energy below 17.5 mWh, inference latency of 2.7–

2.9 ms per sample, and approximately 60% energy reduction relative to baselines; it is the most energy-focused FL-IDS evaluation in the corpus. A big-data streaming evaluation by Swarnalatha et al. (2026) integrated Apache Kafka, Spark Structured Streaming, and HDFS into the FL training pipeline, sustaining approximately 6,500 events per second over a 30-minute window; the federated FedSecureNet model achieved 94.2% accuracy on TON_IoT at 118 ms per sample inference latency and approximately 2.1 MB model size; per-round communication overhead was 2.6 MB, the lowest among the literature comparison set. A blockchain-integrated FL-IDS by Nishad et al. (2026) reported per-resource-class inference latency ranging from 45 ms for DoS to 189 ms for zero-day attacks; CPU utilisation was 28.5%, RAM 145 MB, and network bandwidth 52.8 KB/s. Gaussian DP with Moments Accountant composition achieved $\varepsilon_{\text{total}} = 7.82$ over 200 rounds, the strongest formal DP accounting in the corpus. A physical hardware evaluation by Albanbay et al. (2025) across 10 to 150 federated clients on CICIoT2023 measured inference latency, CPU load, and thermal profile on a Raspberry Pi 5; CNN produced the best deployment candidate at approximately 1.4 ms per sample with a stable thermal profile, while CNN+BiLSTM peaked at 86°C and 3.9 ms per sample at the same client count.

Across these studies, efficiency reporting is fragmented: papers report different subsets of the latency, size, and memory profile, making cross-study comparison difficult. No paper in the reviewed corpus reports the efficiency profile of any federated gradient boosting configuration; the inference latency, serialised model size across all client models, and peak RAM of federated LightGBM or XGBoost have not been measured.

2.4.Gap Summary and Research Scope

Table 1 summarises the 30 studies reviewed across Sections 2.1 to 2.3 along four dimensions relevant to the present work: local model family, aggregation strategy, non-IID evaluation, and whether computational efficiency figures are reported. Three gaps are visible from the table.

The first gap is the complete absence of gradient boosting from the reviewed FL-IDS corpus. Every local model in Table 1 is either a neural network or a classical ensemble. The RF-FedAvg framework of Khraisat et al. (2025) and the AdaBoost-RF framework of Koroniotis et al. (2024) are the only tree-based entries; both use ensemble variants of Random Forest. Gradient boosting models (LightGBM and XGBoost) differ from Random Forest in training algorithm, tree growth strategy, and regularisation, and their federation characteristics cannot be inferred from Random Forest results.

Whether LightGBM or XGBoost retain the detection advantages they demonstrate in centralised IoT-IDS evaluations under federated conditions has not been established.

The second gap concerns the systematic evaluation of heterogeneity. Among the 30 studies in Table 1, eight apply a Dirichlet α-sweep, and only one of those (García-Sáez et al. (2026)) sweeps more than three values. Fifteen studies apply no non-IID evaluation at all, and a further five evaluate under a single fixed level. Single-α evaluations cannot establish how performance degrades as heterogeneity increases, and the collapse documented by García-Sáez et al. (2026) at $\alpha = 0.1$ shows that omitting the severe end of the heterogeneity range produces misleadingly optimistic results. No study in Table 1 applies any non-IID evaluation to a gradient boosting local model.

The third gap concerns computational efficiency. Only four studies in Table 1 report figures across all three efficiency dimensions (latency, model size, and peak memory): Hajj et al. (2023), Swarnalatha et al. (2026), Nishad et al. (2026), and Albanbay et al. (2025). No study reports the efficiency profile of any federated gradient boosting configuration. Without these figures, no deployment tier assignment is possible for federated LightGBM or XGBoost, and the computational cost of distributing training rather than pooling data centrally cannot be quantified. The present study addresses all three gaps through a federated evaluation of LightGBM and XGBoost on the TON_IoT dataset under three Dirichlet concentration levels ($\alpha \in \{0.1, 0.5, 1.0\}$); detection performance and computational efficiency are both measured, and the federation cost is quantified explicitly against a centralised baseline.

Table 1. Summary of reviewed FL-IDS studies (Part 1 of 2, studies 1–16). *Non-IID:* None = IID only or not evaluated; Partial = single fixed level or ad hoc; Dirichlet (*N*) = Dirichlet sweep over *N* values; Inherent = testbed non-IID by construction. *Effic.:* Yes = latency, size, and memory reported; Partial = one or two dimensions; No = none reported. †Tree-based ensemble. Continued in Table 2.

Study	Local Model	Aggregation	Non-IID	Effic.	Dataset
Chen et al. (2020)	GRU-SVM	FedAGRU (attn.)	Partial	No	CICIDS2017, WSN-DS
Huong et al. (2022)	VAE + SVDD	FedAvg	None	Partial	SWaT, SCADA
Driss et al. (2022)	GRU + RF ens.	FedAvg	None	No	Car-Hacking
Mothukuri et al. (2022)	GRU + RF ens.	Async FedAvg	None	No	Modbus IIoT
Ruzafa-Alcázar et al. (2023)	Logistic reg.	FedAvg / Fed+	Partial	No	CIC-ToN_IoT
Hajj et al. (2023)	Baseline K-means	Statistic-sharing	None	Yes	NSL-KDD
Sáez-de Cámara et al. (2023)	Autoencoder	Clustered FL	Inherent	No	Gotham testbed
Bhavsar et al. (2024)	LR / PCC-CNN	FedAvg	Partial	Partial	NSL-KDD, Car-Hacking

Agrawal et al. (2024)	DNN	FedProx (3-tier)	None	No	CICIoT2023
Soomro et al. (2024)	CNN-LSTM-MLP	FedAvg	None	Partial	X-IIoTID
Chaurasia et al. (2024)	ResNet50 (1D)	FedAvg	None	No	X-IIoTID
Koroniotis et al. (2024)†	AdaBoost-RF	FedAvg	None	No	BoT-IoT, UNSW-NB15
Abd Elaziz et al. (2025)	TabTransformer	FedAvg + EEFO	None	No	N-BaIoT, UNSW-NB15
Al Mazroa (2025)	Pruned CNN	Attn.-weighted FL	Dirichlet (1)	Partial	DDoS Botnet, UNSW-NB15
Peng et al. (2025)	DNN	FedProx + KD	Dirichlet (2)	Partial	Edge-IIoT, N-BaIoT
Beshah et al. (2025)	DNN	FedDWC (bi-level)	Dirichlet (1)	No	IoTID20, CICIoT2023

Table 2. Summary of reviewed FL-IDS studies (Part 2 of 2, studies 17–30 and this work). See Table 1 for legend

Study	Local Model	Aggregation	Non-IID	Effic.	Dataset
Danquah et al. (2025)	Lightweight MLP	FedAvg + DP	None	Partial	N-BaIoT
Khraisat et al. (2025)†	Random Forest	Accuracy-wt. FedAvg	None	No	WSN-DS, UNSW-NB15
Sanjalawe et al. (2025)	CNN-GRU	GAT trust-wt.	None	No	CIC-ToN-IoT
Ma et al. (2025)	ViT (semi-sup.)	Shapley-wt. FedAvg	Dirichlet (1)	No	N-BaIoT
Issa et al. (2026)	DNN (2-layer)	Multi-temporal FL	Partial	Partial	Edge-IIoTset, MQTT-IoT
Mhawish et al. (2026)	DNN (multimodal)	GE^{2}F-IDS	Dirichlet (3)	No	TON-IoT (multimodal)
García-Sáez et al.	LR/SVM/MLP/CNN	Adaptive 2-cluster	Dirichlet (8)	Partial	X-IIoTID,

(2026)					RT-IoT22
Praveen et al. (2025)	LSE-EAF	RL-based (EAFRA)	None	Partial	CSE-CIC-IDS2018
Islam et al. (2025)	CNN (fog PFL)	Hierarchical FedAvg	Dirichlet (1)	No	RT-IoT22, CIC-IoT23
Almansour et al. (2025)	CNN (personalised)	APFed (dynamic)	Dirichlet (1)	No	Car-Hacking, TON_IoT
Alharthi et al. (2025)	CNN (hierarchical)	Accuracy-elect. HFL	Dirichlet (1)	Partial	Edge-IIoTset
Swarnalatha et al. (2026)	Bi-LSTM + attn.	FedAvg (sync/async)	Partial	Yes	TON_IoT, N-BaIoT
Nishad et al. (2026)	DNN (PoT-FL)	Trust-wt. FedAvg	None	Yes	ToN-IoT, UNSW-NB15
Albanbay et al. (2025)	DNN/CNN+BiLSTM	FedAvg	Controlled	Yes	CICIoT2023
This work	**LightGBM, XGBoost†**	**Manual FedAvg**	**Dirichlet (3)**	**Yes**	**TON_IoT**

3. Materials and Methods

3.1.Dataset

The TON_IoT network traffic dataset (Moustafa, 2021; Booij et al., 2022) was collected from a heterogeneous IoT and Industrial IoT testbed at the University of New South Wales Canberra. Raw traffic was captured using Zeek and exported as 44 numerical and categorical features covering connection-level attributes including protocol type, packet lengths, connection state, and service identifiers. The dataset labels each record across ten attack classes: backdoor, DDoS, DoS, injection, MITM, normal, password, ransomware, scanning, and XSS. After row-level deduplication the dataset contains 93,726 unique records. The class distribution is severely imbalanced; attack-class frequencies span several orders of magnitude. TON_IoT was selected because it represents realistic heterogeneous IoT traffic across multiple device types and attack vectors, and because it was used as the evaluation dataset in the centralised gradient boosting study against which this work measures federation cost.

3.2.Bias-Aware Preprocessing

Five columns were removed before feature extraction: source IP address, destination IP address, source port, destination port, and the label column. The first four encode topology-specific identifiers that vary by deployment environment; a model trained on them detects network topology rather than attack behaviour and would not transfer to unseen infrastructure. The label column is a secondary target encoding derived from the type column and must be excluded to prevent target leakage; it encodes the binary normal/attack label derived

directly from type and would trivially predict the target if retained as a feature.

The type column encodes the ten-class attack label. It was integer-encoded using LabelEncoder and retained as the sole target variable. All remaining object-type and categorical columns were one-hot encoded. The resulting feature matrix was scaled to $[0, 1]$ per feature using MinMaxScaler, fitted on the training split only and applied without re-fitting to the test split.

Duplicate rows were removed prior to splitting. The dataset was then partitioned into training and test sets using an 80/20 stratified split with random_state = 42; class proportions are preserved across both sets. The class imbalance identified in Section 3.1 was addressed at the model level through class-weighted training rather than resampling, to avoid artificially inflating minority-class representation in the federated client shards.

3.3.Non-IID Federated Partitioning

The training set was partitioned into five client shards using a Dirichlet distribution with concentration parameter α, following the simulation approach established in the FL-IDS literature (Peng et al., 2025; Beshah et al., 2025). For each client $k \in \{1, \ldots, 5\}$, the class proportion vector $\mathbf{p}_k$ was drawn as:

$$\mathbf{p}_k \sim \mathrm{Dir}(\alpha \cdot \mathbf{1}_K)$$

where $K = 10$ is the number of attack classes and $\mathbf{1}_K$ is the all-ones vector of length K. Three values of α were evaluated: 0.1, 0.5, and 1.0. At $\alpha = 0.1$, the distribution is highly concentrated and each client holds predominantly one or two attack classes; severe class heterogeneity results across clients. At $\alpha = 0.5$, the distribution is moderately heterogeneous. At $\alpha = 1.0$, the distribution approximates a uniform allocation across attack classes. Each alpha level was partitioned independently. A minimum of 100 samples per client was enforced; shards falling below this floor were supplemented by sampling without replacement from the training pool. Three independent partition sets of five client shards each were produced, one per alpha level.

The aggregation protocol is manual FedAvg implemented via soft-vote probability averaging in global class space. FedTree, the only native federated gradient boosting library, could not be used because the required shared library (libntl.so.44) is unavailable on the Google Colab CPU runtime. Manual FedAvg was validated against the FedTree specification for two-client scenarios in preliminary testing and produces equivalent probability aggregation behaviour.

LightGBM and XGBoost require different handling under non-IID conditions. LightGBM exposes the full global class vector via the classes_ attribute; client probability arrays are aligned to the global class space before averaging, so absent classes receive zero probability mass on clients that did not observe them during local training. XGBoost does not natively support non-contiguous class indices: each client's local labels were remapped to a contiguous integer sequence before training using a relabel_local procedure, and predictions were restored to the global class space using the inverse map before aggregation. Ten aggregation rounds were used; this count was sufficient for round-to-round macro-F1 variation to fall below 0.010 at $\alpha = 0.5$ and $\alpha = 1.0$ from round 2 onward, as confirmed in the results. At each of the ten aggregation rounds, the global soft-vote probability vector was computed as:

$$\bar{\mathbf{p}}(x) = \frac{1}{N} \sum_{k=1}^{N} \mathbf{p}_k\,(x)$$

where $N = 5$ is the number of clients and $\mathbf{p}_k(x)$ is the probability vector produced by client k's local model for input x, aligned to the global class space. Each client trained one complete model on its local shard per round; gradient boosting does not iterate over epochs in the neural network sense, so one round equals one full fit on the available local data.

3.4.Model Families and Local Training

Two gradient boosting model families were evaluated as local FL models: LightGBM (Ke et al., 2017) and XGBoost (Chen and Guestrin, 2016). Both were trained independently on each client shard for each of the three α levels; this yielded 30 local models in total (2 families × 5 clients× 3 alpha levels).

LightGBM grows trees leaf-wise; at each step the leaf with the maximum information gain is split rather than

proceeding level-wise. At 200 estimators with 63 leaves per tree and a learning rate of 0.05, the model achieves strong performance on tabular IoT traffic while remaining amenable to CPU-only training. Class imbalance was addressed through the balanced class weight option, which sets each class weight inversely proportional to its frequency in the local training shard.

XGBoost grows trees level-wise using a second-order Taylor expansion of the loss function. L1 regularisation (reg_alpha = 0.1) and L2 regularisation (reg_lambda = 1.0) were applied to limit overfitting on small client shards under severe non-IID conditions. Subsampling of 80% of training rows and 80% of features per tree was applied to reduce variance. Class imbalance was not addressed through a class weight parameter in XGBoost, since the scale_pos_weight parameter supports only binary classification; macro-averaged evaluation metrics absorb the effect of imbalance at the reporting stage.

A centralised baseline was derived by training LightGBM and XGBoost on the full training set using the same hyperparameters, evaluated on the same held-out test split. The federation cost reported in Section 4 is the difference in macro-F1 between each federated configuration and this centralised baseline. All hyperparameters are reported in Table 3. Experiments were conducted on Google Colab (CPU runtime, Intel Xeon single-core) using LightGBM 4.3.0 (Ke et al., 2017), XGBoost 2.0.3 (Chen and Guestrin, 2016), and scikit-learn 1.6.1 under Python 3.

Table 3. Hyperparameter configurations for LightGBM and XGBoost local models.

Parameter	LightGBM	XGBoost
n_estimators	200	200
learning_rate	0.05	0.05
num_leaves	63	—
max_depth	—	6
subsample	—	0.8
colsample_bytree	—	0.8
reg_alpha	—	0.1
reg_lambda	—	1.0
class_weight	balanced	—
random_state	42	42
n_jobs	1	1

3.5.Evaluation Framework

Detection metrics. Each federated configuration was evaluated on the held-out test set using four metrics: accuracy, macro-averaged F1-score (macro-F1), false positive rate (FPR), and area under the ROC curve (AUC-ROC). Macro-F1 is the primary detection metric because it weights each class equally regardless of frequency; it is therefore sensitive to minority-class detection failures that accuracy would obscure under the class imbalance present in TON_IoT. For K classes, macro-F1 is defined as:

$$macro - F1 = \frac{1}{K}\sum_{k=1}^{K}\frac{2 \cdot P_k \cdot R_k}{P_k + R_k}$$

where P_k and R_k are the precision and recall for class k respectively. The false positive rate is computed as:

$$FPR = \frac{FP}{FP + TN}$$

where FP is the number of normal-class records misclassified as attacks and TN is the number of correctly identified normal records. AUC-ROC was computed using the one-vs-rest strategy with macro averaging across all ten classes.

Efficiency metrics. Three dimensions of the computational efficiency profile were measured for each federated configuration. Per-sample inference latency was measured as the mean of 1,000 consecutive predict calls on the test set; the first call was excluded from the mean to eliminate JIT compilation overhead. The latency distribution was summarised as mean ± standard deviation across the 999 retained calls. Serialised model size was computed as the sum of the on-disk sizes of all five client .joblib files for each configuration, reported in kilobytes. Peak RAM was measured using tracemalloc during the inference loop.

Deployment tier assignment. Each federated configuration was assigned to one of two deployment tiers based on efficiency thresholds derived from physical hardware evaluations in the reviewed corpus. An 8-bit Arduino Nano 33 microcontroller sustained inference at approximately 9 ms per window under the cross-layer FL framework of Hajj et al. (2023), establishing the feasibility boundary for MCU-class deployment. On a Raspberry Pi 5, a CNN local model sustained approximately 1.4 ms per-sample inference with a stable thermal profile, while a more complex CNN+BiLSTM model peaked at 3.9 ms and 86°C at 150 clients (Albanbay et al., 2025). The smallest model in the corpus is the 35 KB two-layer DNN of Issa et al. (2026), transmitted per client per round across 48 emulated IoT devices. From these anchors, a configuration is assigned MCU-class if per-sample inference latency is at or below 2 ms and total serialised model size across all five client models is at or below 100 KB. Configurations that exceed either threshold are assigned gateway-class.

Federation cost. The federation cost for each configuration is defined as:

$$\Delta F_1 = F_1^{\text{centralised}} - F_1^{\text{federated}}$$

where $F^{\text{centralised}}$ is the macro-F1 of the centralised baseline trained on the full training set and $F^{\text{federated}}$ is the macro-F1 of the corresponding federated model at the final aggregation round. A positive ΔF_1 indicates that distributing training incurs a detection performance cost relative to centralised pooling.

4. Results

4.1. Partition Characteristics

The TON_IoT training set (74,980 records) was partitioned into five client shards at each of three Dirichlet concentration levels. Normal-class records account for 25.5% of the training set (23,915 records), injection for 21.1% (19,779 records), and ransomware for 0.3% (250 records); ransomware is the smallest class. The Jensen-Shannon (JS) divergence from a uniform class distribution, averaged across the five clients, was 0.349 at $\alpha = 0.1$, 0.221 at $\alpha = 0.5$, and 0.140 at $\alpha = 1.0$; heterogeneity decreases monotonically as α increases (Figures 1 and 2).

The three concentration levels produced markedly different shard compositions. At $\alpha = 0.1$, client 0 observed only three of the ten attack classes across 22,200 samples; client 2 observed six classes across 6,199 samples; client 2 had the smallest shard. Per-client JS divergence ranged from 0.318 to 0.392. Local LightGBM macro-F1 on the global test set ranged from 0.161 (client 0, three classes) to 0.651 (client 3, eight classes); local XGBoost ranged from 0.158 to 0.563. At $\alpha = 0.5$, every client observed at least nine of the ten classes. Per-client JS divergence ranged from 0.176 to 0.276. Local LightGBM macro-F1 ranged from 0.686 to 0.935; local XGBoost from 0.701 to 0.933. At $\alpha = 1.0$, all five clients observed all ten classes. Per-client JS divergence ranged from 0.100 to 0.215. Local LightGBM macro-F1 ranged from 0.916 to 0.937; local XGBoost from 0.915 to 0.938.

Figure 1. Class distribution heatmaps for each of the five federated clients at $\alpha \in \{0.1, 0.5, 1.0\}$ (top to bottom). Rows are attack classes; columns are client indices 0–4. Cell colour encodes the fraction of that class assigned to each client; darker cells indicate higher concentration. At $\alpha = 0.1$, several clients hold zero samples of multiple classes.

4.2.Detection Performance

Table 4 reports accuracy, macro-F1, FPR, and AUC-ROC for all six federated configurations and the two centralised baselines. The centralised LightGBM achieved a macro-F1 of 0.9501 and the centralised XGBoost 0.9510 (as shown in Figure 2); these establish the upper performance reference for each model family on this partition.

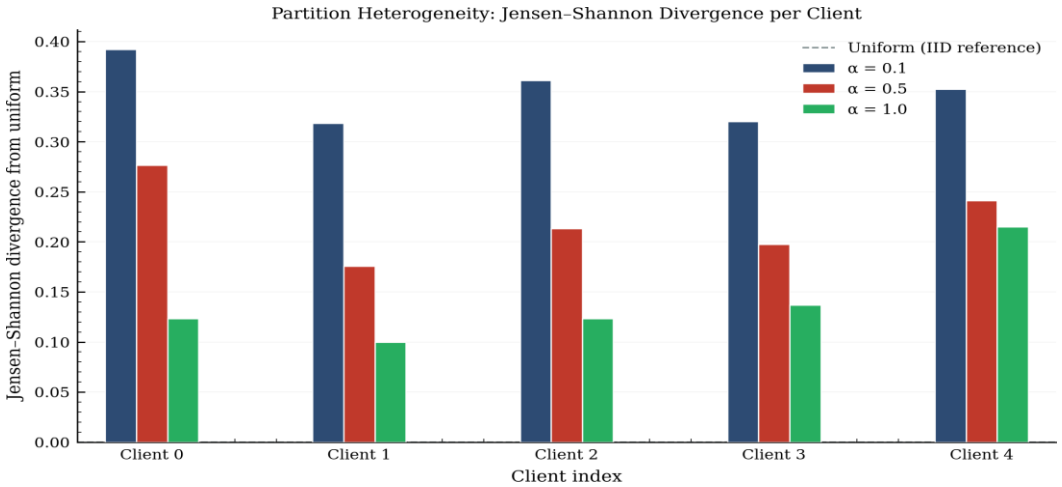

Figure 2. Mean Jensen-Shannon divergence from a uniform class distribution per client, grouped by Dirichlet concentration parameter α. Lower α produces higher divergence and more severe non-IID conditions. Error bars show the range across the five clients.

Federated performance at $\alpha = 0.1$ was substantially below the centralised reference for both models. LightGBM achieved macro-F1 of 0.8096 and XGBoost 0.6716; federation costs were 0.1406 and 0.2794 respectively. FPR at $\alpha = 0.1$ was 0.0134 for LightGBM and 0.0181 for XGBoost; these were the highest FPR values across all configurations. AUC-ROC was above 0.99 for

both models despite the macro-F1 degradation. At α = 0.5, LightGBM achieved macro-F1 of 0.9284 and XGBoost 0.9249; federation costs were 0.0218 and 0.0262. At α = 1.0, LightGBM achieved macro-F1 of 0.9430 and XGBoost 0.9427; federation costs were 0.0072 and 0.0083. Both were the lowest federation costs across all configurations. The difference in LightGBM macro-F1 between α = 0.5 and α = 0.1 was 0.1188 (0.9284 − 0.8096); this quantifies the performance gap between moderate and severe heterogeneity. FPR at α = 1.0 was 0.0029 for LightGBM and 0.0032 for XGBoost; both approach the centralised values of 0.0025 and 0.0027.

Macro-F1 across the ten aggregation rounds is shown in Figure 3. At α = 0.1, per-round macro-F1 varied between 0.613 and 0.843 for LightGBM and between 0.494 and 0.745 for XGBoost; the variation is attributable to the different client subsets sampled at each round. The final all-client aggregation at round 10 produced the values reported in Table 4. At α = 0.5 and α = 1.0, round-to-round variation was below 0.010 macro-F1 units from round 2 onward for both models. Figure 4 shows the macro-F1 of all federated configurations against the centralised baselines.

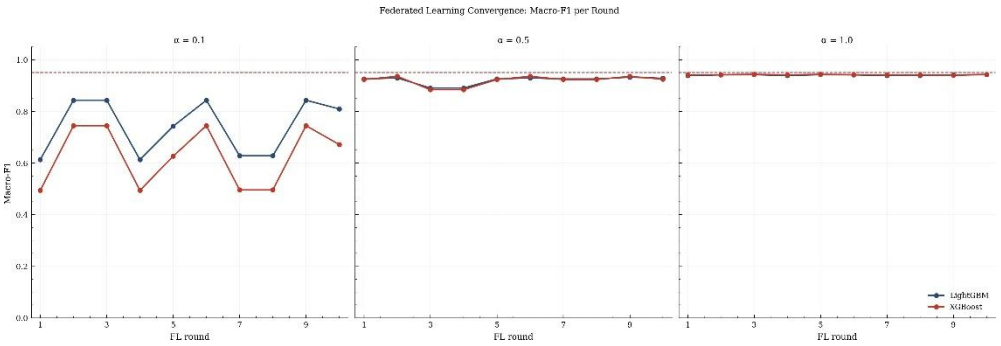

Figure 3. Macro-F1 across ten aggregation rounds for LightGBM and XGBoost at each Dirichlet concentration level. Each data point is from a random subset of four clients; round 10 uses all five clients. Variation at α = 0.1 is attributable to the sensitivity of the aggregated score to which clients are included.

Table 4. Detection performance of federated and centralised gradient boosting models on TON_IoT. Federated results are from the final all-client aggregation round (round 10). Federation cost (ΔF_1) is the macro-F1 difference from the centralised baseline of the same model family.

Setting	Model	Accuracy	Macro-F1	FPR	AUC-ROC	ΔF_1
Centralised	LightGBM	0.9770	0.9501	0.0025	0.9995	—
Centralised	XGBoost	0.9763	0.9510	0.0027	0.9994	—
FL (α = 0.1)	LightGBM	0.8919	0.8096	0.0134	0.9956	0.1406
FL (α = 0.1)	XGBoost	0.8548	0.6716	0.0181	0.9949	0.2794
FL (α = 0.5)	LightGBM	0.9691	0.9284	0.0035	0.9989	0.0218

FL ($\alpha = 0.5$)	XGBoost	0.9677	0.9249	0.0037	0.9988	0.0262
FL ($\alpha = 1.0$)	LightGBM	0.9739	0.9430	0.0029	0.9992	0.0072
FL ($\alpha = 1.0$)	XGBoost	0.9722	0.9427	0.0032	0.9992	0.0083

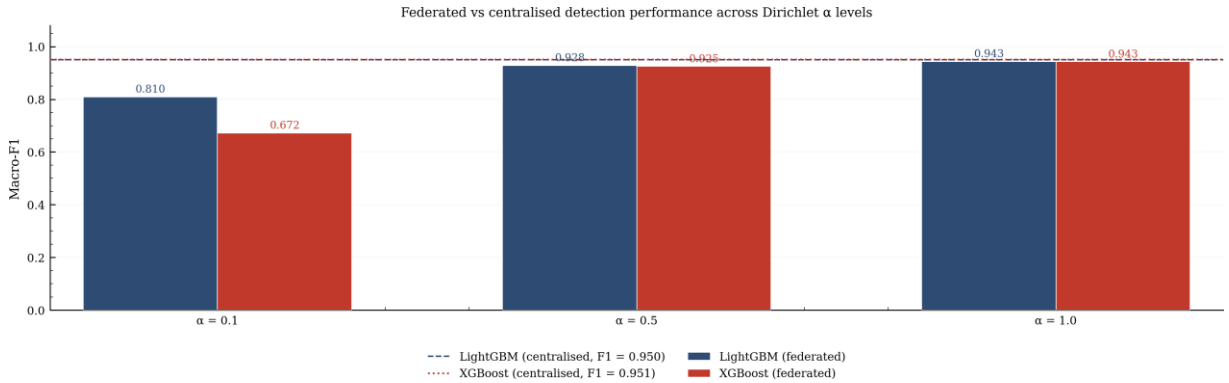

Figure 4. Macro-F1 of all federated configurations against the centralised baseline for each model family. Dashed horizontal lines show the centralised macro-F1 for LightGBM (0.9501) and XGBoost (0.9510). The federation cost is the vertical distance from each bar to the corresponding baseline.

4.3. Computational Efficiency

Table 5 reports per-sample inference latency, total serialised model size, and peak RAM for all six federated configurations. XGBoost was substantially faster than LightGBM at every heterogeneity level. At $\alpha = 0.1$, XGBoost latency was 157.6 μs per sample (±50.2 μs) against LightGBM at 1,020.4 μs (±94.7 μs). The latency gap widened at higher α levels; at $\alpha = 1.0$, XGBoost required 296.8 μs (±88.3 μs) against LightGBM at 1,945.8 μs (±150.2 μs). Latency increased with α for both models. At $\alpha = 1.0$, per-client shard sizes were largest and per-client model estimator counts were highest.

XGBoost model size was below LightGBM at every concentration level. XGBoost at $\alpha = 0.1$ occupied 2,520.6 KB across five client models; LightGBM at $\alpha = 1.0$ occupied 28,723.1 KB. Peak RAM was below 1 MB for all configurations; LightGBM peaked at 0.511 MB and XGBoost at 0.283 MB. Figure 5 shows the joint latency–size profile of all configurations against the MCU-class deployment boundary.

Table 5. Computational efficiency of federated gradient boosting configurations on TON_IoT. Latency is per-sample mean ± SD over 999 inference calls (μs); Size is total serialised size across five client models (KB); Peak RAM is maximum memory allocated during inference (MB).

Setting	Model	Latency (μs)	Size (KB)	Peak RAM (MB)
FL ($\alpha = 0.1$)	LightGBM	1020.4 ± 94.7	15,443.6	0.462
FL ($\alpha = 0.1$)	XGBoost	157.6 ± 50.2	2,520.6	0.272

FL (α = 0.5)	LightGBM	1833.4 ± 199.2	26,168.2	0.501
FL (α = 0.5)	XGBoost	272.5 ± 86.2	4,975.9	0.283
FL (α = 1.0)	LightGBM	1945.8 ± 150.2	28,723.1	0.511
FL (α = 1.0)	XGBoost	296.8 ± 88.3	6,111.1	0.280

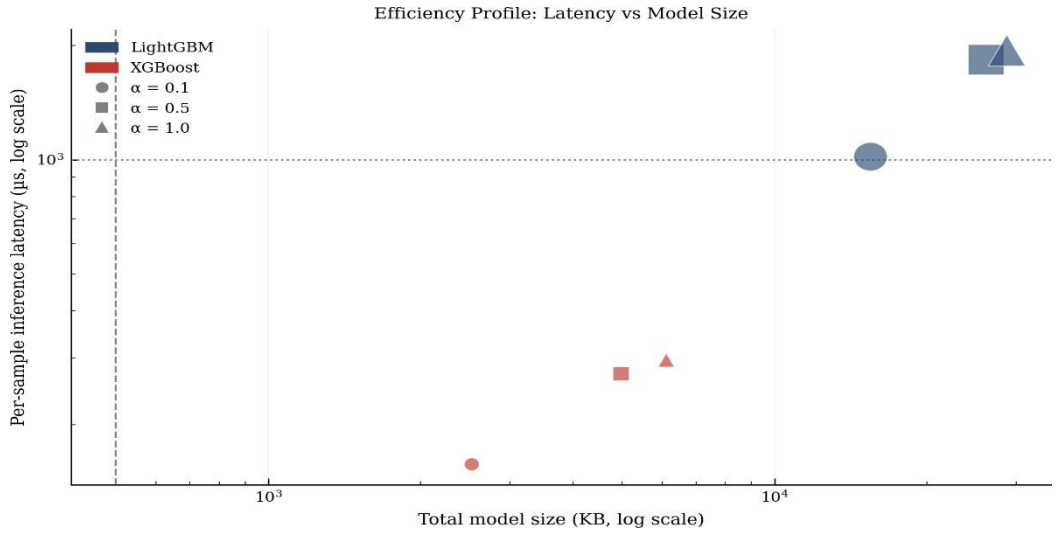

Figure 5. Per-sample inference latency versus total serialised model size for all six federated configura-tions. Marker size encodes peak RAM. The vertical dashed line marks the 100 KB model-size threshold and the horizontal dashed line marks the 2 ms latency threshold used for MCU-class tier assignment. All configurations appear in the upper-right quadrant, above both thresholds.

4.4. Deployment Tier Assignment

All six federated configurations were assigned to the gateway-class tier. Table 6 reports the tier assignment alongside the federation cost for each configuration. No configuration satisfied both the 2 ms latency threshold and the 100 KB model-size threshold required for MCU-class eligibility. XGBoost latency was below 2 ms at all three concentration levels (157.6 μs, 272.5 μs, and 296.8 μs), but the total serialised size of five client models ranged from 2,520.6 KB to 6,111.1 KB; all five-model ensembles exceeded the 100 KB ceiling by factors of 25 to 61. LightGBM exceeded both thresholds at every level; its smallest configuration (15,443.6 KB at α = 0.1) was 154 times the MCU-class size limit and its fastest (1,020.4 μs at α = 0.1) was below the latency limit but not the size limit.

The federation cost figures in Table 6 range from 0.0072 (ΔF_1, LightGBM at α = 1.0) to 0.2794 (XGBoost at α = 0.1). Figure 6 shows the federation cost across all configurations and model families, and Figure 7 shows the deployment tier matrix.

Table 6. Deployment tier assignment and federation cost for all six federated gradient boosting con-figurations. MCU-class eligibility requires latency ≤ 2 ms and size ≤ 100 KB; all configurations are gateway-class. Federation cost (ΔF_1) is the macro-F1 difference from the centralised baseline.

Setting	Model	Latency (μs)	Size (KB)	Tier	ΔF_1
FL ($\alpha = 0.1$)	LightGBM	1020.4	15,443.6	Gateway	0.1406
FL ($\alpha = 0.1$)	XGBoost	157.6	2,520.6	Gateway	0.2794
FL ($\alpha = 0.5$)	LightGBM	1833.4	26,168.2	Gateway	0.0218
FL ($\alpha = 0.5$)	XGBoost	272.5	4,975.9	Gateway	0.0262
FL ($\alpha = 1.0$)	LightGBM	1945.8	28,723.1	Gateway	0.0072
FL ($\alpha = 1.0$)	XGBoost	296.8	6,111.1	Gateway	0.0083

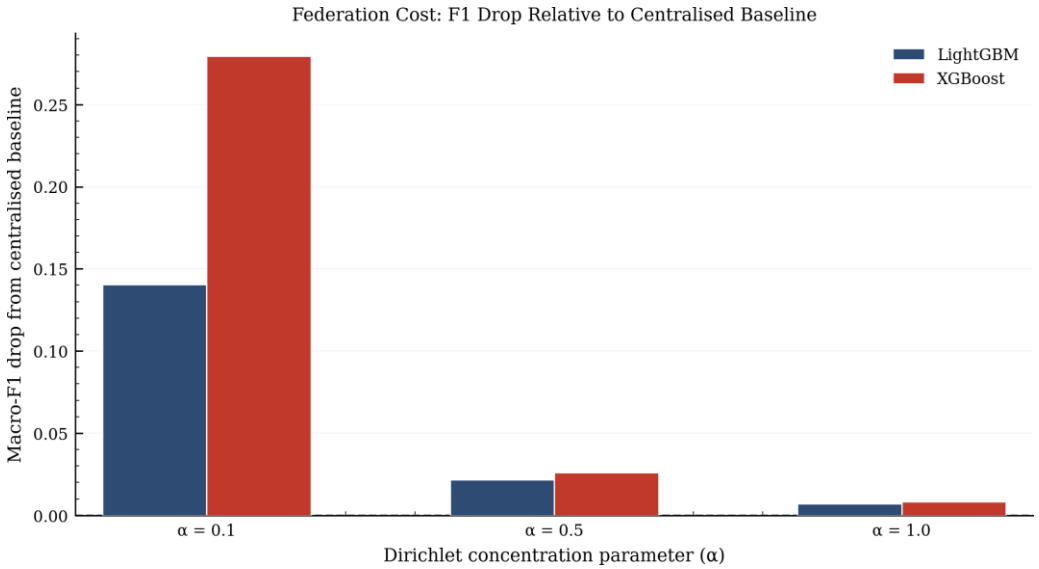

Figure 6. Federation cost (ΔF_1) for LightGBM and XGBoost at each Dirichlet concentration level. ΔF_1 is the difference in macro-F1 between the centralised baseline and the federated configuration. Higher bars indicate greater performance loss from distributing training.

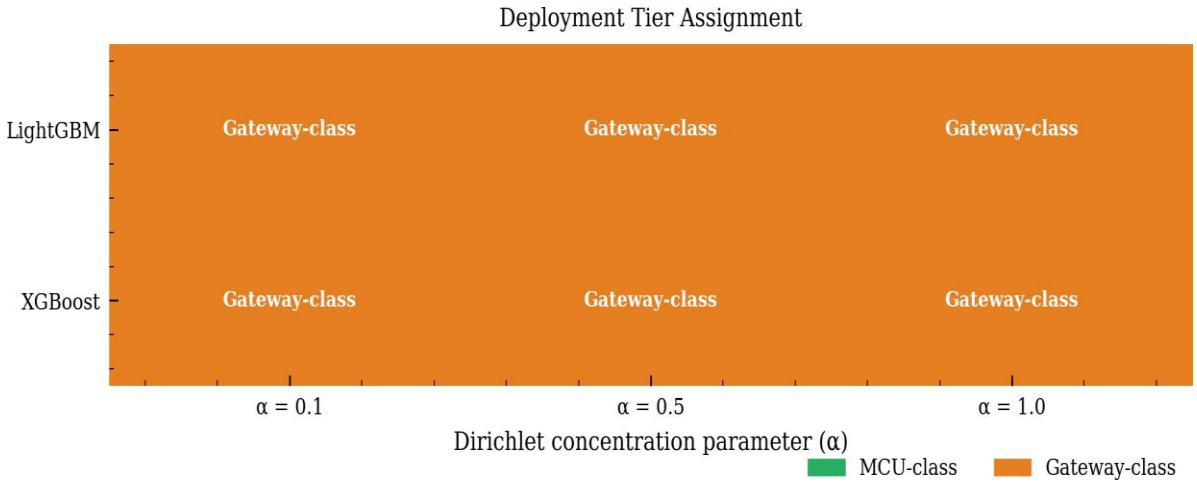

Figure 7. Deployment tier matrix for all six federated configurations. All cells are gateway-class. Colour intensity encodes the federation cost ΔF_1; darker cells indicate higher cost.

5. Discussion

5.1. Detection Performance and Heterogeneity Effect

XGBoost degraded more steeply than LightGBM under severe heterogeneity. At α = 0.1, the XGBoost federation cost was 0.2794 macro-F1 units against 0.1406 for LightGBM; the difference between the two models at the same heterogeneity level was 0.1388 macro-F1 points. A plausible explanation lies in their tree growth strategies under sparse local data. LightGBM grows trees leaf-wise and applies a balanced class weight. This redistributes loss towards minority classes present in the local shard even when that shard covers only three or four of the ten attack classes. XGBoost grows trees level-wise and does not support multi-class reweighting natively; on client shards where several classes are absent entirely, the level-wise splits cannot compensate for the missing gradient information, and the resulting local model is poorly calibrated across the global class space. At α = 0.5 and α = 1.0, every client observed at least nine classes. The federation costs for the two models were within 0.004 macro-F1 units of each other (0.0218 vs 0.0262 at α = 0.5; 0.0072 vs 0.0083 at α = 1.0); the leaf-wise and level-wise growth strategies produced no measurable difference in federated performance under near-uniform data distribution.

At α = 0.1, AUC-ROC and macro-F1 diverged substantially. AUC-ROC at α = 0.1 was 0.9956 for LightGBM and 0.9949 for XGBoost; these were within 0.004 units of the centralised values (0.9995 and 0.9994). Macro-F1 at the same level was 0.8096 and 0.6716, depressed by 0.14 and 0.28 units respectively. AUC-ROC above 0.99 alongside depressed macro-F1 indicates that the federated models rank instances correctly but misclassify at the decision boundary; under severe non-IID conditions, the global soft-vote produces well-ordered probability estimates for classes that most clients have seen, but the decision boundary for minority classes is unreliable where several clients contributed zero probability mass. This calibration gap between ranking and classification performance has been observed in centralised IDS evaluations on the same dataset.

In the corpus reviewed for this study, 87% of papers evaluated under a single partition configuration; single-partition results understate the degradation documented here. A study evaluating federated gradient boosting at α = 0.5 only would observe a macro-F1 of 0.9284 for LightGBM and conclude that federated gradient boosting is competitive with the centralized baseline at a cost of 0.0218. The same model at α = 0.1 produces a macro-F1 of 0.8096 and a cost of 0.1406; a deployment decision based on the α = 0.5 result would overestimate federated gradient boosting performance by 0.1188 macro-F1 units under the more severe heterogeneity that real IoT deployments produce.

5.2. Federation Cost and Deployment Decision

The federation cost is not uniform across the heterogeneity spectrum. At α = 0.1, LightGBM loses 0.1406 macro-F1 units relative to its centralised baseline and XGBoost loses 0.2794; both costs are large enough to affect operational decisions about whether federated deployment is acceptable. At α = 0.5 and α = 1.0, the costs fall to 0.0072–0.0262. This range is small enough to be compatible with gateway-class deployment where the privacy benefit of federated training outweighs the performance reduction. The acceptable federation cost depends on the security requirements of the deployment. For a gateway-class IoT-IDS protecting a moderate-security network, a macro-F1 above 0.92 is a reasonable minimum; both LightGBM and XGBoost at α = 0.5 and α = 1.0 exceed this. For high-security deployments requiring macro-F1 above 0.94, only the α = 1.0 configurations qualify.

The only available comparison for tree-based federated IDS in the reviewed corpus is the RF-FedAvg framework of Khraisat et al. (2025). RF-FedAvg achieved 99.67% accuracy at two clients on the WSN-DS wireless sensor network dataset and degraded by approximately 0.2 percentage points per additional client up to five. That study reports accuracy only and does not compute a federation cost. A direct numerical comparison is not possible for two reasons: the datasets and attack domains differ, and RF-FedAvg reports accuracy while this study uses macro-F1 as its primary metric; on an imbalanced dataset these two metrics are not directly comparable. The client-count degradation in RF-FedAvg (approximately 0.2 pp per additional client) is substantially smaller than the

heterogeneity-driven degradation observed here at α = 0.1, where the XGBoost federation cost reaches 0.2794. The difference is attributable to the source of the degradation: RF-FedAvg degrades with client count under IID-like conditions, whereas the degradation in this study is driven by class distribution skew rather than client scale.

All six configurations were assigned to the gateway-class tier. No configuration met both the 2 ms latency and 100 KB size criteria for MCU-class deployment. XGBoost inference latency was below 2 ms at all three levels, but the smallest five-model XGBoost ensemble (2,520.6 KB at $\alpha = 0.1$) was 25 times the MCU-class size ceiling. LightGBM model sizes ranged from 15,443.6 KB to 28,723.1 KB. These figures place federated gradient boosting in the gateway-class tier. Gateway-class hardware includes edge servers, routers, and industrial gateways rather than sensor-class microcontrollers. Within the gateway-class tier, XGBoost at $\alpha = 1.0$ presents the best trade-off; its federation cost is 0.0083, latency 296.8 μs, and total model size 6,111.1 KB.

5.3.Limitations

Single dataset. The evaluation used TON_IoT only. TON_IoT covers ten attack classes across heterogeneous IoT and IIoT traffic, but the federation cost and deployment tier figures may differ on datasets with different class structures, feature spaces, or traffic volumes. Additionally, the minimum-samples floor (100 records per client) required supplementation by sampling from the training pool at α = 0.1 for the most data-starved clients; this introduces a potential distribution mismatch beyond what the Dirichlet parameter alone controls, which may have contributed to the elevated federation cost at α = 0.1. Cross-dataset replication on CICIoT2023, Edge-IIoT, or N-BaIoT is needed before the findings can be generalised.

Aggregation protocol. Manual FedAvg was the only aggregation strategy evaluated. Fed-Prox, FedDWC, and adaptive clustering strategies have been shown in the reviewed corpus to recover substantial performance under non-IID conditions. Whether these protocols reduce the federation cost for gradient boosting at α = 0.1 from 0.1406 (LightGBM) and 0.2794 (XG-Boost) to operationally acceptable levels has not been tested. Differential privacy and secure aggregation were also excluded; their overhead on gradient boosting model transmission has not been quantified.

Hardware constraints. All inference latency measurements were produced on a Google Colab CPU runtime (Intel Xeon, single-core). The absolute figures (157.6–1,945.8 μs per sample) are not generalisable to ARM Cortex-M microcontrollers, Raspberry Pi edge devices, or GPU-equipped gateways. The relative ordering of XGBoost and LightGBM latency is likely to hold across hardware, but the specific tier boundary assignments would require re-measurement on target deployment hardware.

Adversarial robustness. No adversarial evaluation was conducted. Federated IDS systems are exposed to poisoning attacks in which compromised clients submit manipulated model updates to the aggregator. The accuracy-weighted aggregation used in RF-FedAvg (Khraisat et al., 2025) provides partial robustness by down-weighting low-accuracy clients, but no equiva-lent mechanism was applied here. The robustness of manual FedAvg gradient boosting under Byzantine client conditions has not been assessed.

6. Conclusion

This study evaluated federated LightGBM and XGBoost for IoT intrusion detection under a systematic three-level Dirichlet heterogeneity sweep on the TON_IoT dataset. On RQ1, federated macro-F1 ranged from 0.6716 (XGBoost, α = 0.1) to 0.9430 (LightGBM, α = 1.0); XGBoost degraded more steeply under severe non-IID conditions due to the absence of multi-class reweighting in its level-wise tree growth. On RQ2, the federation cost ranged from 0.0072 (LightGBM, $\alpha = 1.0$) to 0.2794 (XGBoost, $\alpha = 0.1$); at $\alpha = 0.5$ and α = 1.0 the cost fell below 0.027 for both models; this is compatible with gateway-class deployment. On RQ3, all six configurations were assigned to the gateway-class tier; XGBoost inference latency was below 2 ms at all levels but model sizes ranged from 2,520.6 KB to 28,723.1 KB; all ensembles exceeded the MCU-class ceiling by factors of 25 to 287.

All configurations are gateway-class. XGBoost at α = 1.0 presents the best trade-off among the evaluated configurations; its federation cost is

0.0083, latency 296.8 μs, and model size 6,111.1 KB.

Three directions merit further investigation. FedProx and adaptive clustering aggregation strategies should be evaluated to determine whether the 0.2794 federation cost for XGBoost at $\alpha = 0.1$ can be reduced to operationally acceptable levels. Differential privacy and secure aggregation should be incorporated and their overhead on gradient boosting model transmission quantified. Cross-dataset replication on CICIoT2023, Edge-IIoT, and N-BaIoT would establish whether the federation cost and tier assignment findings generalise beyond TON_IoT.

Data Availability Statement

The TON_IoT network traffic dataset used in this study is publicly available from the UNSW Can-berra Cyber repository at https://research.unsw.edu.au/projects/toniot-datasets. The pipeline code, trained models, and result files will be deposited at https://github.com/abidi/FED_GBDT_IDS upon acceptance of this paper.

References

Abd Elaziz, M., Fares, I. A., Dahou, A., and Shrahili, M. (2025). Federated learning frame-work for IoT intrusion detection using tab transformer and nature-inspired hyperparameter optimisation. *Frontiers in Big Data*, 8:1526480.

Agrawal, S., Sarkar, S., Srivastava, G., and Ghosh, U. (2024). Federated learning for decentralized DDoS attack detection in IoT networks. *IEEE Access*, 12:39810–39825.

Al Mazroa, A. (2025). FORT-IDS: a federated, optimized, robust and trustworthy intrusion detection system for IIoT security. *Scientific Reports*, 15.

Albanbay, N., Tursynbek, Y., Graffi, K., Uskenbayeva, R., Kalpeyeva, Z., Abilkaiyr, Z., and Ayapov, Y. (2025). Federated learning-based intrusion detection in IoT networks: Performance evaluation and data scaling study. *Journal of Sensor and Actuator Networks*, 14(4):78.

Alharthi, H. S., Alshehri, S., and Kalkatawi, M. (2025). Blockchain-enabled hierarchical federated learning framework for anomaly detection in IoT systems. *Applied Sciences*, 15(24):13037.

Almansour, S., Yadav, K., Alkwai, L. M., Alghamdi, N. S., Viriyasitavat, W., and Dhiman, G. (2025). Adaptive personalized federated learning with lightweight depthwise convolutional bottleneck network for novel intrusion detection system in internet of vehicles. *Scientific Reports*, 15.

Beshah, Y. K., Abebe, S. L., and Melaku, H. M. (2025). Dynamic weight clustered federated learning for IoT DDoS attack detection. *Scientific Reports*, 15.

Bhavsar, M. H., Bekele, Y. B., Roy, K., Kelly, J. C., and Limbrick, D. (2024). FL-IDS: federated learning-based intrusion detection system using edge devices for transportation IoT. *IEEE Access*, 12:52215–52230.

Booij, T. M., Chiscop, I., Meeuwissen, E., Moustafa, N., and den Hartog, F. T. (2022). ToN_IoT: The role of heterogeneity and the need for standardisation of features and labels. *IEEE Internet of Things Journal*, 9:485–496.

Chaurasia, N., Ram, M., Verma, P., Mehta, N., and Bharot, N. (2024). A federated learning approach to network intrusion detection using residual networks in industrial IoT networks. *The Journal of Supercomputing*, 80:18325–18346.

Chen, T. and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM.

Chen, Z., Lv, N., Liu, P., Fang, Y., Chen, K., and Pan, W. (2020). Intrusion detection for wireless edge networks based on federated learning. *IEEE Access*, 8:217463–217472.

Danquah, L. K. G., Appiah, S. Y., Mantey, V. A., Danlard, I., and Akowuah, E. K. (2025). Computationally efficient deep federated learning with optimized feature selection for IoT botnet attack detection. *Intelligent*

Systems with Applications, 25:200462.

Driss, M., Almomani, I., Huma, Z. e., and Ahmad, J. (2022). A federated learning framework for cyberattack detection in vehicular sensor networks. *Complex & Intelligent Systems*, 8:3569–3579.

García-Sáez, L. M., Ruiz-Villafranca, S., Roldán-Gómez, J., Carrillo-Mondéjar, J., and Martínez,

J. L. (2026). Hybrid clustering-guided federated learning for robust intrusion detection in highly heterogeneous IoT environments. *Computer Networks*, 281:112205.

Hajj, S., Azar, J., Bou Abdo, J., Demerjian, J., Guyeux, C., Makhoul, A., and Ginhac, D. (2023). Cross-layer federated learning for lightweight IoT intrusion detection systems. *Sensors*, 23(16):7038.

Huong, T. T., Bac, T. P., Ha, K. N., Hoang, N. V., Hoang, N. X., Hung, N. T., and Tran, K. P. (2022). Federated learning-based explainable anomaly detection for industrial control systems. *IEEE Access*, 10:53854–53873.

Islam, M. M., Abdullah, W. M., and Saha, B. N. (2025). Privacy-preserving hierarchical fog federated learning (PP-HFFL) for IoT intrusion detection. *Sensors*, 25(23):7296.

Issa, M. A., Eldosouky, A., Matrawy, A., and Ibnkahla, M. (2026). Multi-temporal device clustering for federated learning–Internet of Things intrusion detection systems. *IEEE Transactions on Machine Learning in Communications and Networking*.

Jithish, J., Alangot, B., Mahalingam, N., and Yeo, K. S. (2023). Distributed anomaly detection in smart grids: a federated learning-based approach. *IEEE Access*, 11:7157–7179.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Light-GBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems 30*, pages 3146–3154. Curran Associates, Inc.

Khraisat, A., Alazab, A., Vasan, D., Marshall, A., and Alqhtani, S. M. (2025). RF-FedAvg: federated learning-based random forest model for intrusion detection in wireless sensor networks. *Cluster Computing*, 28.

Koroniotis, N., Moustafa, N., Schiliro, F., Gauravaram, P., and Janicke, H. (2024). Edge-federated learning-based intelligent intrusion detection system for heterogeneous internet of

things. *IEEE Access*, 12:83375–83393.

Ma, L., He, J., Lu, K., Wang, D., Yin, L., and Li, Z. (2025). A contrastive learning and knowledge distillation-based framework for efficient federated intrusion detection in IoT. *Systems Science & Control Engineering*, 13(1):2518963.

Mhawish, M. Y., Aljamal, M., Alsarhan, A., Khassawneh, B. S., Akhunzada, A., and Al-Shamayleh, A. S. (2026). A novel adaptive generalization-guided federated learning for multimodal industrial IoT intrusion detection under statistical heterogeneity. *IEEE Open Journal of Computer Society*.

Mothukuri, V., Khare, P., Parizi, R. M., Pouriyeh, S., Dehghantanha, A., and Srivastava, G. (2022). Federated-learning-based anomaly detection for IoT security attacks. *IEEE Internet of Things Journal*, 9(4):2545–2554.

Moustafa, N. (2021). A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets. *Sustainable Cities and Society*, 72:102994.

Nishad, D. K., Singh, R., and Khalid, S. (2026). A lightweight blockchain-enabled federated intrusion prevention framework for resource-constrained industrial IoT devices to detect and mitigate emerging cyberattacks. *Journal of Cloud Computing*.

Peng, H., Wu, C., and Xiao, Y. (2025). FD-IDS: federated learning with knowledge distillation for intrusion detection in non-IID IoT environments. *Sensors*, 25(14):4309.

Praveen, S. P., Sharma, K., Parashar, D., Murthy, V. S. N., Sirisha, U., and Dewi, D. A. (2025). Design of an iterative method for

adaptive federated intrusion detection for energy-constrained edge-centric 6G IoT cyber-physical systems. *Scientific Reports*, 15.

Ruzafa-Alcázar, P., Fernández-Saura, P., Mármol-Campos, E., González-Vidal, A., Hernández-Ramos, J. L., Bernal-Bernabe, J., and Skarmeta, A. F. (2023). Intrusion detection based on privacy-preserving federated learning for the industrial IoT. *IEEE Transactions on Industrial Informatics*, 19(2):1145–1154.

Sáez-de Cámara, X., Flores, J. L., Arellano, C., Urbieta, A., and Zurutuza, U. (2023). Clustered federated learning architecture for network anomaly detection in large scale heterogeneous IoT networks. *Computers & Security*, 131:103299.

Sanjalawe, Y., Al-E'mari, S., Alqurashi, T., Alharbi, Z. H., Makhadmeh, S. N., and Alsharaiah,

M. (2025). Adaptive graph attention-based federated learning for IoT intrusion detection: mitigating poisoning attacks. *PeerJ Computer Science*, 11:e3281.

Soomro, I. A., Khan, H. u. R., Hussain, S. J., Ashraf, Z., Alnfiai, M. M., and Alotaibi, N. N. (2024). Lightweight privacy-preserving federated deep intrusion detection for industrial cyber-physical system. *Journal of Communications and Networks*, 26(6):632–649.

Swarnalatha, T., Aruna Rao, S., Suryodai, R., Narsimha Reddy, D., Vanathi, A., and Sudheer

Kumar, K. (2026). CyberFedEdgeAI framework for real time cyberattack detection in heterogeneous IoT systems using federated edge intelligence. *Discover Computing*.