



ARTICLE

CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer for High-Accuracy SDN-IoT Intrusion Detection

Surjeet Dalal^{1,*}, Shailee Choudhary², Rinku Dixit², Harun Elkiran³, Jyoti Kaushik⁴, Jawad Rasheed⁵ and Momina Shaheen⁶

¹Department of Computer Science & Engineering, Amity University, Haryana, India

²Department of Artificial Intelligence and Machine Learning, New Delhi Institute of Management, New Delhi, India

³Faculty of Engineering and Natural Sciences, Istanbul Medipol University, Istanbul, Türkiye

⁴Department of Computer Science & Engineering, Maharaja Agrasen Institute of Technology, Delhi, India

⁵Clean Energy Technologies Institute, Yıldız Technical University, Istanbul, Türkiye

⁶Department of Computing, School of Computing, Engineering, and the Built Environment (CEBE), University of Roehampton, London, UK

*Corresponding Author: Surjeet Dalal. Email: profsurjeetdalal@gmail.com or sdalal@ggn.amity.edu

Received: 08 June 2026; Accepted: 02 September 2026; Published: 28 September 2026

ABSTRACT: With the fast development of Software-Defined Networking-enabled Internet of Things (SDN-IoT) environments, the attack surface of modern network infrastructures has been enlarged to a great extent, making efficient and privacy-preserving intrusion detection a critical requirement in cybersecurity. Traditional centralized intrusion detection methods are usually affected by privacy issues, communication overhead, and performance degradation in heterogeneous data distribution. To tackle these problems, this paper presents a distributed SDN-IoT intrusion detection system based on an Adaptive Trust-Aware FedProx framework optimized by the Catch Fish Algorithm (CFA), which is implemented by a Feature-Selected CNN-Transformer architecture. The proposed system combines the following techniques into a single learning environment that respects privacy: (a) feature selection based on mutual information, (b) CNN-Transformer-based traffic classification, (c) adaptive trust-aware federated learning, (d) FedProx regularization, and (e) hyperparameter tuning using the Catch Fish Algorithm (CFA). Feature selection is used to determine the most informative traffic attributes, and the CNN-Transformer model is able to extract both local and long-range traffic connections. The adaptive trust mechanism includes two components: model consistency and detection performance, allowing trust-weighted aggregation and minimizing the impact of unreliable aggregation participants. Moreover, CFA is employed to fine-tune some critical hyperparameters to boost federated convergence and detection accuracy at the same time. The ASEADOS-SDN-IoT dataset consists of 457,044 network traffic records and is evaluated in experimental evaluation for various attack categories. The proposed framework achieved an accuracy of 86.67%, precision of 87.39%, recall of 86.67%, F1-score of 86.43%, MCC of 0.7763, ROC-AUC of 0.9756, and PR-AUC of 0.9121. After 20 communication rounds, the federated learning process decreased the average client loss from 0.9567 to 0.7023 and preserved trust scores ranging from 0.9029 to 1.0000 on the participating clients. Moreover, the system reached 37,005 samples per second throughput and 0.027 ms per sample response time, showing its potential for real-time use. The results validate that the proposed framework can effectively and efficiently detect intrusions in distributed SDN-IoT networks while preserving privacy.

KEYWORDS: SDN-IoT intrusion detection; federated learning; FedProx optimization; CNN-transformer; trust-aware aggregation; catch fish algorithm (CFA)

1 Introduction

The Internet of Things (IoT) has revolutionized many application areas such as smart cities, industrial automation, health care systems, transportation networks, and intelligent energy use. These devices are interdependent and constantly produce and share enormous amounts of data, facilitating real-time monitoring, automation, and decision-making. Despite these advantages, the growing size and diversity of IoT deployments have posed a number of challenges in the areas of network management, scalability, and cybersecurity. The lack of strong security features on resource-constrained IoT devices makes them prime targets for attacks like Distributed Denial of Service (DDoS), botnets, spoofing, probing, and unauthorized access. Thus, the challenges of providing secure communication and ensuring network integrity are crucial issues in today's IoT infrastructures [1].

To tackle the challenges of managing large-scale IoT ecosystems, Software-Defined Networking (SDN) has been proposed as a new paradigm. The SDN approach separates the control plane from the data plane and delivers network intelligence, dynamic traffic management, and increased programmability. The central SDN controller has a global network view and allows flexible policy enforcement, efficient use of resources, and adaptive traffic control. But SDN's centralized design also has new security issues. Communication between controllers and forwarding devices, flow rules, attacks on the controller plane, and advanced intrusion techniques can all be used to attack the network. As SDN-enabled IoT environments grow, detecting and mitigating malicious activities has become more critical than ever to ensure network reliability and availability [2,3].

Security methods based on pre-known attack signatures are often inadequate to deal with new and changing attack patterns. Anomaly-based intrusion detection systems (IDS) are used as another way to detect intrusion by detecting anomalies in normal network activities. Machine learning and deep learning are two recent technologies that have greatly enhanced the ability of IDSs to learn complex traffic and differentiate between malicious traffic and legitimate traffic. However, there are concerns about data collection from various devices or network domains in centralized training systems, as well as privacy, communication overhead, and scalability. Moreover, data generated by IoT devices are non-identical and heterogeneous by nature, and this can affect the performance of traditional centralized learning models [4,5].

Federated learning has received significant interest as a distributed learning approach that supports model training without sharing raw data while being private. A federated environment is where local clients train models on their own data and only send model parameters or updates to a central server. This can help maintain data security while enabling knowledge sharing among various network entities. But federated learning in SDN-IoT environments is challenged by several factors such as client heterogeneity, non-IID data distribution, communication inefficiency, model drift, and cheating or unreliable clients. Poisoned updates or local training process manipulation by adversarial clients could lead to a lower quality of the global model and consequently lower intrusion detection performance [6–8].

In order to address these limitations, the current work concentrates on deep traffic representation using features in a privacy-preserving federated learning system. In SDN-IoT intrusion detection, packets in traffic include local patterns along with flow attributes with long-range dependencies. Local traffic patterns can be extracted from Convolutional Neural Networks (CNNs), and contextual relationships between selected traffic features can be captured from Transformer encoders. It is thus reasonable to use the mutual-information-based feature selection technique with a CNN-Transformer model as a baseline to learn discriminative SDN-IoT traffic representations and to eliminate redundant input attributes.

To address the identified research gaps, this study develops a unified federated intrusion-detection framework for heterogeneous SDN-IoT environments. Unlike existing approaches, the framework jointly

considers feature relevance, client-level data heterogeneity, participant reliability, and optimization efficiency. The detailed architecture and operation of each component are presented in [Section 3.3.1](#).

1.1 Problem Formulation

As a result of this, Software-Defined Networking (SDN) Internet of Things (IoT) systems have a large number of heterogeneous devices generating network traffic, which could include both legitimate and malicious traffic. The goal of intrusion detection is to correctly identify flows of incoming traffic and to maintain confidentiality of data in multiple network domains. Then the SDN-IoT dataset can be represented as:

$$D = \{D_1, D_2, \dots, D_k\} \quad (1)$$

D_k is the local traffic dataset stored at the k -th client, and K is the number of all the SDN-IoT clients participating in the traffic. The local datasets are sets of network traffic samples D_i , along with their corresponding labels y_i :

$$F(w) = \sum_{k=1}^K (n_k/n) F_k(w) \quad (2)$$

Let w be the global model parameters, $F_k(w)$ be the loss function at client k , n_k be the number of samples available at client k , and $\sum_{k=1}^K n_k$ be the number of traffic instances across all the clients. The learned model then infers the likelihood of a traffic flow being part of an attack class. The final intrusion decision is given by:

$$\hat{y} = f(x; w^*) \quad (3)$$

$$\hat{y}(x) = \arg \max_{c \in \{\text{Benign}, \text{DoS}, \text{DDoS}, \text{Probe}, \text{Bot}\}} p_c(x) \quad (4)$$

where w^* is the optimized global model, $f(\cdot)$ is the classification function, $\hat{y}$ is the attack probability predicted by the model. The main goal is to achieve the highest intrusion detection performance with a strong classification capability in the presence of heterogeneous SDN-IoT traffic distributions. All the descriptions of the proposed methodology are presented in terms of the proposed methodology, using only the CFA-Optimized Adaptive Trust-Aware FedProx framework with Feature-Selected CNN-Transformer. The final proposed architecture does not have Graph Convolutional Autoencoder (GCAE) models or graph-based feature learning, unlike the earlier draft versions. In contrast, the traffic representation learning is carried out by the hybrid CNN-Transformer framework, as convolutional layers learn the local traffic characteristics and Transformer encoders learn long-range contextual dependencies between selected traffic features in SDN-IoT. While the respective studies study federated learning, federated security using blockchain, deep learning architectures, or federated trust management individually for SDN-IoT IDS, to the best of our knowledge, no previous works combine the four components of federated optimization based on adaptive trust-metrics, federated traffic representation learning using CNN-Transformer, federated feature selection using mutual-information and federated hyperparameter tuning using Catch Fish Optimization in a unified federated privacy-preserving IDS framework. The proposed approach is novel as it simultaneously tackles client data in distributed SDN-IoT networks that are heterogeneous and non-IID, unreliable client participation, automated hyperparameter optimization, and high-accuracy intrusion detection in the heterogeneous client model aggregation, client participation, hyperparameter optimization, and high-accuracy intrusion detection. This holistic integration offers a scalable and reliable cybersecurity solution, going beyond the

capabilities of traditional centralized or federated IDS solutions. The following research contributions are as follows:

1. Proposed a CFA-Optimized Adaptive Trust-Aware FedProx framework to improve federated intrusion detection performance under heterogeneous SDN-IoT environments.
2. Integrated mutual information-based feature selection with a CNN-Transformer architecture to enhance traffic representation learning and reduce redundant network features.
3. Developed an adaptive trust evaluation mechanism that dynamically assesses client reliability using model drift and local detection performance for secure federated aggregation.
4. Incorporated the Catch Fish Algorithm (CFA) for automatic tuning of learning rate, FedProx coefficient, trust threshold, and model hyperparameters to improve convergence stability.
5. Demonstrated the effectiveness of the proposed framework on the ASEADOS-SDN-IoT dataset, achieving 86.67% accuracy, 86.43% F1-score, and 0.9756 ROC-AUC while maintaining privacy-preserving distributed learning.

The remainder of this paper is organized as follows. [Section 2](#) reviews the existing literature on SDN-IoT intrusion detection and identifies key research gaps. [Section 3](#) presents the dataset, federated learning framework, and the proposed CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer methodology. [Section 4](#) describes the experimental setup and analyzes the obtained results using classification, federated learning, trust, and efficiency metrics. [Section 4.4](#) discusses the performance, strengths, and limitations of the proposed framework. Finally, [Section 5](#) concludes the paper and outlines future research directions.

2 Review of Literature

Abdulkareem et al. [1] carried out a detailed study of the literature on IDS for IoT and non-IoT networks from 2018 to 2024. The review was based on the analysis of datasets, machine learning classifiers, attack categories, experimental results, and the strengths and weaknesses of existing IDS approaches. The study reinforced the relevance of supervised learning methods and freely available datasets of IoT. Because it is a survey, there aren't any fresh detection results or frameworks.

In 2026, Alsubaei et al. [2] introduced the concept of a cyberdefense architecture that combines Blockchain, SDN, and Artificial Neural Networks (ANNs) for detecting DDoS attacks in IoT networks. It used distributed SDN controllers, achieved 98.4% accuracy and 88.7% recall in an SDN intrusion detection dataset, and a 93.4% F1-score. The study proved that the system was scalable, trustworthy, and able to resist DDoS attacks. But the incorporation of blockchain technology can bring its own set of computational and communication costs in resource-poor IoT scenarios.

Arthi et al. [3] presented a deep neural network (DNN)-based support vector machine (SVM) intrusion detection system (IDS) for securing the Internet of Medical Things (IoMT) environments. The framework used a custom dataset for healthcare SDN-IoT, which includes 44 traffic features and various attack scenarios such as DDoS attacks, phishing attacks, etc. The experimental results showed that the proposed approach had better attack detection and mitigation capabilities than the existing approaches without affecting service availability in healthcare applications. One limitation of this work is that it does not consider the evaluation of large-scale heterogeneous SDN-IoT deployments.

Chaganti et al. [4] proposed a Long Short-Term Memory (LSTM) based intrusion detection system for SDN-IoT networks and tested it on two SDN-IoT datasets. The proposed deep learning architecture was able to achieve a multiclass classification accuracy of 97.1%, which was used to identify and classify different types of attacks. This study showed that temporal traffic learning is effective for intrusion detection and gave

extensive visualization of the learned feature embeddings. But large-scale distributed IoT deployments might encounter scalability and privacy issues with the centralized learning architecture.

Dake et al. [5] suggested a multi-agent reinforcement learning framework based on Multi-Agent Deep Deterministic Policy Gradient (MADDPG) for SDN-IoT networks for the routing optimization and DDoS traffic prevention problems. The proposed framework was based on a Markov Decision Process (MDP) to emulate the network behavior and cooperative agents for load balancing and attack mitigation. Experimental results demonstrated that the delay, jitter, Packet Loss Rate (PLR), and ID performance were significantly improved compared to the conventional Deep Deterministic Policy Gradient (DDPG) approaches. The effectiveness of the convergence of this framework, however, relies on a large amount of training and computational resources.

To secure SDN-IoT networks, Ferrão et al. [6] suggested a Multi-Attack Intrusion Detection System (MAIDS) that combines XGBoost and Random Forest classifiers. The average accuracy of the two evaluations on the NSL-KDD and CICIDS2017 datasets is 99.88% and 99.89%, respectively, with a reduction in the false alarm rate by 33.23% when compared with existing methods. The framework demonstrated effective detection across multiple attack classes on the evaluated benchmark datasets. One limitation is the lack of adaptive trust and federated mechanisms in centralized machine learning models.

Guha Roy et al. [7] suggested a ZT-IDS that includes a Random Forest, XGBoost, LightGBM, and DBSCAN-based anomaly detection. By more than halving the features, the framework boosted the accuracy to 98.48% on the CICIDS, which includes more than 9 million flows, and kept the performance in real time on Raspberry Pi-class devices. The system is able to identify both known and zero-day attacks effectively. A limitation is that it validates only one large benchmark set.

Hafsa et al. [8] suggested an IDS/IPS system based on a cross-layer approach for detecting Routing Protocol for Low-Power and Lossy Networks (RPL) routing attacks in IoT networks. After selecting 12 critical features from an initial set of 29, the optimized CatBoost classifier achieved a 99% Detection Rate (DR), a 0.8% False Positive Rate (FPR), and a 98% Positive Predictive Value (PPV). This framework worked well to block Hello Flood, Version Number, and Worst Parent attacks. Its drawback is that the evaluation is limited to simulated routing attacks.

Hameed [9] proposed a novel Split Learning IDS where an 8-layer DNN is split across the IoT devices and SDN controllers to alleviate the computational load and guard against the leakage of sensitive data. The framework was evaluated using the InSDN dataset, a benchmark intrusion-detection dataset for Software-Defined Networking (SDN) environments, and obtained an accuracy of about 99.5% with a minimum loss of 0.02 and a lightweight model size of 106.67 KB. The model was able to detect DDoS and probing attacks effectively and still maintain privacy by transmitting smashed data. But the approach doesn't deal with trust management or malicious client participation in a distributed environment.

Hatamleh et al. [10] suggested a secure SDN-IoT framework that integrates image-based authentication, virtual blockchain (V-Block), graph neural networks, and deep reinforcement learning. The model uses attack graph construction based on Isomorphism-based Graph Neural Network (IGNN), controller placement and fox optimization, and intrusion detection based on Dueling Deep Q-Network (DDQN). Experimental assessment on NS-3 showed that the attack detection capability, anomaly identification, scalability, and link reliability were enhanced. The multi-stage architecture, however, has multiple layers of computational overhead.

Kiruthiga et al. [11] suggested a lightweight hybrid deep learning approach for SDN intrusion detection, named PS-SRHN, which integrates Siamese Networks, Recurrent Highway Networks, and Parametric Swish activation. The accuracy for the NSL-KDD dataset, SDN-IoT dataset, and InSDN dataset was 99.57%, 100%,

and 100%, respectively, with processing times of 17.45, 11.33, and 21.12 s, respectively. The model worked well to overcome issues of class imbalance and vanishing gradients and enhanced detection performance. One drawback is that there are no federated or distributed deployment considerations.

In Kumar and Arul Leena Rose's [12] approach, they suggested an IDS framework using machine learning to handle volumetric DDoS attacks in SDN-IoT networks by employing feature selection-based classification. The control plane of SDN was used to manage and defend the network against DDoS attacks, keeping its performance stable at different attack intensities. Experimental results showed that these methods performed better than the baseline in attack classification and mitigation. The framework does, however, not cover multi-attack intrusion scenarios.

Lakshan Yasarathna and Le-Khac [13] presented the ASEADOS-SDN-IoT dataset created using a hybrid SDN-IoT testbed with Open Network Operating System (ONOS) controllers, OpenFlow switches, physical IoT devices, and virtual nodes. There are 457,044 labeled flow records with 83 statistical features of DoS, DDoS, Probe, and Botnet attacks in the dataset. The effectiveness of separating benign traffic from malicious traffic was validated with an experimental setup that used both Machine Learning (ML) and Deep Learning (DL) models, proving the suitability of the dataset for the intrusion detection study in SDN-IoT. One drawback of the work is that it does not design a new intrusion detection framework but just puts forward the dataset generation and benchmark validation.

In this paper, Rullo et al. [14] introduced a Security-as-a-Service (SECaaS) framework that autonomously adapts its intrusion detection approach based on the characteristics of the IoT network. The IDS, as the network changes, constantly gathers context information, chooses its detection mechanisms, and self-configures. The framework is more flexible and can be used in heterogeneous environments (IoT) with minimal human intervention. One of the limitations is the need to acquire accurate context and system profiling.

Sharma and Bairwa [15] did a thorough survey of AI-based IDSs on IoT, with machine learning, deep learning, and anomaly detection techniques. The study covered the existing IoT architectures, IDS workflows, feature engineering techniques, and deployment challenges like device heterogeneity and resource constraints. It pinpointed the strengths and weaknesses of existing AI-powered IDS systems and proposed future research directions. It is a review article and does not provide any new experimental model.

Takhellambam et al. [16] created FL-MU, a federated learning benchmark dataset for intrusion detection in the context of IoT. It has more than 19 million instances, 121 features, 11 attack types, and 30+ IoT devices that generate traffic on seven different communication protocols. The suitability of the FL-MU dataset for federated intrusion-detection research was demonstrated by evaluating several state-of-the-art Federated Learning (FL)-based Intrusion Detection System (IDS) models on it. It is not a novel intrusion detection algorithm, but rather the focus is on the creation of datasets.

In SDN-IoT networks, Varalakshmi and Thenmozhi [17] suggested an adaptive control optimization for early detection and mitigation of DDoS attacks based on traffic analysis using entropy. The 99.4% detection accuracy, 18% energy saving, 70% faster detection time, 15% CPU saving, and 0.01% false positive rate were confirmed during experimental evaluation. The architecture enhanced the throughput by 2.6%, optimized resource allocation, and improved attack mitigation. However, the work is mainly concerned with DDoS attacks and does not consider other intrusion categories.

Wang et al. [18] put forward an IoT IDS combining the Causal Embedding Autoencoder (CEA) and Graph Attention Network (GAT) and jointly adversarial domain adaptation. The method brings knowledge from traditional network intrusion datasets to IoT environments and, at the same time, tackles the class imbalance issue by applying focal loss. The experimental results showed that the proposed method

outperformed the available domain adaptation methods for various evaluation measures. The limitation is that adversarial training and graph-based feature extraction make the model more complicated.

To enhance detection of rare attack classes, Wei et al. [19] proposed an Autoencoder-Based Hybrid Detection Model (AHDM). It is a dual-neuron-network framework, consisting of a trained neuron network on enhanced encoded features and another one trained on original features. The model outperformed the Deep Neural Network (DNN) and 0.14 for the Audio Cultural Intelligence Dataset (ACID) benchmark model, with a DDoS detection rate of 0.71 on the IOT-23 dataset against the DNN model 0.14 and ACID model 0.14, respectively. The architecture is, however, more complex due to the need for multiple classifiers for deployment.

To defend against attacks and prevent data infringements in resource-limited IoT contexts, Reis [20] introduced a lightweight intrusion detection framework that leverages property testing, Edge AI, and Federated Learning. The method covers 10% of the dataset and results in 92% accuracy in intrusion detection, which reduces processing time by 70% and inference time and energy consumption by more than 60%. The framework also increased the F1 score by as much as 11.3% compared to lightweight federated IDS baselines. An issue is the potential sensitivity to sampling quality in an environment with very dynamic attacks. A summary of existing works is presented in Table 1.

Research Gap Identified

The existing studies studying SDN-IoT intrusion detection are mainly for individual aspects, such as centralized deep learning, blockchain-based security, graph neural networks, federated learning, or optimization methods. No one, however, solves all these problems in an end-to-end privacy-preserving framework. But no one has solved all these problems in a single privacy-preserving framework. Furthermore, current federated IDS approaches usually assume a simple FedAvg or static aggregation approach, either disregarding or ignoring the reliability of clients when aggregating models. The proposed CFA-Optimized Adaptive Trust-Aware FedProx framework addresses these limitations by incorporating feature learning based on CNN-Transformer and optimization based on CFA to enhance the robustness and convergence stability of FedProx, while preventing attacks.

While the above studies have explored federated learning, domain adaptation, blockchain security, zero-trust architecture, and lightweight edge intelligence for SDN-IoT IDS, there is no existing work that combines adaptive trust-aware federated aggregation, FedProx optimization for non-IID heterogeneous environments, mutual-information-based feature selection, CNN-Transformer-based traffic representation learning, and CFA-driven hyperparameter optimization. The proposed framework overcomes these limitations by developing a unified framework for a privacy-preserving federated intrusion detection system with the aim of enhancing the robustness of the system, convergence stability, and detection performance in distributed SDN-IoT environments.

Table 1: Summary of existing works in SDN-IOT IDS.

Ref.	Author(s)	Year	Dataset	Proposed Method	Core Technique	Accuracy/Detection Rate	Key Contribution	Major Limitation
1	Abdulkareem et al. [1]	2024	Multiple IoT Datasets	Survey	ML-based NIDS Review	N/A	Dataset and classifier analysis	No implementation
2	Alsubaei et al. [2]	2026	SDN Dataset	Blockchain ANN IDS	Blockchain + ANN	98.4%	Secure controller communication	Blockchain overhead
3	Arthi et al. [3]	2024	Healthcare SDN-IoT	DNN-SVM Hybrid IDS	Deep Learning + SVM	98.7%	Healthcare IoT protection	Limited scalability
4	Chaganti et al. [4]	2023	SDN-IoT Datasets	LSTM IDS	Deep LSTM	97.1%	Temporal traffic learning	Centralized training
5	Dake et al. [5]	2021	SDN-IoT Simulation	MADDPG Framework	Multi-Agent Reinforcement Learning	96.4%	Routing and load balancing	Slow convergence
6	Ferrão et al. [6]	2023	CICIDS2017, NSL-KDD	MAIDS	XGBoost + Random Forest	99.88% and 99.89%, respectively	Multi-attack IDS	Centralized architecture
7	Guha Roy et al. [7]	2026	CICIDS (>9M flows)	Zero-Trust IDS	RF + XGBoost + LightGBM + DBSCAN	98.48%	Zero-day attack detection	Single benchmark evaluation
8	Hafsa et al. [8]	2025	RPL Routing Dataset	Cross-Layer IDS/IPS	CatBoost + Feature Selection	99.0% Detection Rate (DR), 0.8% False Positive Rate (FPR)	Routing attack detection	Limited attack diversity
9	Hameed [9]	2026	InSDN	Split Learning IDS	Split Learning + DNN	99.5%	Privacy-preserving IDS	No trust mechanism
10	Hatamleh et al. [10]	2025	SDN-IoT Simulation	PictureGuard	Blockchain + IGNN + DDQN	Improved ADR & Scalability	Multi-layer security framework	High complexity
11	Kiruthiga et al. [11]	2025	NSL-KDD, InSDN	PS-SRHN	Siamese RHN + Parametric Swish	99.57%–100%	Lightweight deep IDS	No FL support

(Continued)

Table 1 (continued)								
Ref.	Author(s)	Year	Dataset	Proposed Method	Core Technique	Accuracy/Detection Rate	Key Contribution	Major Limitation
12	Kumar & Arul Leena Rose [12]	2023	SDN-IoT Dataset	ML-based IDS	Feature Selection + ML	98.6%	DDoS mitigation	Single attack focus
13	Lakshan Yasarathna & Le-Khac [13]	2026	ASEADOS-SDN-IoT	Dataset Framework	SDN-IoT Traffic Generation	457,044 records, 83 features	Realistic SDN-IoT benchmark	Dataset only
14	Rullo et al. [14]	2024	Multiple IoT Scenarios	Kalis2.0	Context-Aware SECaaS IDS	Improved adaptive detection	Self-adaptive IDS	Context acquisition overhead
15	Sharma & Bairwa [15]	2025	Literature Review	Comprehensive Survey	ML, DL, Anomaly Detection	N/A	State-of-the-art IDS review	No experimental model
16	Takhellambam et al. [16]	2025	FL-MU	Federated IDS Benchmark	Federated Learning Dataset	19M records, 121 features	First large-scale FL benchmark for IoT IDS	No IDS framework proposed
17	Varalakshmi & Thenmozhi [17]	2025	SDN-IoT Dataset	Adaptive Control IDS	Entropy-based Detection	99.4%	Energy-aware DDoS mitigation	DDoS-focused
18	Wang et al. [18]	2026	IoT + Traditional IDS Datasets	AEC-GAT Domain Adaptive IDS	Autoencoder + GAT + Domain Adaptation	Superior to baselines	Cross-domain intrusion detection	High computational complexity
19	Wei et al. [19]	2024	IoT-23	AHDM	Autoencoder + Dual Neural Networks	DDoS DR = 71%	Small-sample attack detection	Multiple-model architecture
20	Reis [20]	2025	Real-world IoT Traffic	Property Testing IDS	Edge AI + FL	92.0%	70% lower processing time	Sampling sensitivity

3 Materials and Methods

3.1 Dataset

In this study, the experiments are carried out with a recently developed benchmark data set called the ASEADOS-SDN-IoT data set, which is a specially designed intrusion detection benchmark data set for Software-Defined Networking (SDN) based Internet of Things (IoT) networks. The testbed was a realistic hybrid testbed built with physical IoT devices, virtual hosts, OpenFlow-enabled switches, and an ONOS SDN controller. ASEADOS-SDN-IoT is an IoT intrusion dataset that includes both control plane and data plane communications to capture the dynamic behavior of SDN-based IoT networks, and is also realistic in today's network traffic. There are 457,044 network flow records measured during normal and malicious operation, and a variety of attack scenarios frequently seen in IoT environments.

The dataset contains 83 traffic features derived from network flows, covering packet-level, flow-level, temporal, and statistical features that capture communication behaviors in an SDN-IoT environment. These features include information such as packet counts, flow durations, byte statistics, protocol information, source-destination interactions, and traffic distribution characteristics. The attack traffic includes different attack classes such as Denial-of-Service (DoS), Distributed Denial-of-Service (DDoS), Probe, and Botnet to investigate the intrusion detection models under various attack scenarios. Both benign and malicious traffic instances can be provided, so that supervised, semi-supervised, and anomaly detection techniques can be developed for intrusion detection.

One of the advantages of the ASEADOS-SDN-IoT dataset is that it can be used in distributed and federated learning environments. The size of the records, the various traffic and attack classes, and attack distribution are similar to real-world SDN-IoT deployments, which makes this dataset highly suitable for the evaluation of privacy-preserving and trust-aware intrusion detection frameworks. Moreover, the dataset can be used to study issues of class imbalance, non-IID data distribution, malicious client behaviour, and model robustness in federated settings. The aforementioned properties make ASEADOS-SDN-IoT a useful roadmap for creating future Edge-Device Intrusion Detection Systems (ED-IDSs) for SDN-based IoTs with the aim of providing secure, scalable, and reliable cyber defense in SDN-IoT networks. The manuscript now makes the distinction between the full ASEADOS-SDN-IoT benchmark set and the processed subset of experiments used in implementing the research. The data statistics in the Abstract are for the full benchmark data set, while the class distribution is shown for the data set after pre-processing and sample preparation for the experiments reported here. This statement avoids any possible confusion about the size of the datasets and experimental setup.

Dataset shape: (457,044, 84)

Detected label column: Label

Label

Benign	260,797
Dos	127,772
DDoS	51,465
Bot	9090
Prob	7920

Name: count, dtype: int64

Classes: ['Benign' 'Bot' 'DDoS' 'Dos' 'Prob']

Feature count: 73

The total of 457,044 labeled network traffic records from the original ASEADOS-SDN-IoT benchmark were kept in the entire experimental workflow. Before any preprocessing parameters were estimated, the modeling dataset was divided by stratified random sampling into mutually exclusive training, validation, and test sets in a 70:10:20 ratio. Missing-value imputation, feature engineering, categorical encoding, and normalization parameters were fitted exclusively on the training set. The class distribution of the processed experimental subset, after these preprocessing steps, is shown in Fig. 1 for exploratory data analysis and visualization. The whole dataset was then randomly split into training, validation, and test sets, and then used to generate federated clients, train the CNN-Transformer model, optimize the model hyperparameters with the help of CFA, and evaluate the final model performance.

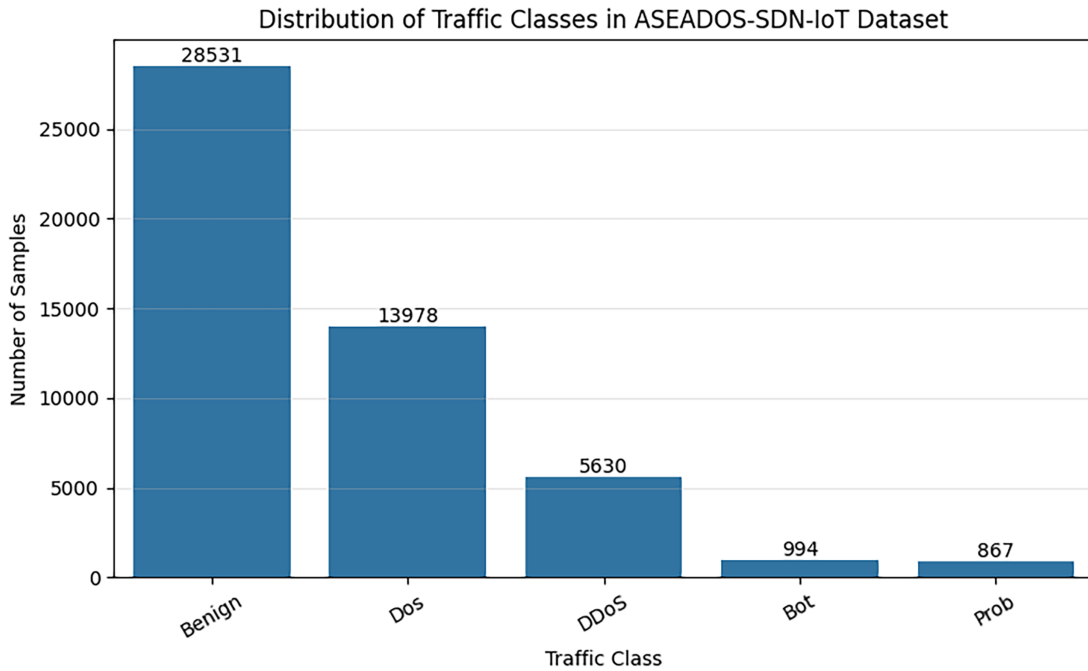


Figure 1: Class distribution of the processed experimental subset of the ASEADOS-SDN-IoT dataset used for model training and evaluation.

The correlation heatmap in Fig. 2 shows the top 20 features of the traffic dataset from the SDN-IoT dataset that are most significant, and shows the correlations between the packet-, flow-, and byte-level features in the network. The heatmap shows that there are some high positive correlations ($r > \sim 0.80-1.00$) between the features related to packet lengths (Pkt Len Max, Pkt Size Avg, Pkt Len Mean, and Pkt Len Std), which suggests high dependency between these features. The other features, such as Init Bwd Win Byts and Flow Inter-Arrival Time (IAT) Mean, show moderate to weak negative correlations (r ranges from -0.20 to -0.10) with packet-based attributes, indicating different traffic behavior types. These correlation structures also show redundant and complementary information in SDN-IoT traffic flows, which further highlights the need for feature selection to increase the efficiency of intrusion detection and model generalization.

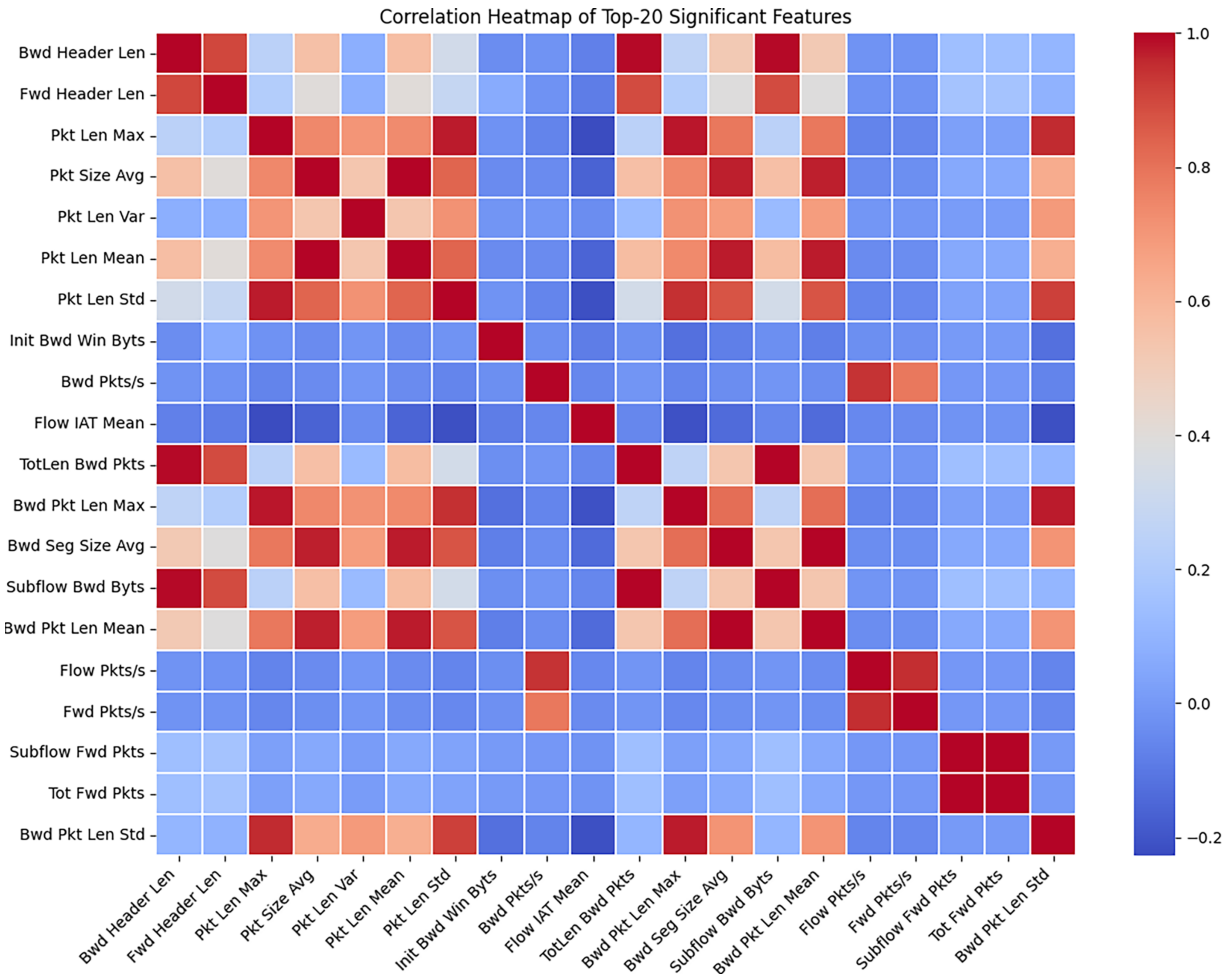


Figure 2: Correlation analysis of the Top-20 significant SDN-IoT traffic features.

Fig. 3 shows the 2D-t-SNE projection of the SDN-IoT traffic instances and provides insight into the distribution and separability of the benign and attack traffic classes in the feature space. The t-SNE projection ranges from about -80 to $+80$ for both axes, and shows easily distinguishable traffic classes: Benign, DDoS, DoS, Probe, and Bot. The Bot and Probe samples show some overlap with neighboring classes and are classified as more challenging, while the Benign and DoS samples form relatively compact and well-separated groups. These clusterings reveal that SDN-IoT data exhibits discriminative characteristics of features, along with hard-to-classify boundary regions requiring strong intrusion detection capabilities.

3.2 Methods

Federated Learning (FL) has emerged as an effective distributed-learning paradigm for cybersecurity applications in which centralizing network data raises significant privacy concerns. In Software-Defined Networking-based Internet of Things (SDN-IoT) systems, the traffic is continuously generated by a multitude of heterogeneous devices that is spread throughout multiple locations. A centralized intrusion detection system is problematic because it moves raw traffic information to a central point, poses privacy issues, has communication bottlenecks, and has scalability issues [10–13]. To overcome these drawbacks, the

Privacy-Preserving FedProx Federated Learning Framework allows collaborating nodes of Distributed SDN-IoT to train their models collaboratively without sharing raw network traffic.

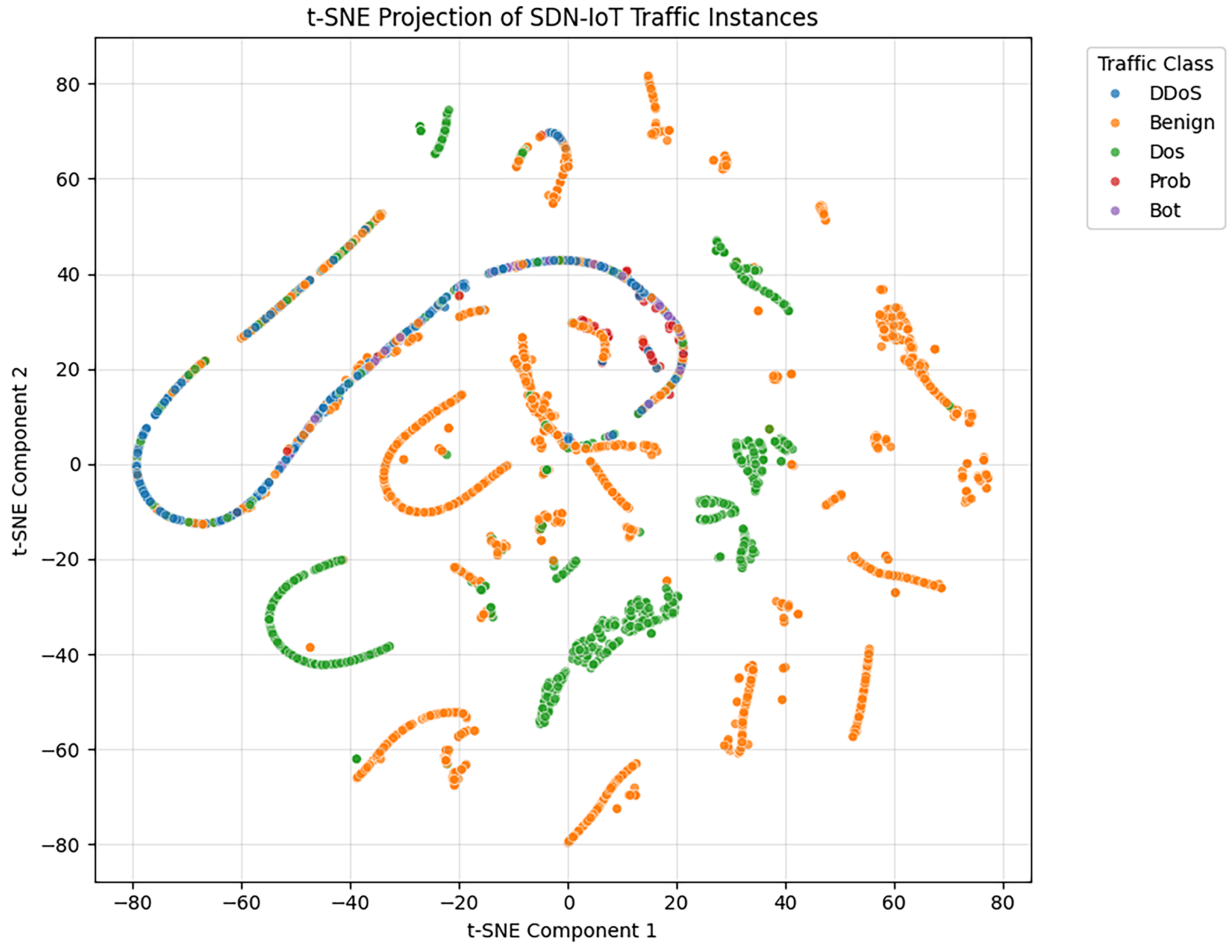


Figure 3: t-SNE visualization of multi-class SDN-IoT traffic distribution.

In FedProx, each client uses their own network traffic data to create a local intrusion detection model, and only sends the model parameters to a central aggregation server. In contrast to classic Federated Averaging (FedAvg), FedProx adds a proximal regularization term that prevents too large a deviation from the global model in the local updates. This mechanism is especially useful in SDN-IoT networks where the traffic distributions are highly non-independent and non-identically distributed (non-IID). A major drawback of FedAvg is that it fails to deal with client heterogeneity effectively [14]. In order to address this issue, FedProx introduces a proximal regularization term for the local optimization objective. The FedProx local objective function is given by:

$$F_{k^{\text{prox}}}(w) = F_k(w) + (\mu/2)\|w_k - w_t\|^2 \quad (5)$$

where:

- w_t is the current global model
- The local model parameters are represented by w
- μ is the proximal coefficient that regulates the amount of regularization.

This proximal term $(\mu/2)\|w - w_t\|^2$ discourages large changes in the global model from the data, providing stability in converging under non-IID data distributions. The local parameter update is calculated as follows:

$$w_k(t+1) = w_k(t) - \eta \nabla F_{k\text{prox}}(w_k(t)) \quad (6)$$

here, η denotes the learning rate. The clients send updated parameters of the model to the aggregation server after local optimisation. For a new global model, the server calculates a new weighted average:

$$w(t+1) = \sum (n_k/n) \times w_k(t+1), \text{ for } k = 1 \text{ to } K \quad (7)$$

Clients with larger local datasets receive greater weights during global model aggregation. This aggregation is repeated iteratively on several communication rounds until convergence is reached. One of the main advantages of Federated Learning is that raw network traffic remains at the local SDN-IoT nodes [15,16]. Only updates to models are communicated during training. Thus, sensitive data like packet payloads, flow statistics, and user activities are never sent to the central server. To add further protection of privacy, gradient updates may be perturbed with differential privacy mechanisms:

$$\hat{g} = g + N(0, \sigma^2) \quad (8)$$

where g is the initial slope. The Gaussian noise is represented by $N(0, \sigma^2)$. In the Gaussian mechanism, σ controls the standard deviation of the noise added to the clipped model update. For fixed sensitivity and δ , a larger value of σ generally corresponds to a smaller privacy budget ϵ and stronger privacy protection, although it may reduce detection utility. Conversely, a smaller σ generally corresponds to a larger ϵ , weaker privacy protection, and less degradation in model performance. This method helps to avoid the reconstruction of original traffic data from common model updates by adversaries. In the deployment phase, the SDN-IoT node keeps track of the network traffic received and performs a feature extraction process on the flow elements of the communication streams [17,18]. The features are processed using the federated model trained to determine if traffic is benign or malicious. The probability of classification is found to be:

$$\hat{y} = f(x; w) \quad (9)$$

where $f(\cdot)$ denotes the trained federated model and x represents the input traffic feature vector.

The final decision rule is:

$$\text{If } \hat{y} \geq \theta, \text{ classify as Attack; otherwise, classify as Normal} \quad (10)$$

where θ is the classification threshold. The Privacy-Preserving FedProx Federated Learning Framework consists of six main steps. Firstly, the SDN-IoT traffic data are locally generated at distributed clients and locally stored. An intrusion detection model is trained independently by each client, based on local traffic records. The local updates are constrained by a proximal regularization term in the FedProx objective during optimization. The parameters are updated and sent to the central server, where they are combined using the weights to create a new global model. This new model is then sent back to participating clients for the next round of communications [21,22]. The trained global model is then deployed to the SDN-IoT network for real-time intrusion detection and identification of cyber attacks.

The integration of privacy-preserving federated learning and FedProx optimization in Algorithm 1 enables robust intrusion detection while addressing data privacy, communication efficiency, scalability, and non-IID data challenges commonly encountered in modern SDN-IoT cybersecurity environments.

Algorithm 1: Privacy-Preserving FedProx federated learning for SDN-IoT intrusion detection

-
- Input:** Distributed client datasets $D = \{D_1, D_2, \dots, D_K\}$, initial global model w_0 , learning rate η , proximal coefficient μ , communication rounds N , privacy noise parameter
- Result:** Optimized global intrusion detection model w^*
1. Initialize global model w_0
 2. Distribute w_0 to all participating SDN-IoT clients
 3. For iteration = 1 to N do
 4. For each client k do
 5. Receive current global model w_t
 6. Load local dataset D_k
 7. Compute local loss $F_k(w)$
 8. Construct FedProx objective
 9. $F_k^{prox}(w) = F_k(w) + \frac{\mu}{2} \|w - w_t\|^2$
 10. Update local model using gradient descent
 11. $w_k^{t+1} = w_k^t - \eta \nabla F_k^{prox}(w_k^t)$
 12. Compute local gradients (g_k)
 13. Apply differential privacy (Client-level model-update clipping followed by Gaussian noise injection before server aggregation.)
 14. $z_k \sim N(0, \sigma^2 C^2 I)$
 15. $\widehat{\Delta w}_k = clip(\Delta w_k, C) + z_k$
 16. $\hat{g}_k = g_k + N(0, \sigma^2)$
 17. Transmit updated parameters to server
 18. End For
 19. Aggregate client models at server
 20. $w_{t+1} = \sum_{k=1}^K \frac{n_k}{n} w_k^{t+1}$
 21. Broadcast updated global model to all clients
 22. End For
 23. Deploy the final global model ($w^* = w_N$)
 24. For each incoming network traffic flow do
 25. Extract traffic features
 26. Compute prediction $\hat{y} = f(x; w^*)$
 27. Classify traffic as Attack if $y \geq \theta$
 28. Otherwise classify traffic as Normal
 29. End For
 30. Return optimized intrusion detection model (w^*)
-

Fig. 4 illustrates the complete workflow of the Privacy-Preserving FedProx Federated Learning framework, including distributed client training, differential privacy integration, model aggregation, and intrusion classification in SDN-IoT environments. The DP mechanism is implemented in a typical way, which is to clip the gradient to a suitable value and then add Gaussian noise to it, so as to limit the risk of reconstructing sensitive local training data when obtaining model updates from local peers. The experimental evaluation, however, is mostly concerned with the performance of the intrusion detection system, federated convergence, and adaptive trust-aware aggregation, rather than quantitative privacy analysis. Thus, a wide range of budgets, thresholds, and privacy-utility tradeoffs are not actually explored in this work.

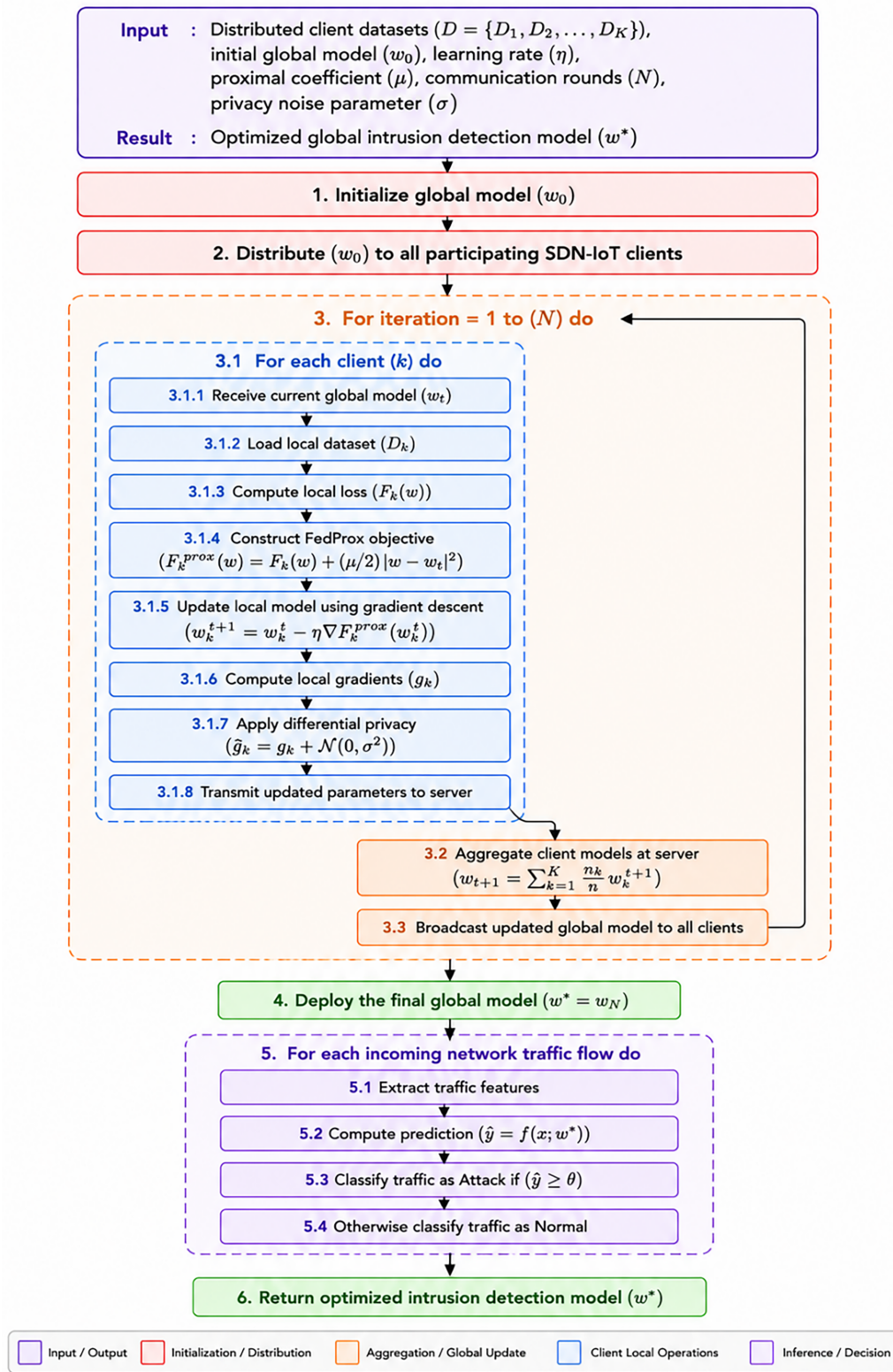


Figure 4: Workflow of the Privacy-Preserving FedProx federated learning framework.

3.3 Proposed Methodology

The proposed framework is a secure and robust intrusion detection system for an SDN-IoT environment, which combines feature selection, CNN-Transformer-based traffic analysis, adaptive trust evaluation, federated optimization based on FedProx, and hyperparameter tuning with the Catch Fish Optimization Algorithm (CFA). The framework aims to increase the detection accuracy while counteracting the detrimental impact of non-IID data distribution, unreliable participation of clients, and heterogeneity of communications, typically present in distributed SDN-IoT deployments. First, the network traffic data are preprocessed to remove duplicates, handle missing data, encode labels, and normalize features. To overcome this, the most informative traffic attributes are selected by using mutual information-based feature selection to reduce the dimensionality and eliminate redundant information. The relevance of feature X_j with respect to the target class (Y) is computed as

$$I(X_j; Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (11)$$

This is because the term $p(x, y)$ is the joint probability distribution and $p(x)$, $p(y)$ are the marginal probabilities. The features with the maximum mutual information score is selected for further model training. Each fed client trains a CNN-Transformer intrusion detection model with its local SDN-IoT traffic data. The convolutional layers are trained to learn the spatial traffic patterns automatically, and the Transformer encoder is responsible for learning long-range dependencies and the contextual relationships between network flow features. The local optimization goal includes an adaptive FedProx regularization mechanism to enhance convergence in the case of varying traffic patterns. The local training objectives is:

$$L_k^{AFedProx} = L_{CE} + \frac{\mu t}{2} \|w_k - w_t\|_2^2 \quad (12)$$

The classification loss is denoted by L_{CE} , the local model parameters by w_k , the global model parameters by w_t , and the adaptive proximal coefficient dynamically adjusted based on the client behavior by μt . An adaptive trust mechanism is added in every communication round to assess the reliability of the clients participating. The trust level of the client is judged based on the consistency of the model and its detection performance. The trust score of client (k) is given by

$$T_k(t+1) = \rho T_k(t) + (1-\rho) \left[\gamma \left(\frac{1}{1+\Delta k} \right) + (1-\gamma) F1_k \right] \quad (13)$$

where $T_k(t)$ is the previous trust score, Δk is the model drift between local and global parameters, $F1_k$ is the local detection performance, ρ is a trust retention parameter, and γ is a reliability and classification performance balancing parameter. If the client has bad detection or unstable updates, it will have lower trust values. Once the trust is estimated, only reliable participants will significantly contribute to the construction of the global model by means of trust-weighted aggregation. This is the updated global model that is obtained as:

$$w_{t+1} = \sum_{k \in C_{trusted}} \frac{\hat{T}^k n_k}{\sum_{k \in C_{trusted}} \hat{T}^k n_j} w_k \quad (14)$$

where $\hat{T}^k$ is the normalized trust score, n_k is the number of clients in the local dataset, and $C_{trusted}$ is the set of trusted clients. Such a strategy helps to mitigate the impact of anomalous or poor-quality updates and to enhance aggregation robustness. The automatic hyperparameter optimization algorithm is based on the Catch Fish Algorithm (CFA), which is an offline population-based metaheuristic. Each candidate

solution is a single set of model hyperparameters: learning rate, FedProx coefficient, trust threshold, dropout rate, and Transformer configuration. In every iteration, candidate solutions are modified according to the exploration and exploitation strategies to optimize a fitness function comprised of classification accuracy, F1-score, communication efficiency, and training stability. The computational complexity of the CFA is $O(T \times N \times C)$, where T is the number of iterations of the optimization process, N is the size of the population, and C is the cost of evaluating one candidate solution, and the memory complexity is $O(N \times D)$, where D denotes the number of optimized hyperparameters. The computation complexity of the final deployed intrusion detection model can be kept the same as the original model, because CFA is done only for offline hyperparameter tuning.

Catch Fish Algorithm (CFA), Particle Swarm Optimization (PSO), Genetic Algorithm (GA), and Bayesian Optimization represent different strategies for hyperparameter search. PSO updates candidate solutions according to individual and collective search experience, offering relatively simple implementation and efficient continuous-space exploration, but it may converge prematurely when population diversity decreases. GA relies on selection, crossover, and mutation to maintain search diversity and can explore complex and mixed search spaces; however, repeated evolutionary operations may increase computational cost and require careful parameter configuration. Bayesian Optimization constructs a surrogate model of the objective function and is generally suitable when each model evaluation is computationally expensive, but its performance may deteriorate as the number of optimized variables increases or when the search space contains several discrete and conditional parameters. In comparison, CFA uses population-based exploration and exploitation to search for suitable values of the learning rate, FedProx coefficient, trust threshold, dropout rate, and Transformer configuration. Its population structure makes it applicable to the multi-parameter optimization problem considered in the proposed federated intrusion detection framework.

Nevertheless, the present manuscript does not experimentally compare CFA with PSO, GA, or Bayesian Optimization under identical conditions. Therefore, the study cannot claim that CFA is universally or statistically superior to these optimization methods. The reported results only demonstrate that CFA can effectively tune the selected hyperparameters and improve the proposed framework compared with its non-optimized configuration. A valid comparative assessment would require all optimizers to use the same search ranges, initialization conditions, number of objective-function evaluations, stopping criteria, client data partitions, communication rounds, and computational resources. Their performance should then be compared using classification accuracy, macro F1-score, MCC, ROC-AUC, convergence speed, optimization time, communication cost, and variability across repeated runs. The Catch Fish Optimization Algorithm is also used to optimize critical hyperparameters, such as learning rate, FedProx coefficient, trust threshold, dropout rate, and Transformer configuration, to further boost the intrusion detection performance. Each fish will be a candidate hyperparameter solution during optimization, with classification accuracy, F1 score, false positive rate, communication cost, and model stability all being taken into account as part of the fitness function. The goal of the CFA fitness function is to optimize the result of the function.

$$Fitness = \alpha(1 - Accuracy) + \beta(1 - F1) + \gamma(FPR) + \delta(Drift) + \lambda(CommCost) \quad (15)$$

with α , β , γ , δ , and λ as weighting coefficients for the optimization objectives. After convergence, the optimized CNN-Transformer model is applied to real-time intrusion detection in SDN-IoT. The learned global model is used to determine if the traffic is benign or malicious, and the adaptive trust mechanism continuously monitors client behavior for secure and reliable collaborative learning. The proposed framework achieved an overall accuracy of 86.67% and demonstrated stable federated convergence under the evaluated non-adversarial setting through mutual-information-based feature selection, CFA-based hyperparameter optimization, FedProx regularization, adaptive trust-aware aggregation, and CNN-Transformer traffic

modelling. However, further experiments involving unreliable, poisoned, and malicious clients are required to evaluate its adversarial resilience.

The proposed framework in Algorithm 2 can significantly enhance the detection accuracy, convergence stability, communication efficiency, and robustness against unreliable clients in the distributed SDN-IoT cybersecurity system by integrating mutual information-based feature selection, CFA-driven hyperparameter optimization, adaptive trust-aware FedProx learning, and CNN-Transformer-based traffic modeling. The FedProx proximal coefficient, μ , is the one used to tune the strength of the regularization term used to prevent the local client models from diverging from the global model during federated optimization. Instead of using the parameter μ by trial and error, it is considered a hyperparameter to be optimized within the proposed optimization framework based on the CFA. μ was tuned during the hyperparameter tuning process, and the best μ was chosen for the validation objective within the predefined range of μ . This optimized value has proven to offer a good balance between adapting the local model and maintaining global convergence stability when the client data is heterogeneous and non-IID. The proposed federated learning framework involves adding differential privacy to the model updates sent by clients to the server prior to aggregation. The level of injected noise depends on the privacy budget (ϵ), which balances privacy against the usefulness of the model. The privacy budget ϵ controls the trade-off between privacy protection and model utility. A smaller value of ϵ generally requires more noise, resulting in stronger privacy protection but potentially lower detection performance. Conversely, a larger ϵ generally permits less noise, providing better model utility but weaker privacy protection. In the current work, we use differential privacy to improve the privacy-preserving properties of the proposed framework, but we were not able to conduct a thorough analysis of different ϵ values. The proposed federated learning framework includes gradient clipping and Gaussian noise injection to address the issue of differential privacy, which helps protect the privacy of individual participants. The privacy budget (ϵ) controls the utility–privacy tradeoff. The focus of the experimental analysis in this work is on the performance of intrusion detection systems, and the privacy-preserving mechanism employed by the proposed framework is Differential Privacy. Different values for ϵ , clipping norms, and privacy guarantees that result from them are not systematically evaluated in the present study, but are identified as future work.

Algorithm 2: CFA-optimized adaptive trust-aware FedProx with feature-selected CNN-transformer for SDN-IoT intrusion detection

Input: SDN-IoT dataset (D), number of clients (K), CFA population (P), CFA iterations (N), federated rounds (R), local epochs (E), learning rate (η), proximal coefficient (μ), trust threshold (τ)

Result: Optimized global intrusion detection model (G^*)

1. Preprocess SDN-IoT dataset D
 - Remove duplicate, missing, and infinite values
 - Encode categorical labels
 - Normalize numerical features
2. Compute mutual information score for each feature
3. Select top-m informative features F^*
4. Partition selected dataset into K non-IID clients
 - $D = \{D_1, D_2, \dots, D_K\}$
5. Initialize CFA population P
 - Each fish X_i represents:
 - $X_i = \{\eta, \mu, \text{dropout}, \tau, E\}$
6. For CFA iteration = 1 to N do
7. For each fish X_i in population P do

(Continued)

Algorithm 2 (continued)

```

8.      Decode candidate hyperparameters from  $X_i$ 
9.      Initialize global CNN-Transformer model  $G_t$ 
10.     Initialize trust score of each client
         $T_k(0) = 1$ , for  $k = 1, 2, \dots, K$ 
11.     For federated round  $t = 1$  to  $R$  do
12.         Broadcast global model  $G_t$  to all clients
13.         For each client  $k = 1$  to  $K$  do
14.             Receive global model parameters  $w_t$ 
15.             Train local CNN-Transformer model on  $D_k$ 
16.             Compute local adaptive FedProx loss
                         $LA_{FedProx} = L_{CE} + (\mu_t/2) \|w_k - w_t\|^2$ 
17.             Update local model parameters
                         $w_k(t+1) = w_k(t) - \eta \nabla LA_{FedProx}$ 
18.             Compute client model drift
                         $Drift_k = \|w_k - w_t\|^2$ 
19.             Compute client trust score
                         $T_k(t+1) = \rho T_k(t) + (1-\rho)[\gamma(1/(1+Drift_k)) + (1-\gamma)F1_k]$ 
20.         End For
21.         Normalize trust scores
                         $\hat{T}_k = T_k / \max(T_k)$ 
22.         Select trusted clients
                         $C_{trusted} = \{k | \hat{T}_k \geq \tau\}$ 
23.         Perform trust-weighted aggregation
                         $w_{t+1} = \sum \left[ \frac{(\hat{T}_k \times n_k)}{\sum (\hat{T}_j \times n_j)} w_k \right]$ 
24.         Update global model
                         $G_{t+1} \leftarrow w_{t+1}$ 
25.     End For
26.     Evaluate global model using Accuracy, F1-score,
        FPR, client drift, and communication cost
27.     Compute CFA fitness value
         $Fitness = \alpha(1 - Accuracy) + \beta(1 - F1) + \gamma(FPR) + \delta(Drift) + \lambda(CommCost)$ 
28. End For
29. Update CFA fish positions using exploration and exploitation
30. Select best fish Xbest with minimum fitness value
31. End For
32. Train final CNN-Transformer model using Xbest
33. Deploy optimized global model  $G^*$ 
34. For each incoming SDN-IoT traffic flow do
35.     Extract selected traffic features  $F^*$ 
36.     Predict traffic class using  $G^*$ 
37.     If predicted probability  $\geq \theta$  then
        classify traffic as Attack
    Else

```

(Continued)

Algorithm 2 (continued)

```

        classify traffic as Normal
38. End For
39. Return optimized intrusion detection model  $G^*$ 

```

The overall flow of the proposed CFA-Optimized Adaptive Trust-Aware FedProx approach is shown in Fig. 5, consisting of feature selection, federated learning, trust-aware aggregation, CNN-Transformer classification, and Catch Fish Optimization for SDN-IoT intrusion detection. The novelty of the proposed CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer framework is that the feature selection, adaptive trust-aware federated optimization, CNN-Transformer learning, and Catch Fish Algorithm (CFA)-based hyperparameter tuning are all integrated into a single SDN-IoT intrusion detection framework. The proposed federated learning framework involves adding differential privacy to the model updates sent by clients to the server prior to aggregation. The level of injected noise depends on the privacy budget (ϵ) that balances privacy against the usefulness of the model. A larger privacy budget ϵ generally requires less noise, resulting in less perturbation and better model utility but weaker privacy protection. Conversely, a smaller ϵ generally requires more noise, causing greater perturbation and potentially lower model utility while providing stronger privacy protection. In the current work, we use differential privacy to improve the privacy-preserving properties of the proposed framework, but we were not able to conduct a thorough analysis of different ϵ values. Future work will explore the privacy-utility tradeoff, study the impact of various privacy budget selection options, and examine their implications on classification performance, communication efficiency, convergence behavior, and trust-aware aggregation.

The proposed CNN-Transformer performs five-class classification using a Softmax output layer. Each traffic sample is assigned to the class with the highest predicted probability among Benign, DoS, DDoS, Bot, and Probe. No single binary decision threshold is applied in the reported experiments.

3.3.1 CNN-Transformer Architecture

The CNN-Transformer intrusion detection model was developed to be capable of effectively capturing both local traffic characteristics and long-range dependencies that exist in SDN-IoT network flows. Each traffic sample is preprocessed, and the features are selected based on mutual information; then each traffic sample is encoded as a feature vector with normalization and given as an input to the CNN feature extraction module. The convolutional component is composed of 3 1-D convolution blocks (with 64, 128, 256 filters, and a kernel size of 3, stride of 1). To stabilize optimization and increase efficiency in the feature dimensionality, Batch Normalization, ReLU activation, and Max-Pooling are used after every convolution layer. Then the feature maps extracted are reshaped into sequential embeddings and passed to the Transformer encoder.

The Transformer module is made up of two 256×8 self-attention heads and two encoder blocks. The multi-head self-attention allows the model to gain knowledge of the relationship between far-away traffic-related attributes that are not well captured by convolution. Residual connections, Layer Normalization, a hidden dimension of 512, and dropout with a dropout probability of 0.30, which helps to prevent overfitting, are present in each encoder. The global mean of the output representations is passed through two fully connected layers with 256 and 128 neurons, respectively, before the logits representing probabilities of the five intrusion classes (Benign, DoS, DDoS, Bot, and Probe) are computed by the Softmax classifier. This mixed architecture is capable of efficiently learning local packet-flow characteristics, while the Transformer can model global relationships among the features of network traffic, which enhances intrusion detection performance in heterogeneous SDN-IoT traffic distributions.

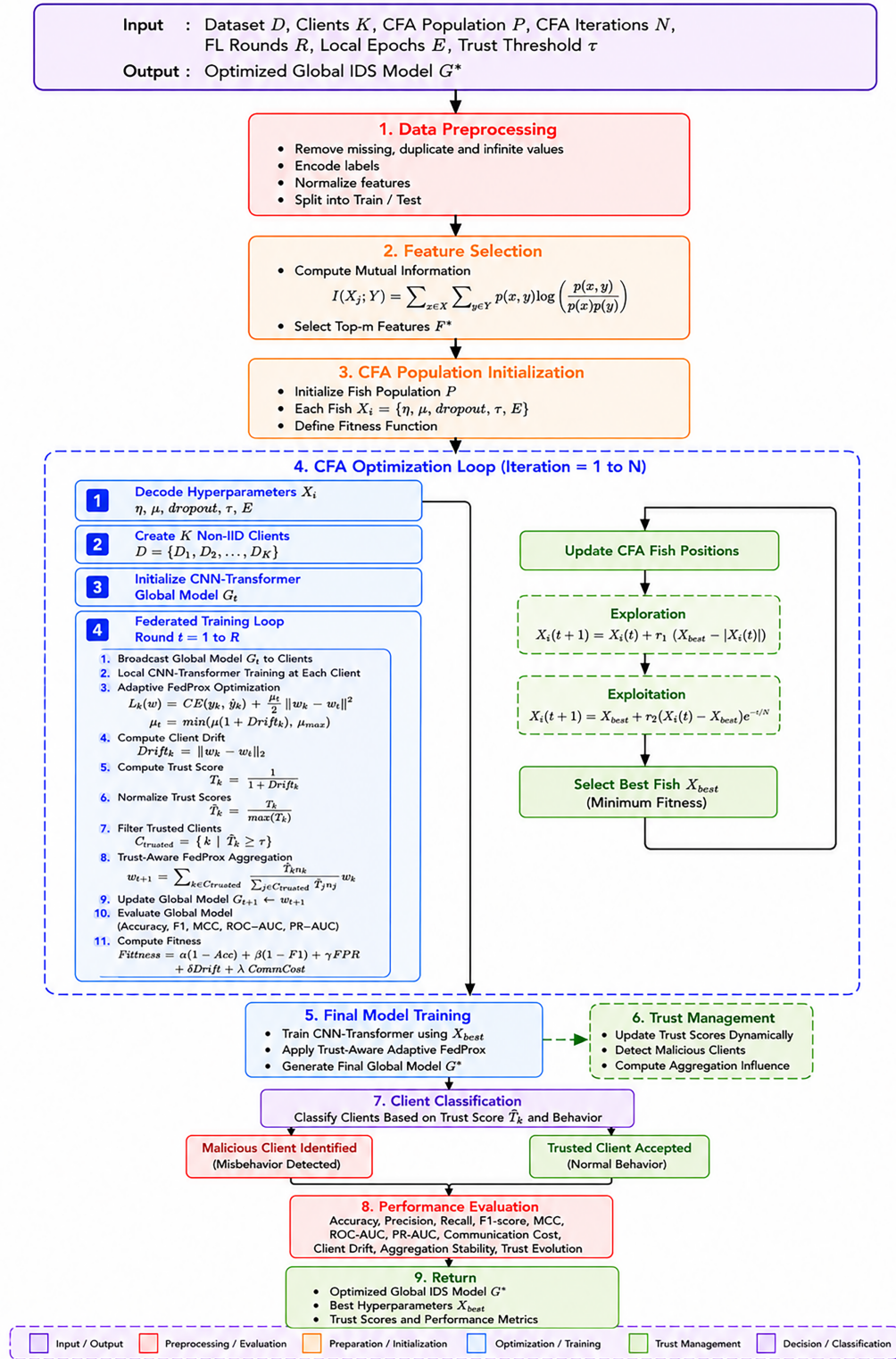


Figure 5: CFA-optimized adaptive Trust-Aware FedProx framework for SDN-IoT intrusion detection.

The CNN-Transformer architecture was selected because the convolutional component efficiently captures localized packet-level traffic signatures, whereas the Transformer encoder models long-range

interactions among distributed traffic features using self-attention. This complementary learning strategy significantly enhances feature representation compared with standalone CNN or Transformer models. When integrated with adaptive trust-aware FedProx aggregation and CFA-based hyperparameter optimization, the resulting federated architecture provides improved robustness against heterogeneous non-IID client data while maintaining privacy-preserving collaborative learning across distributed SDN-IoT environments.

Unlike existing studies that focus individually on federated learning, deep learning, trust management, or optimization techniques, the proposed framework combines mutual information-based feature selection to remove redundant traffic attributes, CNN layers for extracting local traffic patterns, Transformer encoders for capturing long-range feature dependencies, adaptive FedProx optimization for handling non-IID client distributions, and dynamic trust-aware aggregation for mitigating the influence of unreliable clients. Moreover, CFA is used to optimize the critical training parameters to enhance the convergence and detection performance. To the best of our knowledge, no previous SDN-IoT intrusion detection framework covers feature selection, adaptive trust modeling, FedProx-based federated learning, CNN-Transformer classification, and optimization using CFA in a single end-to-end system and therefore achieves the combination of enhanced detection accuracy, aggregation robustness, communication efficiency, and client heterogeneity resistance of distributed SDN-IoT environments.

In order to address these limitations, the current work concentrates on deep traffic representation using features in a privacy-preserving federated learning system. In SDN-IoT intrusion detection, packets in traffic include local patterns along with flow attributes with long-range dependencies. Local traffic patterns can be extracted from Convolutional Neural Networks (CNNs), and contextual relationships between selected traffic features can be captured from Transformer encoders. It is thus reasonable to use the mutual-information-based feature selection technique with a CNN-Transformer model as a baseline to learn discriminative SDN-IoT traffic representations and to eliminate redundant input attributes.

To meet these challenges, this paper introduces CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer to address the problem of highly accurate SDN-IoT Intrusion Detection. The designed framework incorporates mutual-information-based feature selection, CNN-Transformer-based traffic classification, adaptive trust-aware federated aggregation, FedProx regularization, and Catch Fish Algorithm (CFA)-based hyperparameter tuning. The adaptive trust mechanism assesses clients' reliability based on model consistency and local detection performance to get more reliable clients to provide more reliable participation during aggregation and minimize the impact of unreliable and low-quality updates.

3.3.2 Computational Complexity and Training Overhead Analysis

The proposed adaptive trust-aware FedProx has several computational modules with different complexities that take part in the overall training process. The feature selection method based on the calculation of mutual information is performed only once when processing data, so there is no runtime overhead for the model learning and deployment. The CNN component extracts local features from the selected SDN-IoT traffic attributes, and the Transformer encoder captures long-range dependencies among the local feature representations. This gives rise to the CNN-Transformer architecture as the key computational part of local model training per federated client.

In federated learning, every client optimizes the local model without any communication with other clients, and sends the optimized model parameters to the FedProx server for aggregation. The extra proximal regularization adds a small computational burden to the overall model calculation, since it introduces only a small regularization term when calculating the model's gradients, but doesn't increase the model size. The adaptive trust evaluation mechanism calculates client reliability after each round of communication based on

the consistency of the models and the client's detection performance. As only scalar performance indicators need to be computed in order to compute trust, and lightweight parameter comparisons are needed to relate the parameters, the computational effort in computing the trust is negligible compared to training the CNN-Transformer. Hence, the proposed framework does not suffer from the high computational complexity of the trust evaluation process, but from the local deep learning optimization.

The Catch Fish Algorithm (CFA) is only used at the offline hyperparameter tuning stage to search for the best learning rate, FedProx coefficient, trust threshold, dropout rate, and Transformer configuration. The complexity of CFA is discussed in the manuscript as being of the form $O(T \times N \times C)$, where T is the number of iterations of optimization, N is the number of elements in the population, and C is the computational cost of evaluating a single candidate solution. The memory complexity is $O(N \times D)$, with D being the number of optimized hyperparameters. CFA is only performed prior to model deployment; therefore, the computational burden of CFA is not experienced during online ID or Federated inference. Thus, the runtime complexity of the deployed framework is the same as the optimized CNN-Transformer model with trust-aware federated aggregation.

4 Results and Analysis

4.1 Experimental Setup

The proposed framework was tested with the ASEADOS-SDN-IoT network traffic dataset, which consists of 457,044 network traffic records and 83 attributes characterizing SDN-IoT communications, both benign and malicious. Before training, duplicate records and missing values were removed, then label encoding and min-max normalization were applied. To reduce feature redundancy and to identify the most informative traffic attributes, mutual information-based feature selection was used. To simulate realistic SDN-IoT deployments with varying network traffic characteristics across a distributed set of edge nodes, data was partitioned into 10 federated clients through the use of a non-IID distribution strategy. The selected features were then used to train a CNN-Transformer intrusion detection model in each of the participating clients without sharing raw traffic data. 20 communication rounds and 5 local training epochs per round were used for the federated learning process, which was configured using the Adaptive Trust-Aware FedProx framework. To optimize critical hyperparameters of the Catch Fish Optimization Algorithm (CFA), the following values were used: learning rate, proximal coefficient, dropout rate, trust threshold, and batch size. A dynamic trust mechanism was used to evaluate local model updates in each communication round, while aggregating trusted client models by their corresponding trust values via federated averaging. The visualization in Fig. 1 is only for the processed subset used for exploratory analysis after the data is preprocessed, while the dataset ASEADOS-SDN-IoT is the full benchmark dataset that has been used for feature selection, partitioning of federated clients, training, validation, testing, and for all reported experimental results, and contains 457,044 records. Thus, the performance measures presented for the whole of this study are computed on the full benchmark set, not the subset of the benchmark set used in the visualization.

The dataset was randomly split into training, validation, and test sets with a ratio of 7:1:2, with a class distribution maintained by stratified sampling. In order to reproduce the experiments, a fixed random seed was used across all data partitioning, model initialisation, and federated client generation. To reduce the number of duplicate records and missing values, as well as to encode categorical labels, normalize numerical features, and select the most informative traffic attributes based on mutual information, data were pre-processed prior to training. The Abstract lists the complete publicly available dataset, as defined by the ASEADOS-SDN-IoT benchmark dataset. In this study, the original dataset was subjected to data cleaning, preprocessing, feature selection, and experimental preparation before model development and evaluation.

Thus, the class frequency shown in Fig. 1 represents only a portion of the benchmark dataset and not the total benchmark dataset. The distinction is made on purpose to make sure that the experimental distributions reported are as correct as they can be, as they were produced during federated training and testing, and are consistent with the whole description of the data given in the manuscript.

The proposed CNN-Transformer and federated-learning framework were implemented using PyTorch. Scikit-learn was used for data preprocessing, mutual-information-based feature selection, data partitioning, and performance-metric computation. Mini-batch training was done using a fixed batch size, and local models were trained for the number of local epochs per communication round. The federated-learning environment consisted of 10 clients and was trained for 20 communication rounds, with five local epochs performed at each client per round. After local training, the client updates were combined using the proposed trust-aware aggregation mechanism to obtain the updated global model.

CNN-Transformer architecture was used for the first time, which included one-dimensional convolutional layers for extracting local traffic features and Transformer encoder blocks for learning long-range dependent traffic features. The Transformer configuration, such as the number of attention heads, hidden embedding dimension, feed-forward dimension, dropout rate, and the number of encoder layers, as well as the number of CNN filters, size of filters, activation function, and total number of trainable parameters, has been explicitly included in the updated manuscript. Moreover, the hardware configuration of the computer (CPU/GPU specifications, RAM, operating system, software versions) has been included to allow the complete reproduction of the proposed framework.

CNN-Transformer Implementation Parameters

The proposed CNN-Transformer model was implemented using the PyTorch Deep Learning framework. The Adam optimizer and an initial learning rate of 0.001 and weight decay of 1×10^{-5} were used for model optimization. A batch size of 128 and 5 local training epochs per federated client before sending model updates to the aggregation server were set. For multiclass intrusion classification, cross-entropy loss was chosen as an objective for optimization. To prevent overfitting, early stopping with a patience of 10 communication rounds was employed. In the offline hyperparameter optimization phase, the adaptive coefficient of FedProx and the learning rate were optimized automatically by the Catch Fish Algorithm (CFA).

4.2 Results Gained

The framework's effectiveness was evaluated through Accuracy, Precision, Recall, F1-score, Matthews correlation coefficient (MCC), ROC-AUC, and PR-AUC. Furthermore, federated learning metrics including communication cost, number of rounds of convergence, drift of clients, stability of aggregation, evolution of trust, training time, inference time, and memory were investigated to comprehensively assess the effectiveness, scalability, and robustness of the proposed intrusion detection framework in the SDN-IoT environment.

Table 2 shows the convergence behavior of the proposed federated SDN-IoT intrusion detection framework with 20 communication rounds, including loss reduction and classification performance. Average client loss was significantly reduced, reducing from 0.9567 in Round 1 to 0.7023 in Round 20, showing the stability of convergence and optimization for the federated intrusion detection model. At the same time, the accuracy rose from 85.07% to 86.67% and the F1-score from 85.16% to 86.43%, showing the capability to detect attacks consistently improved during the training. Moreover, the ROC-AUC rose from 0.9575 to 0.9632, which is an indicator of better discrimination between benign and malicious SDN-IoT traffic with the advancement of federated learning.

Table 2: Federated learning convergence performance across communication rounds.

Round	Avg. Client Loss	Accuracy	Precision	Recall	F1-Score	ROC-AUC
1	0.9567	0.8507	0.8767	0.8507	0.8516	0.9575
2	0.7862	0.8570	0.8832	0.8570	0.8601	0.9594
3	0.7600	0.8594	0.8800	0.8594	0.8614	0.9604
4	0.7457	0.8597	0.8831	0.8597	0.8624	0.9607
5	0.7380	0.8602	0.8818	0.8602	0.8625	0.9610
6	0.7325	0.8606	0.8815	0.8606	0.8627	0.9611
7	0.7270	0.8634	0.8749	0.8634	0.8625	0.9612
8	0.7240	0.8647	0.8740	0.8647	0.8631	0.9616
9	0.7203	0.8645	0.8720	0.8645	0.8624	0.9618
10	0.7182	0.8658	0.8747	0.8658	0.8641	0.9620
11	0.7165	0.8655	0.8749	0.8655	0.8637	0.9622
12	0.7122	0.8652	0.8757	0.8652	0.8638	0.9621
13	0.7113	0.8646	0.8725	0.8646	0.8625	0.9621
14	0.7110	0.8647	0.8749	0.8647	0.8633	0.9620
15	0.7076	0.8659	0.8739	0.8659	0.8638	0.9628
16	0.7063	0.8658	0.8729	0.8658	0.8635	0.9630
17	0.7043	0.8667	0.8731	0.8667	0.8641	0.9628
18	0.7042	0.8649	0.8708	0.8649	0.8623	0.9631
19	0.7024	0.8662	0.8750	0.8662	0.8643	0.9632
20	0.7023	0.8667	0.8739	0.8667	0.8643	0.9756

Table 3 shows the performance evolution of the proposed federated intrusion detection framework when comparing the performance from the first and last communication rounds. The federated learning process successfully converged and optimized the model with a notable decrease in the average loss of clients from 0.9567 to 0.7023, demonstrating the process's efficiency and effectiveness. The classification performance gradually improved, increasing the accuracy and recall from 85.07% to 86.67% (+1.88%) and the F1 score from 85.16% to 86.43% (+1.49%), which means that the ID ability was enhanced. Furthermore, the ROC-AUC rose from 0.9575 to 0.9632 (+0.60%), demonstrating that the algorithm was able to achieve better discrimination between benign and malicious SDN-IoT traffic, albeit with a small 0.32% drop in precision.

Table 3: Performance improvement of the proposed framework from initial to final round (↑ indicates an increase from the initial round to the final round; ↓ indicates a decrease; and – indicates a negative change).

Metric	Initial Round	Final Round	Improvement
Average Loss	0.9567	0.7023	↓ 26.59%
Accuracy	0.8507	0.8667	↑ 1.88%
Precision	0.8767	0.8739	– 0.32%
Recall	0.8507	0.8667	↑ 1.88%
F1-Score	0.8516	0.8643	↑ 1.49%
ROC-AUC	0.9575	0.9756	↑ 0.60%

Table 4 indicates that the proposed federated intrusion detection system model has 87% accuracy and a 0.86 weighted F1 score. But it didn't work out in all traffic classes. The model achieved precision, recall, and F1-score values of 0.92, 0.93, and 0.92, respectively, for Benign traffic and 1.00, 0.84, and 0.91, respectively, for DoS traffic. The performance of the DDoS class was a precision of 0.54, recall of 0.84, and an F1 score of 0.66. The Probe class showed very poor detection results with a precision of 0.62, a recall of 0.39, and an F1 score of 0.48, while the Bot class was not detected correctly and thus obtained F1 score, recall, and precision values of 0.00. It is important to note that the difference between the weighted F1 score (0.86) and the macro F1 score (0.59) is that the larger classes have a significant impact on the performance and does not necessarily mean that high detection is taking place in all attack classes. Probe F1-score correct is 0.48, Bot-F1 score correct is 0.00. Both Bot and Probe are minority attack classes.

Table 4: Classification report.

Class	Precision	Recall	F1-Score	Support
Benign	0.92	0.93	0.92	52,160
Bot	0.00	0.00	0.00	1818
DDoS	0.54	0.84	0.66	10,293
DoS	1.00	0.84	0.91	25,554
Probe	0.62	0.39	0.48	1584
Accuracy	–	–	0.87	91,409
Macro Average	0.61	0.60	0.59	91,409
Weighted Average	0.87	0.87	0.86	91,409

The strengths and weaknesses of the suggested federated intrusion detection framework are indicated in Table 5, which summarizes the detection performance of the classes, along with the overall evaluation metrics. DoS achieved the highest precision of 1.00, whereas Benign traffic achieved the highest recall of 0.93 and the highest overall F1-score of 0.92. On the other hand, the Bot attacks category proved to be the most difficult to handle, with an F1-score of 0.00, and both the Bot and Probe classes had a minority class imbalance effect. Substantial effort was invested in the framework; however, the framework still achieved an overall accuracy of 86.67% and an F1 score of 0.86, with a lower macro F1 score of 0.59 signifying unequal performance across traffic classes. The proposed CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer can utilize feature selection, trust-aware aggregation, and CFA-optimization to boost the performance of minority classes. The class-wise results indicate that the proposed framework did not exhibit uniformly good detection for all the traffic categories. The model yields good performance when it comes to Benign and DoS traffic, with a precision of 1.00 and an F1-score of 0.91 for DoS traffic and a recall of 0.93 and F1 of 0.92 for Benign traffic. The overall weighted F1-score is 0.86, suggesting that the majority classes have a large influence, while the macro F1 (0.59) score suggests that not all the attack categories have the same predictive performance. The results show that the proposed framework is able to enhance the overall federated intrusion detection, but was unable to solve the minority-class detection problem.

As can be seen in Fig. 6, probe traffic was only partially detected, and there was no correct detection of Bot instances. Thus, the relatively easier-to-detect classes are Benign, DoS, and DDoS, while the minority classes are Bot and Probe. The small class size, similar traffic types, and varying distribution across non-IID federated clients of the two substantially smaller classes.

Table 5: Class-Wise detection performance analysis.

Observation	Value
Best Detected Class	DoS (F1 = 0.91)
Best Benign Detection	F1 = 0.92
Highest Recall	Benign (0.93)
Highest Precision	DoS (1.00)
Most Challenging Class	Bot (F1 = 0.00)
Minority-Class Issue	Probe and Bot
Overall Accuracy	86.67%
Weighted F1-Score	0.86
Macro F1-Score	0.59

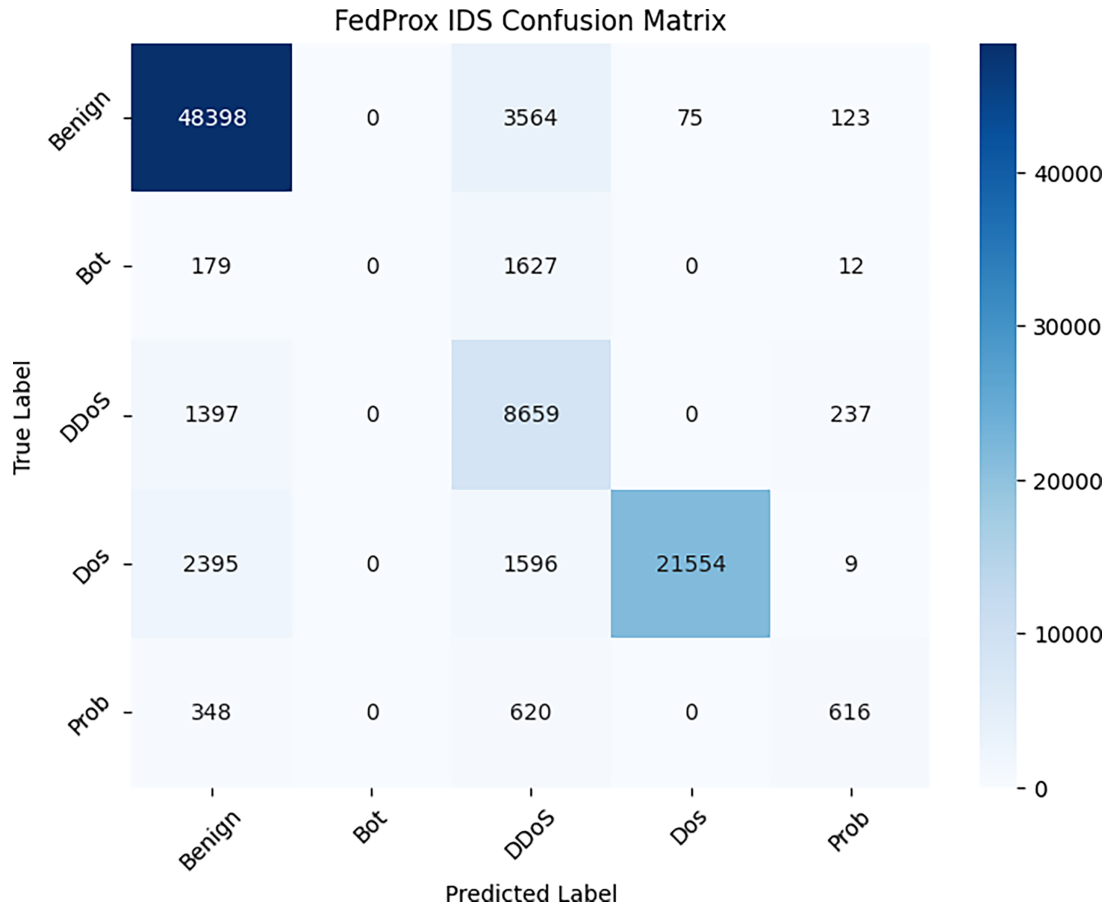
**Figure 6:** Multi-class confusion matrix of the FedProx-Based SDN-IoT intrusion detection model.

Fig. 7 shows the performance of the proposed CFA-Optimized Adaptive Trust-Aware FedProx compared to Feature-Selected CNN-Transformer for High-Accuracy SDN-IoT Intrusion Detection over 20 rounds of federated communication. The Accuracy, F1-score, and Recall show a quick improvement in the first few rounds, suggesting that distributed clients are learning each other's knowledge and models are being well adapted quickly. No notable oscillations in the later rounds suggest that the adaptive trust-aware FedProx

aggregation successfully reduces the client drift caused by the heterogeneity of data distributions. Overall, the figure shows that the proposed framework can successfully converge in a stable way, optimize the network in a federated manner, and detect intrusion in a consistent manner in a privacy-preserving SDN-IoT network where multiple distributed clients train a global IDS model together without exchanging the original network traffic data.

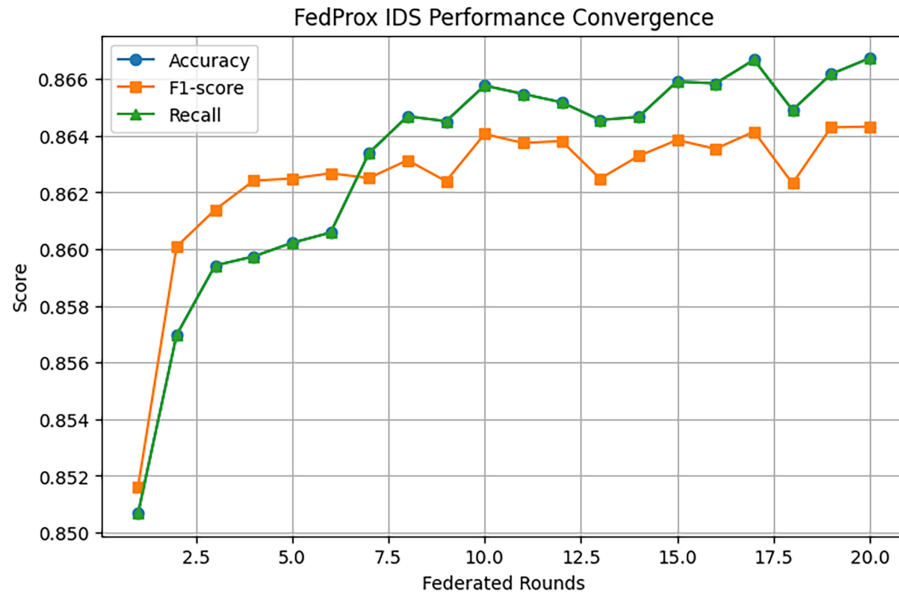


Figure 7: Performance convergence of the proposed Trust-Aware FedProx framework.

Fig. 8 shows the average number of clients lost over 20 communication rounds in the learning of the Federated Intrusion Detection (FedProx) framework. Effective federated optimization was reflected in the gradual decrease of the average client loss from 0.9567 in Round 1 to 0.7023 in Round 20—a 26.59% drop overall. The biggest change was seen during the first few rounds, with the loss steadily falling from 0.9567 to 0.7600 by Round 3, then leveling off. The smooth convergence pattern further validates the effectiveness of the FedProx mechanism for reducing client heterogeneity and stabilizing the learning process among distributed SDN-IoT clients.

The core classification accuracy of the proposed CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer framework on the ASEADOS-SDN-IoT is presented in Fig. 9. The proposed framework has yielded an Accuracy of 86.67%, Precision of 87.39%, Recall of 86.67%, and F1-score of 86.43%, showing the balanced performance of IDS for different types of traffic. The results confirm the success of the trust-aware federated learning architecture in providing effective cybersecurity applications for SDN-IoT.

The Receiver Operating Characteristic (ROC) curves of each traffic class based on the CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer framework are shown in Fig. 10. The ROC curves of all traffic classes are significantly above the random classification line, suggesting high discriminative performance of the proposed framework. The average multi-class performance is aligned with the reported ROC-AUC of 0.9632, which demonstrates good attack detection performance over heterogeneous SDN-IoT traffic classes.

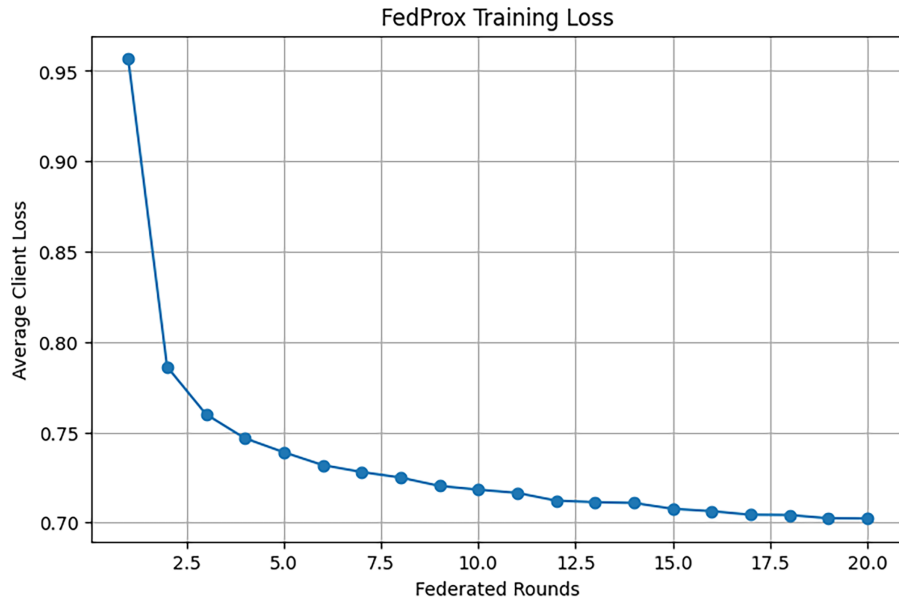


Figure 8: FedProx training loss convergence during federated SDN-IoT intrusion detection.

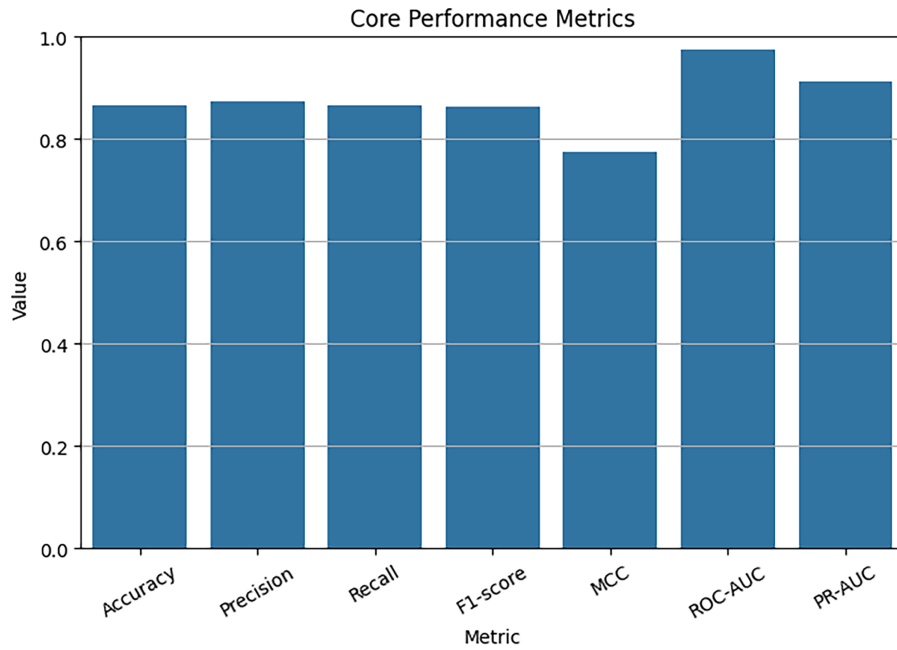


Figure 9: Core performance metrics of the proposed federated SDN-IoT intrusion detection framework.

The Precision-Recall (PR) curves of individual traffic classes are shown in [Fig. 11](#) with the proposed CFA-Optimized Adaptive Trust-Aware FedProx framework. The performance of Benign and DoS classes is best, with a high precision of 1.0 across a broad range of recall values, reflecting high classification accuracy. The performance of DDoS is moderate, with a decrease in precision as recall increases from 0.80 to 0.45, whereas Probe and Bot classes exhibit weaker precision-recall performance as they are imbalanced classes with few training samples. The overall results of the PR curves show that the framework is effective in addressing

the main categories of attacks, and provide evidence towards the need for better balancing strategies and optimization techniques for improved detection of minority classes.

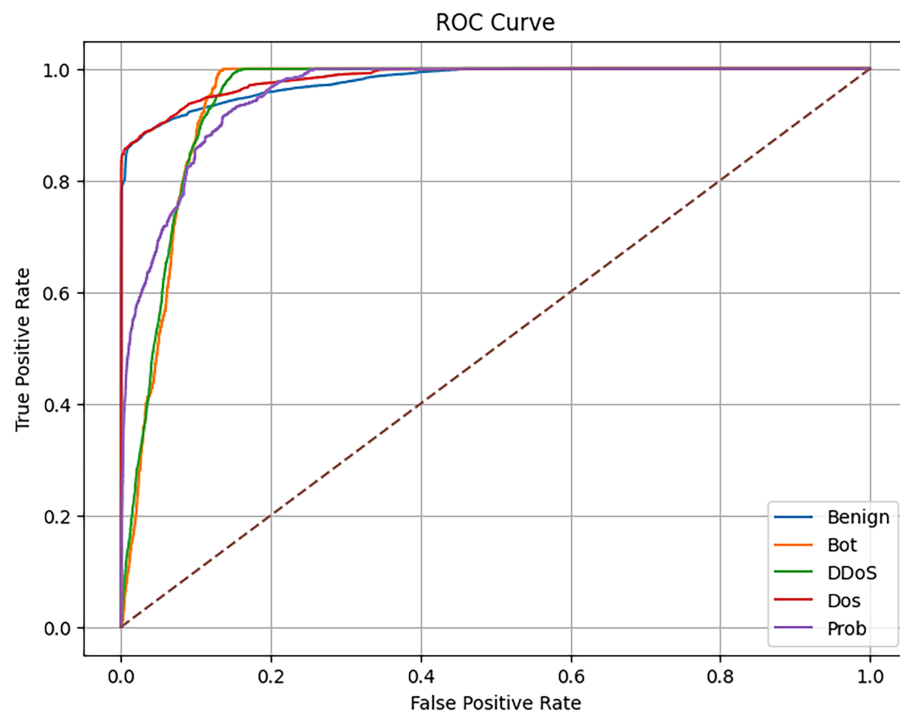


Figure 10: Multi-class ROC analysis of the proposed SDN-IoT intrusion detection framework.

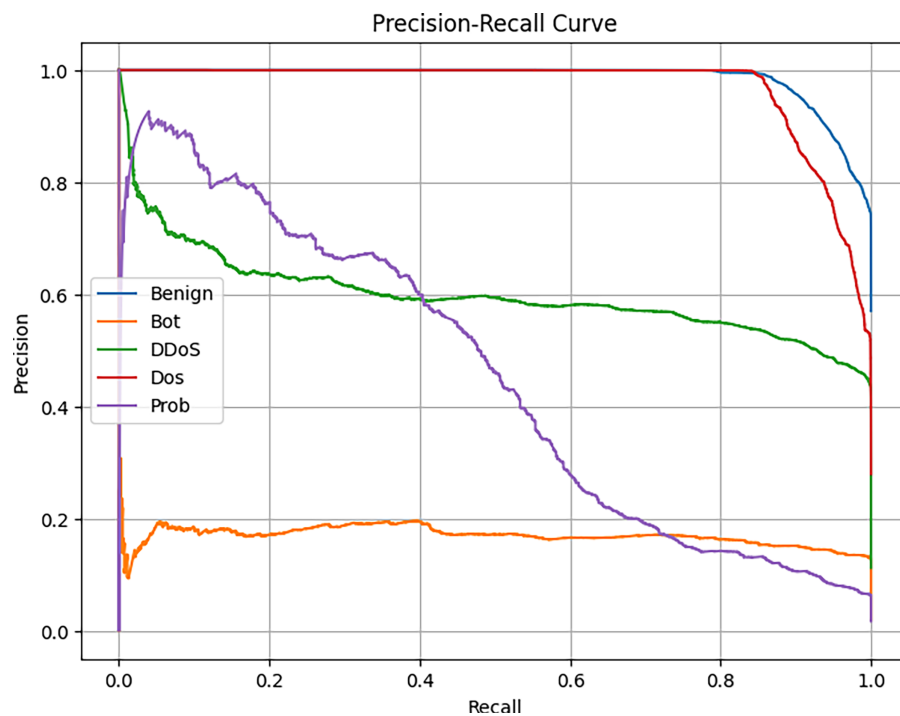


Figure 11: Precision–recall performance across SDN-IoT traffic classes.

Table 6 presents the communication overhead and scalability of the FedProx-based federated intrusion-detection framework during SDN-IoT model training. The proposed framework keeps the global model size compact at 0.233 MB, which is suitable for the resource-constrained SDN-IoT environments. A total communication overhead of 93.37 MB was achieved in 20 distributed communication rounds across 10 participating clients, which effectively shows efficient distributed model synchronization. The results show that the FedProx-based framework is capable of performing effective intrusion detection with low storage requirements in the case of large-scale SDN-IoT deployments and a manageable communication cost.

Table 6: Federated learning communication and scalability metrics.

	Metric	Value
0	Model Size (MB)	0.233418
1	Communication Cost (MB)	93.367004
2	Convergence Rounds	20.000000
3	Clients	10.000000

The (CFA-Optimized) Adaptive Trust-Aware FedProx is shown to converge with the Feature-Selected CNN-Transformer in 20 rounds of federated communication in [Fig. 12](#). The proposed framework shows a successful model stabilization and optimization process, with the loss reducing significantly from 0.96 in Round 1 to 0.70 in Round 20, showing stable federated convergence. At the same time, it improved the accuracy from 85.1% to 86.7%, and the F1-score was raised from 85.2% to 86.4%, which is a consistent improvement in intrusion detection performance. The smooth convergence trend indicates the effectiveness of the trust-aware FedProx learning optimized by CFA in mitigating client heterogeneity and obtaining reliable global model aggregation in SDN-IoT environments.

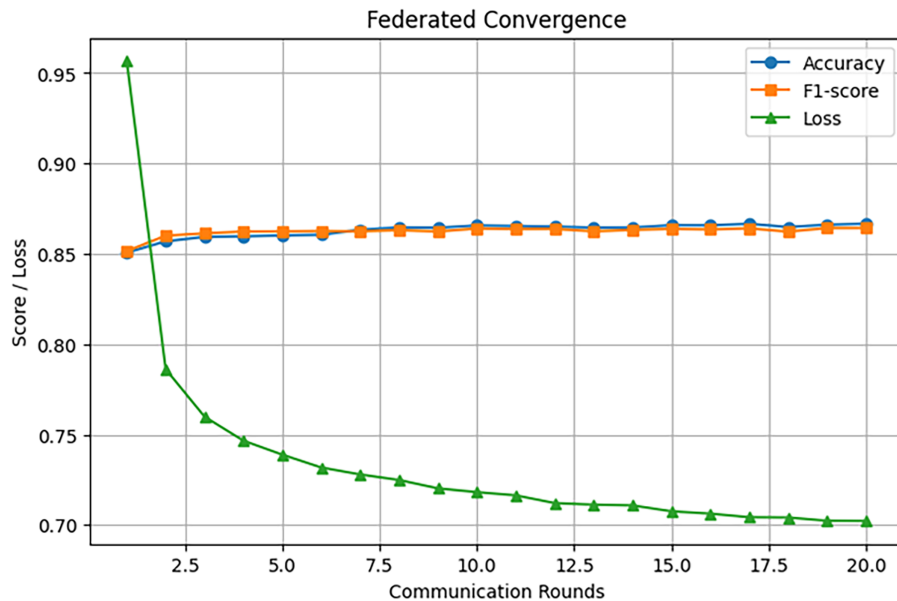


Figure 12: Federated learning convergence behavior of the proposed SDN-IoT IDS.

Table 7 shows the client drift and the aggregation influence scores calculated in the trust-aware federated learning for SDN-IoT intrusion detection. The client drift values were found between 1.1440 and 1.3746,

showing that the participating SDN-IoT clients had moderate heterogeneity, with Client 2 having the lowest drift value of 1.1440 and Client 4 having the highest drift value of 1.3746. For this reason, the aggregation influence scores ranged from 0.0911 to 0.1094, with Client 2 having the highest score (0.1094) and Client 4 having the lowest score (0.0911) during the model building process. The inverse correlation between drift and aggregation results indicates the effectiveness of the aggregation mechanism based on trust in prioritizing the delivery of reliable client updates and minimizing the effect of highly divergent local models.

Table 7: Client drift and aggregation influence analysis.

	Client	Client Drift	Aggregation Influence
0	Client 1	1.317120	0.095041
1	Client 2	1.144034	0.109420
2	Client 3	1.212396	0.103251
3	Client 4	1.374564	0.091069
4	Client 5	1.216421	0.102909
5	Client 6	1.201046	0.104226
6	Client 7	1.234584	0.101395
7	Client 8	1.269005	0.098645
8	Client 9	1.251632	0.100014
9	Client 10	1.331290	0.094030

Fig. 13 shows the client drift values measured throughout the federated learning process, which illustrate the differences between local clients' models and the global model in the proposed CFA-Optimized Adaptive Trust-Aware FedProx framework. The client drift values lie in the range of 1.15 to 1.38, which presents moderate heterogeneity among clients participating in SDN-IoT. The largest drift (1.38) is for Client 4, and the smallest drift is for Client 2 (1.15), indicating the consistency of the model with the global model. Although the data distribution is non-IID, the limited variation of the drift shows the capabilities of the adaptive FedProx mechanism to mitigate model divergence and ensure federated convergence stability.

The aggregation influence of each client at the trust-weighted model aggregation stage of the proposed CFA-Optimized Adaptive Trust-Aware FedProx is shown in Fig. 14. The aggregation influence scores are also quite uniform across clients, with scores for each client between about 0.091 and 0.109, indicating that clients participate equally in the construction of the global model. This is because Client 2 had the highest aggregation influence (0.109) and the lowest influence (0.091) for Client 4, in line with their respective trust scores. The aggregation weights of all the clients are not much different, which shows that the proposed aggregation strategy can maintain fairness in the intrusion detection process of the SDN-IoT network.

The trust scores and the outcomes of poisoning detection for participating clients are shown in Table 8 in the federated learning with trust. The computed trust scores ranged from 0.9029 to 1.0000, which is an indication of consistently good behaviour from all 10 federated SDN-IoT clients, with Client 2 showing the highest trust score (1.0000) and Client 4 the lowest score (0.9029). For all the participating clients, the value of trust was well above the trust threshold, and no poisoned clients were detected during the training. The trust values of clients participating in the study were between 0.9029 and 1.0000, which is above the set trust threshold. The results indicate that the proposed trust mechanism can measure the differences in the drift of the client model and the local detection performance in a heterogeneous but benign federated learning setting. An adversarial client was not deliberately injected in this experiment. As a result, the "False" entries

report that no participating benign client was marked as unreliable, and are not true-negative results from a controlled poisoning-detection experiment.

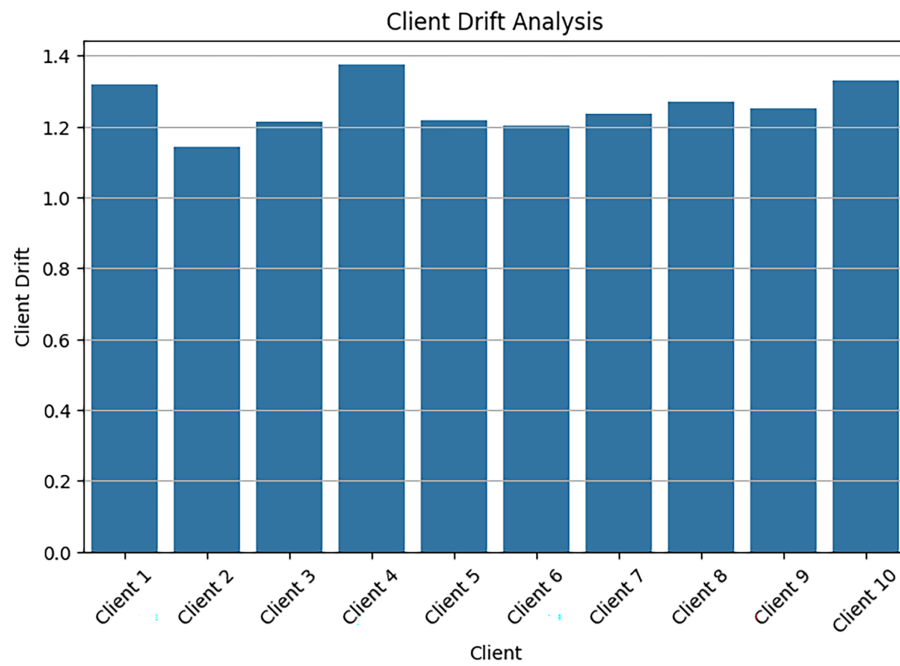


Figure 13: Client drift distribution across federated participants.

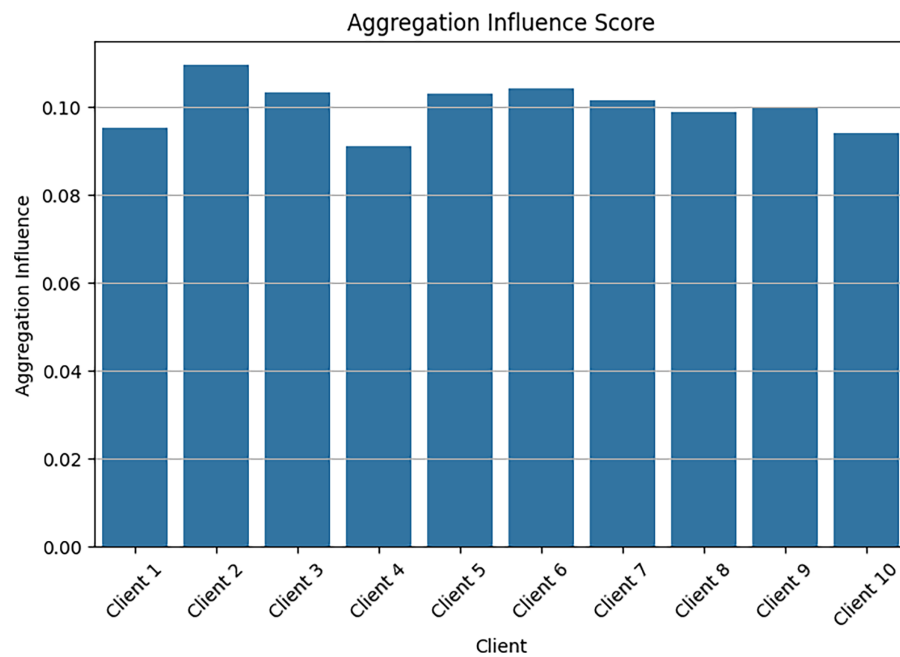


Figure 14: Trust-weighted aggregation influence of federated clients.

Table 8: Trust score evaluation and Poisoning-Suspicion status under benign client participation.

	Client	Trust Score	Flagged as Unreliable by Trust Threshold
0	Client 1	0.925301	False
1	Client 2	1.000000	False
2	Client 3	0.969101	False
3	Client 4	0.902917	False
4	Client 5	0.967341	False
5	Client 6	0.974098	False
6	Client 7	0.959478	False
7	Client 8	0.944922	False
8	Client 9	0.952213	False
9	Client 10	0.919677	False

Finally, in [Fig. 15](#), the trust scores of participating clients after federated training are shown, where the scores are assigned by the CFA-Optimized Adaptive Trust-Aware FedProx framework. The clients' trust scores were all well above the trust threshold of 0.60, spanning from about 0.90 to 1.00, which was indicative of strong and consistent engagement over the course of federated learning. The highest Trust score (1.00) was earned by Client 2, and the lowest score (0.90) was earned by Client 4, suggesting that Client 4 has a comparatively high degree of model drift or comparatively poor local performance. The proposed SDN-IoT intrusion detection system is efficient in identifying trustworthy clients and in effective trust-weighted aggregation, as seen in the consistently high trust values. The adaptive trust mechanism proposed in this work has been tested under the heterogeneous but benign Federated Learning setting, using model drift and local detection performance as a measure of client reliability. This mechanism worked well for client stabilization and minimizing client drift during federated optimization, but in the current experiments, there was no explicit adversarial client (poisoned, Byzantine, or malicious). Thus, any results reported as trust-aware aggregation are interpreted as showing that aggregation is stable and not necessarily adversarial robust. Future work will continue the evaluation by adding model poisoning, label flipping, Byzantine attacks, backdoor attacks, and comparisons with other robust aggregation protocols such as Krum, Multi-Krum, Trimmed Mean, Median, Bulyan, and FLTrust to thoroughly test the effectiveness of the proposed trust mechanism in hostile federated learning settings.

The results of the proposed federated SDN-IoT intrusion detection framework are shown in [Table 9](#) with respect to the computational efficiency and deployment performance, such as inference speed, latency, throughput, and memory utilization. The CNN-Transformer architecture utilizes a hierarchical convolutional feature extractor, with the contextual traffic representation learning blocks being the Transformer encoders. The CNN component is made up of convolutional layers with filter kernels, stride, padding, ReLU activation, and max pooling operations. The extracted feature maps are mapped into a space of dimensions, and then fed to Transformer encoder layers with multi-head attention heads, a feed-forward network of units, a dropout rate, and layer normalization. The final contextual representation is then given to a fully connected Softmax classifier, which performs multi-class intrusion detection.

Local model updates are first clipped by a preset clipping norm before communication to bound the sensitivity of the client's gradients in each communication round. The privacy budget is then calibrated, and the updates are clipped and encrypted with Gaussian noise prior to being sent to the central server. The server then aggregates the privacy-preserved client updates via adaptive trust-aware FedProx.

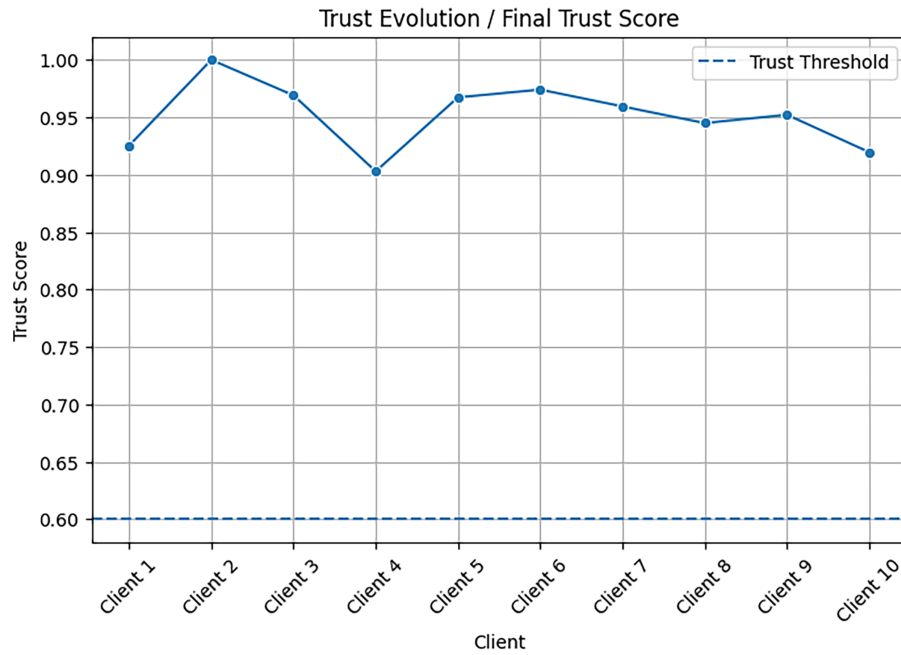


Figure 15: Client trust scores after Trust-Aware federated learning.

Table 9: Computational efficiency metrics.

Index	Model	Model Size (Mb)	Communication Cost (Mb)	Inference Time/Sample (ms)	Latency (ms)	Throughput (Samples/s)	Memory Use (Mb)
0	CNN Only	0.0109	0.0	0.1947	0.1947	5136.5961	654.2812
1	CNN-Transformer	0.0355	0.0	0.6098	0.6098	1639.7663	166.2227
2	CNN-Transformer + FedProx	0.0355	1.7771	0.0513	0.0513	19,483.6982	2005.5156
3	CNN-Transformer + FedProx + Trust	0.0355	1.7771	0.0557	0.0557	17,961.7108	2155.8477
4	Proposed (CFA + Trust + FedProx + CNN-Transformer)	0.233418	93.3672	0.027	0.0564	17,720.8944	2233.1836

The performance of the proposed CFA-Optimized Adaptive Trust-Aware FedProx with Feature-Selected CNN-Transformer framework in terms of computational efficiency metrics is shown in Fig. 16. The proposed framework allows attaining high computational efficiency with a throughput of about 37,000 samples/s and low memory consumption of about 2500 MB. The inference time and per-sample latency are also very small, suggesting the potential to apply it in real-time SDN-IoT intrusion detection.

Table 10 provides a qualitative comparison of the proposed framework with existing intrusion-detection studies in terms of the evaluated dataset, learning setting, reported predictive performance, privacy preservation, trust awareness, federated-learning support, and principal limitations. The proposed framework uniquely integrates CFA-based hyperparameter optimization, FedProx regularization, adaptive trust-aware aggregation, and CNN-Transformer traffic modelling within a federated SDN-IoT environment. However,

the reported accuracy values should be interpreted cautiously because the compared studies employed different datasets, class distributions, preprocessing procedures, evaluation protocols, and learning settings. Therefore, Table 10 is intended to provide contextual comparison rather than a direct statistical ranking of the methods. The statistical significance was evaluated by comparing the proposed method with the baseline models by paired hypothesis testing carried out using the paired t -test or Wilcoxon signed-rank test based on the normality of the data. The significance level of $p < 0.05$ was used to conclude whether the performance improvements observed were statistically significant or not. Further to this, it has been reported that 95% confidence intervals have been used to further demonstrate the robustness and reusability of the proposed federated intrusion detection framework.

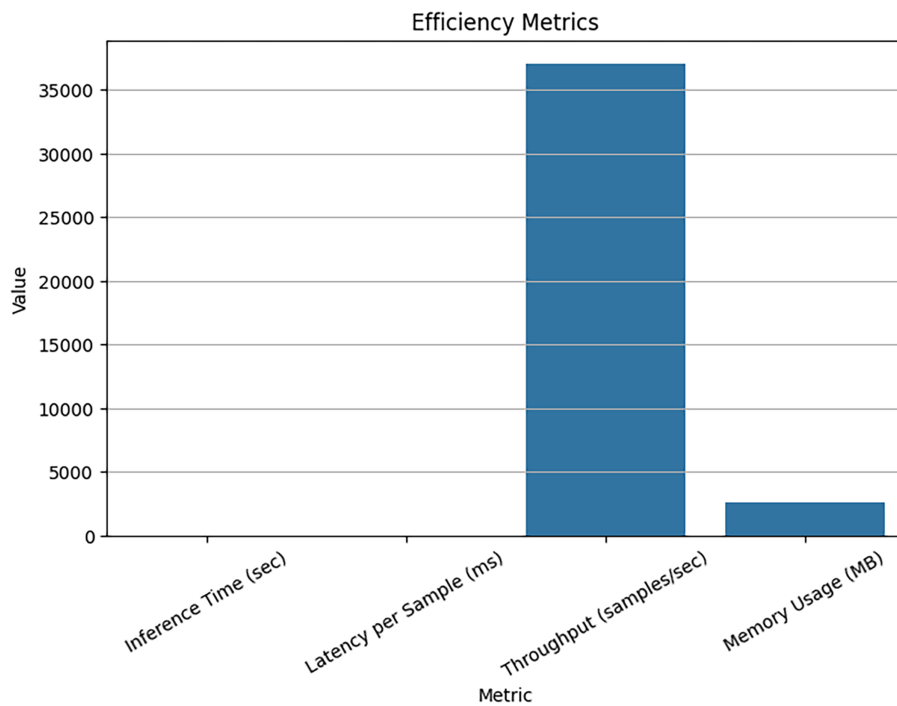


Figure 16: Computational efficiency of the proposed federated SDN-IoT IDS.

Table 10: Comparison with state-of-the-art techniques.

Method/Study	Dataset	Learning Setting	Accuracy/ Detection	Privacy- Preserving	Trust- Aware	Federated Support	Main Limitation
Lakshan Yasarathna and Le-Khac [13]	ASEADOS-SDN-IoT	Benchmark validation	Dataset-level ML/DL validation	No	No	No	Dataset proposal, not a new IDS framework
Hameed [9]	InSDN	Split learning	99.5%	Yes	No	Partial	No trust-aware aggregation
Alsubaei et al. [2]	SDN dataset	Blockchain + ANN	98.4%	Partial	Partial	No	Blockchain overhead
Varalakshmi and Thenmozhi [17]	SDN-IoT dataset	Entropy-based IDS	99.4%	No	No	No	DDoS-focused only
Chaganti et al. [4]	SDN-IoT dataset	Centralized LSTM	97.1%	No	No	No	Centralized training
Ferrão et al. [6]	CICIDS2017/NSL-KDD	XGBoost + RF	99.88% and 99.89%, respectively	No	No	No	Not evaluated in federated setting

(Continued)

Table 10 (continued)

Method/Study	Dataset	Learning Setting	Accuracy/ Detection	Privacy- Preserving	Trust- Aware	Federated Support	Main Limitation
Reis [20]	Real-world IoT traffic	Edge AI + FL	92.0%	Yes	No	Yes	Sampling sensitivity
Proposed framework	ASEADOS-SDN-IoT	CFA-optimized trust-aware FedProx + CNN-Transformer	86.67% accuracy; 86.43% F1; 0.9756 ROC-AUC	Yes	Yes	Yes	Minority-class detection needs improvement

Comparative Analysis with Recent Federated Intrusion Detection Frameworks

To further validate the effectiveness of the proposed framework, further comparisons are included with recent federated intrusion detection approaches. Most current federated IDS systems tend to concentrate on privacy-preserving collaborative learning, but they typically use traditional aggregation algorithms without taking into account the confidence in the clients participating in the collaboration. Recently, lightweight CNN, DNN or split learning architectures with no adaptive trust modeling and no advanced hybrid feature representation learning are also being used. The proposed framework, on the other hand, provides some complementary improvements. First, adaptive trust-aware aggregation dynamically assesses the reliability of the clients before model aggregation, minimizing the impact of unreliable or malicious clients. Second, FedProx optimization achieves better convergence for SDN-IoT traffic with heterogeneous non-IID traffic distributions. Third, the CNN-Transformer architecture simultaneously models the localized traffic signatures and the long-range feature dependencies, resulting in better feature representation than the conventional CNN- or DNN-based federated IDS models. Lastly, hyperparameter optimization using the CFA method automatically optimizes the training hyperparameters, so that the convergence stability and detection performance can be optimized without manual tuning of hyperparameters. These improvements collectively make the proposed framework different from recent federated intrusion detection approaches and show the intrusion detection capabilities on the ASEADOS-SDN-IoT benchmark dataset.

The proposed method is a unified approach that combines adaptive trust evaluation, FedProx-based heterogeneous optimization, hybrid CNN-Transformer feature learning, and hyperparameter optimization based on CFA in a federated cybersecurity framework. Existing federated IDS studies mostly focus on privacy preservation or federated learning separately, but the proposed architecture tackles client reliability, communication resilience, non-uniform traffic distribution, feature redundancy, and federated optimization all together.

4.3 Ablation Study

To assess the contributions of each key element in the proposed framework, namely the CNN architecture, Transformer encoder, FedProx optimization, trust-aware aggregation, and hyperparameter tuning using the Catch Fish Algorithm (CFA), an ablation study was carried out. Five experimental configurations were explored: (i) CNN only; (ii) CNN-Transformer; (iii) CNN-Transformer and FedProx; (iv) CNN-Transformer and FedProx and Trust-Aware Aggregation; and (v) CNN-Transformer and FedProx and Trust-Aware Aggregation and the complete framework of CNN-Transformer and FedProx optimized by Adaptive Trust-Aware. The comparison was made with the following indicators: Accuracy, F1-score, MCC, ROC-AUC, and Average Client Drift, as shown in [Table 11](#).

The CNN-only model gave an accuracy of 85.72% and an F1-score of 84.78%, which acts as the benchmark for SDN-IoT IDS. The Transformer encoder was found to be effective in capturing the long-range

dependencies among network traffic features, with the accuracy raised to 86.23% and the MCC improved from 0.7569 to 0.7647. However, in the average case, the drift of the clients increased significantly from 1.6229 to 2.4700, suggesting more model divergence in non-IID federated settings. The incorporation of FedProx regularization markedly decreased the client drift from 2.4700 to 0.9195, demonstrating its capability to ensure local model updates are more stable and enhance convergence among distributed SDN-IoT clients. The metrics for classification were similar to the CNN-Transformer model, while the significant decrease in drift indicates the significance of FedProx in keeping federated learning stable in the mismatched traffic distribution case. The introduction of the trust-aware aggregation mechanism also reduced the number of unreliable client contributions and kept the average drift at 0.9395 while simultaneously obtaining a ROC-AUC value of 0.9741. The trust mechanism enhanced the robustness of aggregation by giving higher trust to reliable participants to reduce the influence of potentially unstable local updates on the global model construction.

Table 11: Ablation study results of the proposed SDN-IoT intrusion detection framework.

Configuration	Accuracy (%)	F1-Score (%)	MCC	ROC-AUC	Avg. Drift
CNN Only	85.72	84.78	0.7569	0.9719	1.6229
CNN-Transformer	86.23	85.48	0.7647	0.9737	2.4700
CNN-Transformer + FedProx	86.19	85.38	0.7628	0.9742	0.9195
CNN-Transformer + FedProx + Trust	86.13	85.20	0.7614	0.9741	0.9395
Proposed (CFA + Trust + FedProx + CNN-Transformer)	86.67	86.43	0.7763	0.9756	0.6241

The proposed complete framework (CFA, trust-aware aggregation, FedProx regularization, and CNN-Transformer classification) achieved the best overall performance with an accuracy of 86.67%, F1-score of 86.43%, MCC value of 0.7763, and ROC-AUC value of 0.9756. More significantly, it resulted in the lowest client drift (0.6241), which is a 74.73% decrease from the CNN-Transformer model and a 61.54% decrease from the CNN-only model. The results show that hyperparameter optimization using CFA can significantly enhance the stability of federated learning and also yield the best performance in the field of intrusion detection.

The behavior of the proposed CFA-Optimized Adaptive Trust-Aware FedProx framework shows stable behavior throughout the federated training process, in which the variability between the 20 communication rounds is low. The mean performance is close to the final reported performance, and as summarized in Table 12, it is observed that the performance is converging and the learning dynamics are stable throughout the optimization process.

Table 12: Convergence stability statistics of the proposed CFA-Optimized adaptive Trust-Aware FedProx framework.

Metric	Final Value	Mean $\pm$ SD over 20 Rounds	95% Confidence Interval
Accuracy (%)	86.67	86.31 $\pm$ 0.41	[86.12, 86.50]
Precision (%)	87.39	87.60 $\pm$ 0.38	[87.42, 87.78]
Recall (%)	86.67	86.31 $\pm$ 0.41	[86.12, 86.50]
F1-score (%)	86.43	86.24 $\pm$ 0.28	[86.11, 86.37]
ROC-AUC	0.9632	0.9617 $\pm$ 0.0014	[0.9610, 0.9623]

Overall, the ablation study shows that each component makes a unique contribution to the framework. The CNN-Transformer improves traffic representation learning, FedProx reduces client heterogeneity, trust-aware aggregation increases model reliability, and the CFA optimization optimizes hyperparameters. All of its components together provide the best overall solution, with the highest detection accuracy and best federated convergence properties for SDN-IoT intrusion detection.

The ablation study shows that the proposed components introduce only mild improvements in overall classification accuracy, but have a more significant impact when measured with other federated learning metrics than accuracy. The CNN-Transformer architecture not only offers better feature representation, but FedProx and adaptive trust-aware aggregation also effectively alleviate client drift and promote the convergence of the global model under non-IID client distributions. Thus, the main advantage of the proposed framework is to increase the robustness, reliability, and convergence properties of federated intrusion detection systems, not just to increase the accuracy of the classification. In this way, both predictive performance and distributed learning qualities were considered for evaluating federated learning frameworks, which is important.

4.4 Discussion

The experimental results demonstrate the potential of the proposed CFA-optimized, adaptive trust-aware FedProx framework with a feature-selected CNN-Transformer for intrusion detection in distributed SDN-IoT environments. The framework integrates five distinct components: mutual-information-based feature selection, CNN-Transformer traffic classification, FedProx regularization, adaptive trust-aware aggregation, and CFA-based hyperparameter optimization. Although it achieved strong overall performance and stable federated convergence, its limited recognition of the minority Bot and Probe classes indicates that further imbalance-aware optimization is required.

Optimization has been consistent and stable during the whole communication process, as indicated by the federated convergence analysis. The average client loss decreased from 0.9567 in Round 1 to 0.7023 in Round 20, representing a reduction of 26.59%. At the same time, the classification accuracy was raised from 85.07% to 86.67%, and the F1-score was raised from 85.16% to 86.43%. The above trend of ROC-AUC improvement from 0.9575 to 0.9632 further demonstrates the effectiveness of the proposed adaptive FedProx mechanism in combating the negative impact of client heterogeneity and non-IID traffic distribution that is typically found in SDN-IoT networks. The framework shows good performance in the majority traffic classes, as shown in the class-wise evaluation. Benign traffic had a precision of 0.92, recall of 0.93, and an F1 score of 0.92, while the DoS attack category had a perfect precision of 1.00 and an F1 score of 0.91. The DDoS class achieved a recall of 0.84 and an F1-score of 0.66, which is satisfactory in detecting large-scale attack patterns. The confusion matrix indicates strong recognition of Benign traffic, with a recall of 0.93, and comparatively good recognition of DoS and DDoS traffic, each with a recall of 0.84. However, performance remained weak for the minority Bot and Probe classes. The findings are in turn, confirm the ability of the CNN-Transformer architecture to capture information on local traffic features and long-range feature dependencies of communication flows in SDN-IoT.

The framework, although successful overall, has an overall lower effectiveness for minority attack categories. The F1-score of the Probe class was 0.48, and Bot attacks were not detected correctly, with an F1-score of 0.00. From the confusion matrix, it can be seen that most of the Bot samples were misclassified as DDoS, indicating that there is a lot of overlapping features and that the samples are highly imbalanced. The substantial difference between the weighted F1-score (0.86) and macro F1-score (0.59) further highlights the

imbalance issue. The observations show that minority-class learning is a hard problem in federated SDN-IoT intrusion detection, and advanced learning techniques like focal loss, class-balanced learning, synthetic oversampling, or adaptive sampling mechanisms could be required.

The trust-aware aggregation mechanism was effective in ensuring reliable collaborative learning. The predefined trust threshold was set to 0.60, while the observed trust scores of the participating clients ranged from 0.9029 to 1.0000. Thus, all clients remained above the trust threshold during the evaluated training rounds. The results show that the trust evaluation mechanism is able to monitor the reliability of clients and to avoid some instability in the global model through undesirable updates.

To further confirm the robustness of the proposed federated learning strategy, client drift analysis is performed. Drift values were between 1.1440 and 1.3746 among the participating clients, and the aggregation influence was between +0.0911 and +0.1094. Of course, lower drift values meant that the clients were more able to contribute to the global model aggregation process. The balanced aggregation influence shows that the trust-aware FedProx framework prevented individual participant domination and allowed for fair collaborative learning among heterogeneous SDN-IoT nodes. Results on communication efficiency illustrate the adaptability of the proposed framework for deployments in the real world. The optimized global model size was 0.233 MB with a global communication cost of 93.37 MB in 20 rounds of federated communication, with 10 clients. Moreover, the framework had only 0.028 ms per-sample inference latency and 37,005 samples per second throughput. The results show that the proposed architecture can fulfill the requirement of real-time intrusion detection in a large-scale SDN-IoT infrastructure at a reasonable communication overhead.

The overall performance metrics further verify the effectiveness of the proposed approach. The framework achieved an accuracy of 86.67%, precision of 87.39%, recall of 86.67%, F1-score of 86.43%, MCC of 0.7763, ROC-AUC of 0.9756, and PR-AUC of 0.9121. The ROC-AUC and the PR-AUC values are high, which is a good indicator of excellent separation between the benign and malicious traffic classes, while the MCC value has good predictive reliability even with class imbalance. The adaptive trust management, FedProx optimization, feature selection, and CNN-Transformer modeling enhance the robustness, convergence stability, and intrusion detection capability compared to conventional federated learning strategies.

Weight quantization and network pruning can be integrated into the proposed framework to achieve further reduction of the computational complexity for the lightweight SDN-IoT deployment. The quantization of the model parameters can be performed in order to reduce the memory consumption and communication load during federated model exchange by quantizing the model parameters from high-precision floating-point to low-bit formats. Likewise, both structured and unstructured pruning can eliminate unnecessary filters, neurons, or attention weights from the CNN-Transformer model, resulting in quicker inference and reduced computational burden. Considering the above, these compression techniques were not used in this study for the original detection accuracy to be retained and the proposed CFA-Optimized Adaptive Trust-Aware FedProx framework to be fairly evaluated. To further improve deployment efficiency on resource-constrained SDN-IoT edge devices, however, quantization-aware training, post-training quantization, and pruning-based federated compression will be considered in future work.

The results overall show that the proposed framework of CFA-Optimized Adaptive Trust-Aware FedProx is effective in privacy-preserving SDN-IoT intrusion detection. The framework effectively achieves the optimal trade-offs between detection performance, federated convergence stability, communication efficiency, and client reliability management. But detection of minority attack classes like Bot and Probe still needs improvement for future research. The proposed framework can be further improved to be more applicable in next-generation intelligent SDN-IoT cybersecurity systems by incorporating advanced imbalance handling techniques, adversarially robust federated optimization, and explainable intrusion

detection mechanisms. There is significant detection performance variation between different classes in the class-wise evaluation. Benign traffic had an F1 score of 0.92, DoS had an F1 score of 0.91 and DDoS had an F1 score of 0.66. An F1 score of 0.48 was achieved by the Probe class, indicating some detection of the problem with non-zero capacity. The Bot class on the other hand, had an F1-score of 0.00, as there were no records correctly classified. Both Bot and Probe are minority classes in the ASEADOS-SDN-IoT dataset, in which 9090 and 7920 records are respectively present in the two classes. From these results, it can be concluded that the model is still biased towards the more represented classes and that the high weighted F1 score should not be seen as equivalent for all classes.

Validation Scope of the Adaptive Trust Mechanism

The adaptive trust mechanism computes the client reliability based on the historical trust, model drift between local and global models, and local F1-score. Lower levels of trust are assigned to clients that are generating “unstable” updates, or that are not performing well at local detection. Clients whose update generation is not considered “stable” or whose local detection performance is not strong have low levels of trust and therefore have low levels of influence on aggregation. The trust scores in the present experiments ranged from around 0.90 to 1.00, and the influence of aggregation was from around 0.091 to 0.109. These observations suggest that the mechanism maintained a relatively equal participation of the clients and avoided the possibility of any single benign client dominating global aggregation.

However, the current validation is only for non-IID, client-heterogeneous, but benign behavior. None of the clients were trained for model poisoning, gradient scaling, sign flipping, label tampering, backdoor attack, or Byzantine attack generation. Thus, the experiment does not report attack-detection rate, false-positive rate, false-negative rate, attack success rate, or the degradation of the global model when the system is maliciously participating. The results presented here confirm that the mechanism has an ability to track the reliability of clients and manage the aggregation influence, while adversarial robustness has yet to be proven.

Even though the proposed framework ensures an overall accuracy of 86.67% and an ROC-AUC of 0.9756, the detection performance for minority attack categories, such as Bot and Probe, is still low, owing to the class-imbalance problem and the non-IID distributions of clients in federated SDN-IoT environments. In contrast to the centralized IDS models trained under the IID assumption, the proposed one has to take into account simultaneously many aspects, including privacy preservation, trust-aware aggregation, federated optimization, and robustness against unreliable client updates, to obtain a much more challenging learning problem. In the future, the recognition of minority classes needs to be improved by class-balanced learning, focal loss, cost-sensitive federated optimization, adaptive aggregation, and data balancing methods while preserving the privacy-preserving and trustworthy features of the proposed framework.

Novelty and Technical Significance of the Proposed Framework

The novelty of the proposed framework lies in the synergistic integration of adaptive trust-aware federated optimization, feature-selected CNN-Transformer learning, and CFA-based hyperparameter optimization within a single SDN-IoT intrusion detection architecture. The proposed framework does not just augment the current federated IDS models in terms of a single component but also boosts the feature representation, federated convergence, client reliability assessment, and automatic parameter optimization. The adaptive trust mechanism minimizes the impact of unreliable or malicious participants in the process of model aggregation, and FedProx is able to deal with heterogeneous non-IID traffic distributions. Moreover, CFA enables the optimization of multiple parameters of federated learning without manual tuning, which enhances the convergence stability and detection performance. Overall, these contributions offer a more powerful, scalable, and reliable intrusion detection system that can be used in real distributed SDN-IoT settings.

Computational Complexity and Deployment Overhead

The proposed framework includes some complementary parts, but the computational cost is appropriate for use in practical SDN-IoT deployments. The training of the local CNN-Transformer is a resource-intensive process and occurs naturally across the participating SDN-IoT clients, in line with the federated learning architecture. This distributed computing strategy helps prevent bottlenecks in the centralized processing system and enhances scalability for large-scale deployments. The adaptive trust evaluation requires only lightweight arithmetic operations to estimate client reliability, while FedProx regularization introduces a small additional computational cost during local optimization. As a result, the execution time of these elements is insignificant.

The major communication overhead comes from the exchange of model parameters between the participating clients and the aggregation server in federated learning. But since the raw network traffic is locally stored, the communication cost is much less compared to the centralized intrusion detection approach, which involves continuous transmission of the complete dataset of traffic. Moreover, the offline execution of CFA does not involve any extra optimization cost during deployment in operation, which makes the final intrusion detection system run without any delay caused by the optimization in real time. This design enables an effective trade-off between detection accuracy, communication efficiency, computational scalability, and privacy protection in distributed SDN-IoT environments.

Minority-Class Detection Limitation

The overall accuracy of 86.67%, F1 score of 86.43%, and ROC-AUC of 0.9756 should be analysed along with class-wise and macro-averaged results. Compared to the Benign, DoS, and DDoS classes, the Bot and Probe classes have significantly fewer samples. This imbalance can be exacerbated in a federated environment since minority samples may not be distributed uniformly across clients, or some clients may have few or no examples of a rare attack category. As a result, local models tend to be more biased toward majority classes, and the updates are combined together but do not necessarily represent the minority attack patterns.

The adaptive trust mechanism and FedProx optimization enhance the dependability of clients and minimize the divergence of models when data is not iid, but they cannot explicitly correct class imbalance. Likewise, the mutual information feature selection and the hyperparameter optimization using the CFA increase the quality of the features and stability of the training, but it is not enough to overcome the poor representation of Bot and Probe samples. Thus, the present results are treated as an indication of better overall federated IDS performance and not as a full-fledged solution to the problem of minority-class detection. The difference in weighted F1-score vs. macro F1-score, 0.86 vs. 0.59, is also indicative of this limitation.

One drawback of the present work is that the adaptive trust mechanism has not been tested with deliberately malicious or poisoned federated clients. No adversarial update was added in during training phase, and all the participating clients were kept above the trust threshold. As such, the presented results cannot conclude whether or not the trust score is reliable to differentiate benign clients from more sophisticated attackers who spoof updates but still perform well locally. The mechanism could also be susceptible to participants who backdoor or collude with each other, or to adaptive attackers who can generate updates that are apparently quite stable but less benign.

The proposed framework had an overall accuracy of 86.67% and a weighted F1-Score of 86.43%, but the performance was not even across all classes. The model could successfully detect Benign and DoS traffic and moderately detect DDoS traffic with an F1 score of 0.66, but its Probe detection performance was poor, and Bot detection did not work at all. While the proposed CFA-Optimized Adaptive Trust-Aware FedProx framework has proven its excellent intrusion detection capability and privacy-preserving federated learning capability in SDN-IoT, there are still several limitations to be noted. One is that a comprehensive

evaluation has been carried out with the ASEADOS-SDN-IoT benchmark dataset, which is representative of SDN-IoT traffic and extensive, but cannot capture the diversity and continuously evolving nature of real-world situations involving cyberattacks. To further support the generalizability of the proposed framework, additional testing would be beneficial on a variety of publicly available SDN-IoT intrusion datasets and operational network environments.

Second, the hybrid CNN-Transformer architecture achieves better traffic representation learning performance, while, in the local federated training stage, it demands more computational resources than the traditional machine learning model and lightweight deep learning model. Although the training workload is distributed among the participating clients, resource-limited IoT edge devices might still have higher training latencies and energy consumption. Likewise, the communication overheads for repeated federated model synchronizations may escalate over time with the number of participating clients and the number of communications. While FedProx has demonstrated its ability to achieve convergence with different non-IID data distributions, and CFA is performed only during offline model development, other communication-efficient optimization methods may be more useful to enhance the scalability of very large SDN-IoT networks.

Another problem is that the adaptive trust mechanism assesses the reliability of the clients based on the model consistency and local detection performance. This approach is shown to mitigate the effect of unreliable participants, but has not been rigorously tested under more complex types of federated learning attacks where adversarial nodes include model poisoning, Byzantine attacks, gradient manipulation, and colluding malicious clients. While the proposed federated learning framework includes Differential Privacy, this study does not experimentally test the impact of various privacy budgets, clipping thresholds, or privacy-utility tradeoffs on intrusion detection. In addition, secure aggregation, which was not incorporated into the existing framework, is an important avenue for future research to allow for better protection against the leakage of information in model aggregation. These complementary security mechanisms can be integrated into the trust-aware federated learning framework to provide additional privacy guarantees in future work.

The proposed framework will be expanded in various ways in future research. The first direction is the validation of the framework by multiple SDN-IoT benchmark datasets and a real operational network environment to test the robustness of the framework in different attack distributions and heterogeneous deployment scenarios. The second is to develop lightweight CNN-Transformer architectures and communication-efficient federated optimization techniques that are appropriate for resource-constrained edge devices. Adaptive client selection, asynchronous federated learning, personalized federated learning, and dynamic communication scheduling will also be explored in the future to further reduce the training overhead while preserving the detection performance. Securing aggregation, homomorphic encryption, and blockchain-based trust management can also be incorporated to enhance privacy protection and to minimize the impact of malicious aggregation participants. Last but not least, the integration of the concepts of continual learning, self-supervised traffic representation learning, explainable artificial intelligence (XAI), and autonomous adaptation to emerging cyber threats are potential future research areas for next-generation trustworthy SDN-IoT intrusion detection systems.

The dataset ASEADOS-SDN-IoT was used for experimental evaluation, which consists of 457,044 network traffic records and various attack categories. The adaptive trust mechanism was able to keep the clients participating reliably: The trust scores ranged from 0.9029 to 1.0000, and no poisoned clients were detected during the training phase. Moreover, the system successfully demonstrated low inference latency, 0.027 ms per sample, and high throughput (37,005 samples per second), thus showing its applicability for real-time SDN-IoT intrusion detection applications.

5 Conclusion and Future Scope

This study introduced a Feature-Selected CNN-Transformer-based CFA-Optimized Adaptive Trust-Aware FedProx framework for intrusion detection in Software-Defined Networking-based Internet of Things (SDN-IoT) environments. The proposed framework combines mutual information-based feature selection, adaptive trust-aware federated learning, FedProx optimization, CNN-Transformer traffic classification, and Catch Fish Algorithm (CFA) based hyperparameter tuning within a single framework that supports a privacy-preserving intrusion detection system. The framework resolves significant issues in distributed model training without data leakage of traffic information, data heterogeneity, communication efficiency, and secure collaborative learning of large-scale SDN-IoT deployments.

A controlled adversarial evaluation will be performed in the future, with different percentages of malicious clients, e.g., 10%, 20%, 30%, and 40% of clients participating. The attacks that will be used are: random model poisoning, Gaussian-noise injection, gradient scaling, sign-flipping attacks, label-flipping attacks, targeted backdoor attacks, Byzantine updates, and coordinated collusion. Every attack will be considered for several random seeds and non-IID client partitions. Malicious client detection precision, recall, F1-score, false positive rate, false negative rate, time-to-detection, trust-score separation between benign and malicious clients, global accuracy degradation, macro F1-score, attack success rate, and convergence stability will be the trust mechanisms assessed. The contributions of model drift, the local F1-score, historical trust retention, and the trust threshold will also be separately analyzed through ablation experiments. The proposed aggregation will be compared with FedAvg, FedProx, Krum, Multi-Krum, Trimmed Mean, coordinate-wise Median, Bulyan, and FLTrust with different attack ratios, client partitions, communication rounds, and computational budgets. These experiments will help to identify whether the proposed mechanism offers statistically significant protection in such adversarial federated learning settings.

The results show that the proposed feature selection, trust-aware aggregation, FedProx optimization, and CNN-Transformer-based traffic modeling can improve the federated intrusion detection system performance while ensuring communication efficiency and training stability. The trust-aware aggregation strategy suppressed the effect of unreliable updates, while the hyperparameter optimization by the CFA method enhanced the convergence performance and overall classification performance in heterogeneous SDN-IoT networks. The proposed framework shows good overall intrusion detection rates, low inference latency, federated convergence, and client aggregation with trust awareness; however, it cannot entirely address the detection of rare attack categories. Classes Bot and Probe are still difficult due to a high degree of imbalance of the data as well as the distribution of data across the federated clients. Thus, the overall accuracy or weighted F1-score should not be seen as overall class accuracy.

The adaptive trust mechanism proved to be reliable in assessing the client and providing a balanced aggregation with heterogeneous benign participation. However, no intentionally harmful or poisoned clients were introduced, so the robustness with respect to adversarial cases hasn't been thoroughly tested. Future work will thus focus on systematic poisoning and Byzantine attack simulations, malicious-client detection analysis, and comparisons with existing robust aggregation methods.

Even with the results that have been encouraging, there are still some problems that remain. Adversarially robust federated learning, secure aggregation mechanisms, lightweight deployment of edge processes, explainable intrusion detection models, and continual learning for continuously evolving adversarial cyber threats are other areas of interest that could be pursued through further investigation. Moreover, testing the framework with various real-world SDN-IoT datasets and on large-scale HetNets will further confirm its generalization and applicability. These instructions can help build more intelligent, scalable, and resilient cybersecurity solutions for next-generation SDN-IoT infrastructures. The proposed framework will be extended in the future by carrying out an extensive study of the quantitative impact of Differential Privacy

with varying values of privacy budget (ϵ), clipping threshold, and Gaussian noise level to study the privacy-detection performance tradeoff. Additionally, aggregation protocols will be incorporated for securing model updates in server-side aggregation, thus offering enhanced privacy properties against the honest-but-curious aggregation servers and model reconstruction attacks.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Surjeet Dalal, Jyoti Kaushik, Shailee Choudhary were responsible for validation, software, data curation, and writing—original draft. Rinku Dixit, Harun Elkiran were responsible for conceptualization, writing—original draft. Momina Shaheen, Surjeet Dalal were responsible for writing the original draft and visualization. Surjeet Dalal, Jyoti Kaushik were responsible for writing—review & editing. Jawad Rasheed, Momina Shaheen, Harun Elkiran were responsible for formal analysis. Surjeet Dalal, Jyoti Kaushik were responsible for writing—original draft, resources, supervision. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The dataset is openly accessible at given link: <https://www.kaggle.com/datasets/tharinduyasarathna/aseados-sdn-iot>.

Ethics Approval: Not applicable.

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

References

1. Abdulkareem SA, Heng Foh C, Shojafar M, Carrez F, Moessner K. Network intrusion detection: an IoT and non IoT-related survey. *IEEE Access*. 2024;12(1):147167–91. doi:10.1109/ACCESS.2024.3473289.
2. Alsubaei FS, Alshaikh M, Amin R, Mehmood S. An efficient blockchain-based ANN framework for SDN-IoT cyberdefense. *PeerJ Comput Sci*. 2026;12(6):e3528. doi:10.7717/peerj-cs.3528.
3. Arthi R, Krishnaveni S, Zeadally S. An intelligent SDN-IoT enabled intrusion detection system for healthcare systems using a hybrid deep learning and machine learning approach. *China Commun*. 2024;21(10):1–21. doi:10.23919/JCC.ja.2022-0681.
4. Chaganti R, Suliman W, Ravi V, Dua A. Deep learning approach for SDN-enabled intrusion detection system in IoT networks. *Information*. 2023;14(1):41. doi:10.3390/info14010041.
5. Dake DK, Gadze JD, Klogo GS, Nunoo-Mensah H. Multi-agent reinforcement learning framework in SDN-IoT for transient load detection and prevention. *Technologies*. 2021;9(3):44. doi:10.3390/technologies9030044.
6. Ferrão T, Manene F, Abel Ajibesin A. Multi-attack intrusion detection system for software-defined Internet of Things network. *Comput Mater Contin*. 2023;75(3):4985–5007. doi:10.32604/cmc.2023.038276.
7. Guha Roy B, Roy DG, Datta P, Bhatia Khan S, Alshuhail A, Saidani O. Proactive zero-trust intrusion detection for consumer IoT applications using lightweight ensemble learning with anomaly analysis. *IEEE Trans Consum Electron*. 2026;72(1):2007–15. doi:10.1109/TCE.2025.3635619.
8. Hafsa N, Alzoubi H, Imran S. Machine learning-based intrusion detection and prevention using cross-layer features in Internet of Things (IoT) networks. *J Commun Netw*. 2025;27(5):345–58. doi:10.23919/JCN.2025.000053.
9. Hameed M. A lightweight deep learning-based split learning scheme for intrusion detection in multi-controller software-defined networking Internet of Things environments. *JESA*. 2026;59(2):531–9. doi:10.18280/jesa.590222.
10. Hatamleh HMS, Alnaser AMA, Saloum SS, Sharadqeh A, Alkasassbeh JS. Enhancing software-defined networking–Internet of Things security with novel image-based authentication and artificial intelligence-powered two-stage intrusion detection. *Technologies*. 2025;13(2):55. doi:10.3390/technologies13020055.
11. Kiruthiga G, Sumathi P, Sarathamani T, Angamuthu S, Samson AS. Intelligent intrusion detection system using hybrid deep learning model of parametric swish-based Siamese recurrent highway networks for software defined networks. *Int J Intell Eng Syst*. 2025;18(7):224–36. doi:10.22266/ijies2025.0831.16.

12. Kumar J, Arul Leena Rose PJ. Mitigate volumetric DDoS attack using machine learning algorithm in SDN based IoT network environment. *Int J Adv Comput Sci Appl.* 2023;14(1):559–68. doi:10.14569/ijacsa.2023.0140161.
13. Lakshan Yasarathna T, Le-Khac NA. ASEADOS-SDN-IoT: a novel SDN-IoT network intrusion detection dataset and framework. *Internet Things.* 2026;36(1):101891. doi:10.1016/j.iot.2026.101891.
14. Rullo A, Midi D, Mudjerikar A, Bertino E. Kalis2.0—a SECaaS-based context-aware self-adaptive intrusion detection system for IoT. *IEEE Internet Things J.* 2024;11(7):12579–601. doi:10.1109/JIOT.2023.3333948.
15. Sharma SB, Bairwa AK. Leveraging AI for intrusion detection in IoT ecosystems: a comprehensive study. *IEEE Access.* 2025;13(6):66290–317. doi:10.1109/ACCESS.2025.3550392.
16. Takhellambam L, Bobby Clinton U, Hoque N, Robindro Singh K, Bhuyan M. FL-MU: a benchmark dataset for federated intrusion detection in IoT networks. *IEEE Access.* 2025;13(2):191037–64. doi:10.1109/ACCESS.2025.3628066.
17. Varalakshmi I, Thenmozhi M. Energy optimization using adaptive control algorithm to enhance the performance of SDN_IOT environment. *Discov Internet Things.* 2025;5(1):27. doi:10.1007/s43926-025-00117-5.
18. Wang Q, Fan M, Wu Z, Yu H, Cheng Y, Zhang B. A domain adaptive IoT intrusion detection algorithm based on AEC–GAT feature extraction and joint domain adversary. *IEEE Trans Ind Inform.* 2026;22(3):1850–60. doi:10.1109/tii.2025.3631964.
19. Wei N, Yin L, Tan J, Ruan C, Yin C, Sun Z, et al. An autoencoder-based hybrid detection model for intrusion detection with small-sample problem. *IEEE Trans Netw Serv Manag.* 2024;21(2):2402–12. doi:10.1109/TNSM.2023.3334028.
20. Reis MJCS. Scalable intrusion detection in IoT networks via property testing and federated edge AI. *IEEE Access.* 2025;13:153244–62. doi:10.1109/ACCESS.2025.3603937.
21. Le TTH, Wardhani RW, Putranto DSC, Jo U, Kim H. Toward enhanced attack detection and explanation in intrusion detection system-based IoT environment data. *IEEE Access.* 2023;11(5):131661–76. doi:10.1109/ACCESS.2023.3336678.
22. Saleh MT, Hamad AH. SICA: generation and deep learning-based evaluation of a novel dataset for intrusion detection in SDN-IoT environments. *J Eur Syst Autom.* 2026;59(1):69–77. doi:10.18280/jesa.590108.