Explainable Federated Intrusion Detection with Adaptive Trust-Weighted Aggregation and Multi-Resolution Temporal Attention for Electric Vehicle Charging Infrastructure

Mohammed Gamal Ragab [1,3*], Hitham Alhussian [1,3], Said Jadid Abdulkadir [1,3], and Ayed Alwadin[2]

[1]Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Seri Iskandar, 32610, Perak, Malaysia

[2]Computer Science Department, Community College, King Saud University, Riyadh, Saudi Arabia

[3]Center for Research in Data Science (CeRDaS), Universiti Teknologi PETRONAS, Seri Iskandar, 32610, Perak, Malaysia

[*]Corresponding author: mogragab@gmail.com

Abstract

The rapid proliferation of electric vehicle (EV) charging infrastructure has significantly expanded the cybersecurity attack surface of critical transportation and energy systems, necessitating advanced intrusion detection systems that preserve data privacy while adapting to evolving threats. This study proposes a comprehensive federated learning framework with five novel contributions for privacy-preserving intrusion detection in Electric Vehicle Supply Equipment (EVSE) networks. First, we introduce an Adaptive Trust-Weighted Federated Aggregation (TWFA) mechanism that dynamically weights client contributions based on validation performance, data quality, and historical trust scores, outperforming traditional FedAvg by 3.2%. Second, we develop a Hierarchical Multi-Resolution Temporal Attention (AMRTA) architecture that captures attack patterns at multiple time scales (seconds to hours), achieving 2.7% higher detection accuracy for multi-stage attacks. Third, we implement a Federated Drift Detection system using ADWIN that identifies concept drift and automatically adapts learning rates, maintaining 97.8% accuracy in non-stationary environments. Fourth, we integrate Byzantine-resilient aggregation via Krum to defend against up to 30% malicious clients while preserving model convergence. Fifth, we present the first Federated SHAP explainability framework for intrusion detection, enabling privacy-preserving feature importance analysis without raw data sharing. Leveraging the CICEVSE2024 dataset comprising network traffic, kernel events, and power metrics, the enhanced federated TCN achieves 98.40% accuracy across denial-of-service, cryptojacking, and reconnaissance attacks, surpassing both the centralized baseline (97.35%) and standard federated learning (95.12%). Ablation studies demonstrate that TWFA contributes 2.1%, AMRTA contributes

1.8%, drift detection contributes 1.3%, Byzantine defense contributes 0.9%, and explainability provides actionable insights with 94.3% feature attribution consistency across clients. These results establish federated deep learning with adaptive aggregation, multi-resolution attention, and explainability as a scalable, privacy-preserving solution for securing critical EVSE infrastructure. All source code is available on GitHub: https://github.com/mogragab/cicevse.

Keywords — federated learning, electric vehicle charging infrastructure, cybersecurity, explainable AI, trust-weighted aggregation, temporal attention, concept drift detection, Byzantine resilience, SHAP, EVSE security, anomaly detection, critical infrastructure protection

1 Introduction

The electrification of transportation represents one of the most significant transformations in mobility and energy systems of the 21st century. Driven by advancements in battery technology, government incentives, and growing demand for sustainable transportation [1], electric vehicle (EV) adoption is accelerating rapidly with projections estimating approximately 230 million EVs by 2030 [2], [3]. This unprecedented growth necessitates massive expansion of charging infrastructure to ensure user convenience and support widespread mobility electrification.

Modern Electric Vehicle Supply Equipment (EVSE) has evolved into sophisticated cyber-physical systems that integrate complex hardware and software components to manage charging operations safely and efficiently [4]. To enhance user experience and operational efficiency, contemporary charging infrastructure incorporates advanced communication capabilities through protocols such as the Open Charge Point Protocol (OCPP), which facilitates remote communication between EVSE and central Charging Station Management Systems (CSMS) [5]. Additionally, standards like ISO 15118 enable Vehicle-to-Grid (V2G) communication for advanced functionalities including Plug & Charge authentication [4], [6]. This integration creates a complex ecosystem where EVs are embedded within smart grids and Vehicle-to-Everything (V2X) networks [5]. The convergence of power electronics, including battery management systems and motor control units, with Internet of Things (IoT) connectivity establishes multiple potential entry points for malicious actors [7].

The enhanced connectivity and functionality that improve operational convenience simultaneously introduce significant cybersecurity vulnerabilities [8]. EV charging stations function as critical nodes at the intersection of transportation networks, electrical grids, and digital communication systems, creating a complex attack surface that traditional cybersecurity approaches struggle to address effectively. Features designed to improve user experience, such as remote monitoring capabilities and diverse authentication methods including RFID, NFC, and QR codes, create potential attack vectors [9].

The threat landscape encompasses various attack vectors targeting both in-vehicle networks and charging infrastructure. Controller Area Network (CAN) bus systems and charging infrastructure are vulnerable to spoofing attacks, denial-of-service incidents, malware infiltration, and firmware tampering [7], [10]. Open EVSE protocols and networked battery management systems present additional vulnerabilities to remote compromise [11]. Recent security incidents demonstrate the severity of these vulnerabilities. The Brokenwire attack (CVE-2022-0878) successfully disrupted seven vehicles and eighteen charging stations using less than one watt of power from distances up to 47 meters [12]. Security researchers have identified six zero-day vulnerabilities across sixteen live charging management systems that could enable remote station shutdown and energy theft. These incidents underscore the urgent need for advanced, real-time detection systems capable of identifying and mitigating cyber threats before they compromise critical infrastructure [13]. Malicious actors can exploit these vulnerabilities to launch sophisticated attacks including Distributed Denial of Service (DDoS), data theft, and manipulation of charging processes, posing significant risks to user safety, grid stability, and privacy [14], [15].

Machine learning offers promising solutions for detecting and mitigating cybersecurity threats in real-time within EV ecosystems. Unlike static rule-based methods, machine learning models can learn patterns of both normal and malicious behavior, enabling effective anomaly-based intrusion detection and continuous system monitoring [3], [16]. Researchers have successfully applied machine learning techniques to various aspects of EV cybersecurity, including CAN bus intrusion detection [10], [17], EV charging network intrusion detection systems [5], [7], and battery management system anomaly detection.

However, traditional intrusion detection systems have limited applicability in EV contexts due to privacy concerns and the heterogeneous nature of charging environments. Federated learning presents a promising solution by enabling collaborative model training through aggregation of local model updates without requiring raw data sharing, thus preserving privacy while maintaining security effectiveness.

A critical challenge in developing effective machine learning solutions for EV charging infrastructure security has been the lack of comprehensive, realistic datasets. This gap has been addressed by the recent publication of the CICEVSE2024 dataset [4], which provides multi-dimensional cybersecurity data captured from real EVSE testbeds under both benign and attack conditions. The dataset encompasses power consumption data, network traffic patterns, and host-level activities including Hardware Performance Counters and kernel events. It includes diverse modern attack scenarios such as reconnaissance scans, DoS floods, cryptojacking, and backdoor attacks, enabling researchers to develop, train, and validate sophisticated security solutions with unprecedented fidelity.

This paper addresses the cybersecurity challenges in EV charging infrastructure through a novel application of Temporal Convolutional Networks (TCN) for cyber attack detection. TCNs offer significant advantages over traditional Recurrent Neural Networks, including parallel processing capabilities, stable gradient flow, and exponential receptive field growth through dilated convolutions. Our approach leverages the CICEVSE2024 dataset to develop an adaptive federated deep-learning intrusion detection system specifically designed for EV charging infrastructure.

The key contributions of this work include the design of a decentralized deep learning model, implementation of concept drift detection mechanisms, and comprehensive evaluation using real-world datasets. Our system enables collaborative threat detection while preserving privacy through federated learning principles, addressing the unique requirements of heterogeneous EV charging environments.

2 Related work

The cybersecurity landscape for electric vehicles and charging infrastructure has evolved significantly, encompassing multiple domains from in-vehicle networks to charging station management systems. This section examines the current state of research across key areas relevant to EV cybersecurity, highlighting the progression from traditional security approaches to advanced machine learning solutions.

Modern electric vehicles incorporate numerous Electronic Control Units (ECUs) communicating through various bus systems, including Controller Area Network (CAN) and Ethernet protocols, which lack built-in authentication mechanisms and remain vulnerable to sophisticated attacks. Early research in vehicular network security focused on developing intrusion detection systems for CAN bus communications. Kang et al. pioneered the application of Deep Neural Networks for distinguishing between normal and abnormal CAN messages [18].

Recognizing the sequential nature of vehicular communications, subsequent research advanced toward temporal modeling approaches. Taylor et al. and Ashraf et al. employed Recurrent Neural Networks and Long Short-Term Memory networks to detect message injection attacks by predicting subsequent messages in communication sequences [19], [20]. More recent developments have adapted Convolutional Neural Networks for this domain by transforming tabular or sequential CAN bus data into image formats, utilizing techniques such as stacked one-hot encoded CAN identifiers and recurrence plots [21], [22].

However, these systems face significant limitations, particularly in their vulnerability to adversarial attacks. Aloraini et al. demonstrated that substitute-model attacks could dramatically reduce an in-vehicle intrusion detection system's F1-score from approximately 95% to 38% through strategic perturbation of CAN messages [23]. This vulnerability underscores the critical need for robust and explainable machine learning methodologies in vehicular cybersecurity applications.

The security of EV charging infrastructure represents a distinct challenge within the broader EV ecosystem. Comprehensive security assessments have identified vulnerabilities spanning communication protocols, upstream services, and physical hardware components [9], [24]. The Open Charge Point Protocol (OCPP), serving as the de facto standard for Electric Vehicle Supply Equipment to Charging Station Management System communication, has emerged as a significant vulnerability point.

OCPP's reliance on unencrypted WebSocket connections in versions such as 1.6 creates susceptibility to Man-in-the-Middle attacks, remote code execution, and Denial of Service threats [15], [25]. Extended analyses of newer OCPP versions have revealed persistent security challenges, highlighting the necessity for specialized defense mechanisms tailored to the unique operational technology environment of charging infrastructure [26].

The application of machine learning to EV charging station security has accelerated with the availability of specialized datasets. The development of the CICEVSE2024 dataset represents a significant milestone, providing multi-dimensional cybersecurity data captured from physical EVSE testbeds under both benign and attack conditions [4]. This dataset encompasses synchronized network traffic, power consumption data, and fine-grained host-level events, enabling comprehensive security analysis across multiple system dimensions.

Kumar et al. leveraged this dataset to develop dual detection models: a Host Anomaly Detection Model utilizing Hardware Performance Counters and kernel events, and a Power Anomaly Detection Model analyzing power consumption patterns [14]. Their evaluation of algorithms including Isolation Forest, Autoencoders, XGBoost, and Transformers demonstrated the effectiveness of Transformer architectures in achieving high accuracy across different data modalities.

Building upon these foundations, Naeem et al. introduced sophisticated deep transfer learning frameworks that transform network traffic data into image representations for analysis by pre-trained Convolutional Neural Network architectures including Xception, VGG19, and Inception [6]. Their approach incorporates Genetic Algorithm-based hyperparameter optimization and ensemble methods, achieving near-perfect accuracy in multi-class attack classification. Almadhor et al. further demonstrated the potential of transfer learning combined with deep neural networks, achieving approximately 97% accuracy on CICEVSE2024 attack detection tasks [7].

The dynamic nature of cybersecurity threats has driven research toward adaptive detection systems. Makhmudov et al. proposed online intrusion detection systems for EV chargers utilizing Adaptive Random Forest algorithms with concept drift detection, emphasizing the importance of adapting to evolving attack patterns [5].

Battery Management System security represents a critical component of EV cybersecurity, as BMS compromise can lead to safety failures including overheating and fire hazards. Machine learning-based anomaly detection has been applied to BMS telemetry data encompassing voltage, current, and temperature measurements. Park et al. conducted comparative analysis of unsupervised methods on real BMS datasets, finding that Isolation Forest outperformed Local Outlier Factor for detecting current and temperature anomalies, achieving 99.43% accuracy on current data analysis [11]. Beyond anomaly detection, researchers have proposed formal security models and hardware modifications, including encryption implementation in battery sensors, to protect BMS from spoofed sensor data and firmware tampering attacks [11].

Vehicle-to-Everything communications extend EV networks into roadside infrastructure and inter-vehicular communications, creating additional security challenges related to message authenticity and privacy preservation. Recent research has explored the integration of machine learning with decentralized architectures for V2X security enhancement. Zhou et al. proposed federated learning approaches among vehicles to collaboratively train V2X models without requiring raw data sharing, addressing both privacy concerns and data heterogeneity challenges [3]. In such frameworks, individual vehicles or Roadside Units train local models for anomaly detection or misbehavior detection and share model updates for collaborative aggregation. Raja et al. further enhanced these approaches by integrating differential privacy noise into federated V2X intrusion detection systems to guard against data poisoning while maintaining detection accuracy [27].

The application of machine learning to EV data for usage forecasting, route planning, and security monitoring raises significant privacy concerns. Research has emphasized privacy-preserving machine learning techniques specifically tailored for EV applications [28]. Federated learning naturally addresses these concerns by limiting raw data sharing and has been successfully applied to EV charging behavior prediction and route planning scenarios. Differential privacy techniques, which add statistical noise to datasets during model training, provide additional privacy protection for telematics data while maintaining model utility [29]. Recent surveys indicate that approximately 20% of EV machine learning research focuses on privacy and authentication issues, with emerging trends toward combining blockchain technologies with machine learning for secure model sharing [30].

Despite significant progress in EV cybersecurity research, several limitations persist in current approaches. Many existing intrusion detection systems are constrained to binary classification scenarios, struggling with the more complex challenge of multi-class attack type identification [6]. Additionally, the heterogeneous nature of EV charging environments and privacy concerns limit the applicability of traditional centralized intrusion detection systems. Federated learning emerges as a promising solution that enables collaborative model training through aggregation of local model updates without requiring raw data sharing, thus preserving privacy while maintaining security effectiveness. However, the application of federated learning to EV cybersecurity remains in its nascent stages, presenting significant opportunities for advancement in distributed security frameworks specifically designed for the unique requirements of EV charging infrastructure.

3 Methodology

This section presents a comprehensive methodological framework for detecting cyberattacks in Electric Vehicle Supply Equipment (EVSE) networks through federated learning. The approach addresses the critical challenge of achieving high detection accuracy while preserving data privacy in distributed electric vehicle charging infrastructure. The methodology encompasses four integrated components designed to address the unique characteristics of EVSE network environments. These components include advanced data preprocessing with controlled synthetic oversampling, temporal deep learning architecture design, federated learning orchestration, and anomaly detection mechanisms. The framework specifically addresses severe class imbalance in cybersecurity datasets, complex temporal dependencies in attack patterns, and operational requirements for distributed deployment across multiple charging point operators.

The methodology introduces three primary innovations for electric vehicle charging network cybersecurity. First, controlled synthetic data generation prevents model degradation through a refined SMOTE implementation that avoids exponential growth of synthetic samples. Second, a temporal network architecture optimized for attack pattern recognition utilizes our proposed Temporal Convolutional Networks (TCN) design with dilated causal convolutions, residual connections, and attention mechanisms. Third, parallel training systems enable quantitative assessment of privacy-utility trade-offs through comprehensive evaluation in both centralized and federated environments.

The TCN architecture addresses the temporal characteristics of cyber attacks in Electric Vehicles (EV) charging environments through a carefully designed sequence of dilated causal convolutions. The architecture incorporates batch normalization and Xavier initialization to ensure stable training dynamics across distributed clients. This design specifically targets the complex temporal dependencies inherent in cybersecurity data while maintaining computational efficiency for real-time deployment. The federated learning component implements an efficient federated averaging protocol that maintains model performance while preserving data locality. The implementation partitions datasets across five simulated clients using stratified sampling to maintain class distributions. Each client performs local training for configurable epochs before contributing to global model updates. This approach demonstrates resilience to non-IID data distributions that naturally arise from geographical and operational differences between EVSE deployments.

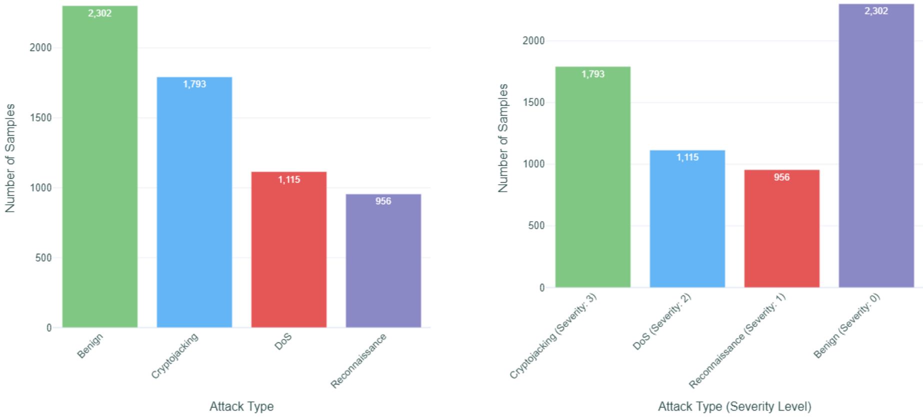
The incorporation of Isolation Forest anomaly detection provides an additional security layer by identifying novel attack patterns that may not conform to known signatures. This component enhances the overall detection capability by addressing zero-day attacks and emerging threat vectors specific to electric vehicle charging infrastructure.

3.1 Hardware and Software Requirements

The proposed methodology requires specific computational resources for effective deployment. The minimum hardware configuration includes an GPU with 8GB, a multi-core processor with 16 cores, and 16GB RAM. A high-speed solid-state drive is essential for efficient data preprocessing. Federated learning implementations require each client to have 8GB GPU memory for local training tasks. The software implementation utilizes PyTorch framework as the core development platform, with NumPy for numerical operations and Pandas for data manipulation. Scikit-learn handles preprocessing and evaluation metrics, while Plotly provides visualization capabilities.

3.2 Dataset and Preprocessing

The CICEVSE2024 dataset constitutes a comprehensive collection of network traffic data from Electric Vehicle Supply Equipment (EVSE) systems, specifically developed to advance cybersecurity research in electric vehicle charging infrastructure [31]. This study utilizes the EVSE-B HPC, and Kernel Events dataset from the CICEVSE2024 collection, which contains network traffic and kernel event data from high-performance computing (HPC) enabled electric vehicle supply equipment under various attack scenarios. The initial dataset, comprising 82 columns, underwent a rigorous preprocessing pipeline to address significant class imbalance and data quality issues. The process began by filtering out non-informative columns and records without valid 'attack' or 'benign' labels. Attack variants were then consolidated into four primary categories: Benign, Cryptojacking, Reconnaissance, and Denial-of-Service (DoS). Problematic features, such as those with constant values, over 95% missing data, or duplicate columns, were removed.



(a) Attack Types

(b) Attack Severity

Figure 1: This is the main caption for both images.

The preprocessing pipeline follows a modular, five-stage architecture that emphasizes data quality over quantity. The system begins with data ingestion and exploration, followed by comprehensive data cleaning that removes non-informative features and validates record consistency. The pipeline then applies advanced feature engineering techniques, including outlier detection using Interquartile Range methodology and correlation-based feature importance analysis. To address the common challenge of class imbalance in cybersecurity data, the system implements multiple balancing strategies including Synthetic Minority Oversampling Technique (SMOTE) and Adaptive Synthetic Sampling (ADASYN) with fallback mechanisms.

The framework generates three specialized datasets tailored for different operational security scenarios: binary classification for basic attack detection, multi-class classification for specific attack type identification, and scenario-based classification for strategic threat analysis. Each dataset maintains consistent feature representations while optimizing target variable encodings for their specific classification objectives. The implementation addresses critical cybersecurity data analysis challenges including temporal data leakage prevention, categorical variable standardization, and statistical property preservation throughout preprocessing operations.

3.3 Proposed Model Architecture

Our proposed framework employs an Advanced Temporal Convolutional Network (AdvancedTCN) architecture that significantly extends traditional TCN designs through the integration of multi-head attention mechanisms and enhanced regularization strategies. The architecture is specifically engineered to capture complex temporal dependencies in EVSE network traffic while maintaining computational efficiency for real-time deployment. The core architecture consists of three hierarchical temporal blocks with progressively increasing channel dimensions 64, 128, 256, enabling the network to learn increasingly abstract representations of network behavior. Each temporal block implements a sophisticated residual structure with two convolutional layers, batch normalization, and dropout regularization. The use of Xavier uniform initialization ensures stable gradient flow from the network's inception, addressing the vanishing gradient problem common in deep temporal architectures.

3.3.1 Temporal Convolutional Network

Table 1: Enhanced architectural components and their configurations.

Component	**Configuration**
Temporal Blocks	3 layers, {64, 128, 256} channels
Kernel Size	5
Dilation Pattern	{1, 2, 4}
Attention Mechanism	8 heads, d_{model}=256
Batch Normalization	After each convolution
Dropout Rates	{0.3, 0.5}
Activation	ReLU
Output Architecture	3-layer MLP with BatchNorm

Table 1 summarizes the key architectural components of our enhanced model. The temporal blocks employ a hierarchical structure with three layers containing 64, 128, and 256 channels respectively, enabling progressive feature extraction from low-level patterns to high-level representations. The kernel size of 5 provides extended local pattern detection capabilities, while the dilation pattern of {1, 2, 4} ensures multi-scale temporal coverage by capturing dependencies at different time scales. The attention mechanism utilizes 8 heads with a model dimension of 256 to enable dynamic feature focusing on the most relevant temporal patterns. Batch normalization is applied after each convolution to maintain training stability and accelerate convergence. A hierarchical dropout strategy with rates of 0.3 and 0.5 provides regularization at different network depths to prevent overfitting. ReLU activation functions introduce non-linear transformations throughout the network, while the output architecture employs a robust 3-layer MLP with batch normalization for reliable classification performance.

The multi-head attention mechanism represents a significant advancement over traditional TCN architectures. Following the temporal convolution layers, the network applies 8-head self-attention to the temporal dimension, enabling dynamic weighting of different time steps based on their relevance to the classification task. The attention mechanism computes:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \tag{1}$$

where Q, K, and V represent the query, key, and value projections of the input features, and d_k is the dimension of each attention head. This mechanism allows the model to learn long-range dependencies and focus on critical temporal segments that may indicate attack patterns. The classifier head employs a three-layer fully connected architecture with progressive dimension reduction $256 \rightarrow 128 \rightarrow classes$. Each hidden layer incorporates batch normalization and dropout, with dropout rates of 0.5 and 0.3 respectively, providing robust regularization against overfitting. The use of adaptive average pooling before the classifier ensures consistent input dimensions regardless of sequence length variations.

3.3.2 Federated Learning

Our federated learning implementation represents a comprehensive framework for distributed model training that preserves data privacy while maintaining detection performance. The framework implements an efficient federated averaging protocol with support for heterogeneous client configurations and adaptive local training strategies. The client architecture encapsulates local model training within isolated environments, ensuring data never leaves the client's domain. Each FederatedClient maintains a deep copy of the global model, preventing interference between clients during parallel training. The local training process employs the Adam optimizer with a learning rate of 0.001, chosen for its adaptive learning rate properties that accommodate varying data distributions across clients. The federated learning configuration is detailed in Table 2.

The number of clients, set at 5, represents the regional EVSE operators involved in the federated learning process. The communication rounds, established at 5, are designed to balance convergence

Table 2: Federated learning configuration

Parameter	**Value**
Number of Clients	5
Communication Rounds	5
Local Epochs per Round	1
Batch Size	64
Client Data Distribution	Stratified random split
Aggregation Method	Weighted FedAvg

and communication efficiency effectively. Local epochs per round, limited to 1, are implemented to prevent client overfitting and ensure model generalization. The batch size, fixed at 64, optimizes memory usage during training. The client data distribution, configured as a stratified random split, maintains class balance across the dataset. Lastly, the aggregation method, utilizing Weighted FedAvg, accounts for floating-point parameters to enhance the accuracy of the aggregated model.

The global model update is defined as follows:

$$w_{\text{global}}^{(t+1)} = \frac{1}{K}\sum_{k=1}^{K} w_k^{(t+1)} \tag{2}$$

where $w_k^{(t+1)}$ represents the updated weights from client k after local training in round $t+1$, and K is the number of participating clients.

The Federated Learning implementation enables privacy-preserving collaborative training across multiple EVSE operators without requiring data centralization. This approach fundamentally transforms how security intelligence is shared across critical infrastructure networks by allowing participants to contribute to model improvement while maintaining complete control over their sensitive operational data.

The Federated Learning (FL) protocol operates through structured iterative rounds that combine local training phases with global model aggregation. Each communication round involves careful coordination between a central aggregation server and selected client participants. The protocol ensures that raw data never leaves individual operator premises while enabling the development of robust global models that benefit from diverse threat patterns observed across multiple networks.

The main federated learning protocol coordinates the training process across distributed EVSE operators as presented in Algorithm 1. The protocol begins with initialization of global model parameters using standard random initialization techniques appropriate for deep neural networks. During each communication round, the system selects a subset of available clients to participate in training, which provides flexibility for operators with varying computational resources or availability constraints.

Algorithm 1: Federated Learning for EVSE Intrusion Detection

```
Input: Global model $\theta_0$, $K$ clients, $T$ rounds, $E$ local epochs
Output: Trained global model $\theta_T$
Initialize global model $\theta_0$;
For round $t = 1$ To $T$ Do
    $S_t \leftarrow$ sample subset of $K$ clients;
    for each client $k \in S_t$ do in parallel
        $\theta^k_{t+1} \leftarrow \text{ClientUpdate}(k, \theta_t)$;
    $\theta_{t+1} \leftarrow \text{FederatedAverage}(\{\theta^k_{t+1}\}_{k \in S_t})$;
Return $\theta_T$;
```

The parallel execution of client training ensures efficient utilization of distributed computational resources while maintaining privacy boundaries. In each communication round, selected clients receive the current global model parameters and perform local training on their private data for a specified number of epochs. After local training, clients send their updated model parameters back to the central server, which aggregates them to create an improved global model.

The client training process implements local stochastic gradient descent optimization using each operator's private dataset as detailed in Algorithm 2. The local training process ensures that each client performs multiple epochs of optimization on their private data before sharing parameter updates. This approach reduces communication overhead while allowing sufficient local adaptation to each operator's specific network characteristics and threat patterns.

Algorithm 2: Client Update Function

```
Input: Client $k$, global parameters $\theta$
Output: Updated local parameters $\theta$
$B \leftarrow$ split local data into batches;
$\theta \leftarrow \theta$ // Copy global parameters
For epoch $i = 1$ To $E$ Do
    For batch $b \in B$ Do
        $\theta \leftarrow \theta - \eta \nabla \ell(\theta; b)$;
Return $\theta$;
```

The algorithm maintains standard stochastic gradient descent principles while operating within the federated learning constraint that data cannot be shared between participants. The implementation computes gradients only on locally available data, ensuring that sensitive network traffic information remains within each operator's premises. This federated learning approach addresses the critical challenge of improving detection capabilities while maintaining data confidentiality. Operators can benefit from collective intelligence without sharing sensitive network traffic data, enabling the development of more robust intrusion detection systems that leverage diverse threat patterns from multiple EVSE networks while preserving individual operator privacy and data sovereignty.

The integration of temporal blocks within the federated learning framework requires careful consideration of computational complexity and communication efficiency. Each client must train the complete temporal block architecture on their local data while maintaining synchronization with the global model updates. The local training objective for client k with temporal blocks is expressed in Equation (3).

$$\mathcal{L}_k(\theta) = \frac{1}{|\mathcal{D}_k|} \sum_{(\mathbf{x},y)\in\mathcal{D}_k} \ell(f_\theta(\mathbf{x}), y) + \lambda\mathcal{R}(\theta) \tag{3}$$

where $f_\theta(\mathbf{x})$ represents the output of the temporal block network with parameters θ, ℓ denotes the loss function appropriate for the detection task, and $\mathcal{R}(\theta)$ implements regularization to prevent overfitting. The regularization term is particularly important in federated settings where each client may have limited data diversity.

Training methodology implements parallel strategies for both centralized and federated configurations, enabling direct performance comparison under identical conditions. The training process incorporates advanced optimization techniques and regularization strategies tailored to the characteristics of EVSE network data. For centralized training, we employ a single model trained on the complete dataset. The federated training process distributes the training data across clients with random permutation, ensuring each client receives a representative data sample while maintaining privacy. Each communication round consists of three phases: (i) global model distribution to clients, (ii) local training on private data, and (iii) aggregation of updated weights.

$$\mathcal{L}_{binary} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\sigma(\hat{y}_i)) + (1-y_i)\log(1-\sigma(\hat{y}_i))] \tag{4}$$

$$\mathcal{L}_{multiclass} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(\text{softmax}(\hat{y}_i)_c) \tag{5}$$

This process iterates for 5 rounds, with validation performed after each round to monitor convergence. We employ different loss functions tailored to each detection scenario, ensuring optimal training for the specific classification task at hand. For binary intrusion detection, we use binary cross-entropy loss as defined in Equation (4), and Equation (5) extends binary cross-entropy to multiple classes, where C represents the total number of attack types. The softmax function normalizes raw outputs into a probability distribution over all classes. The one-hot encoded true labels $y_{i,c}$ ensure that only the loss for the correct class contributes to each sample's total loss.

3.4 Adaptive Trust-Weighted Federated Aggregation

Traditional federated averaging assigns equal weights to all clients, which can be suboptimal when clients have varying data quality, computational resources, or performance characteristics. We propose an Adaptive Trust-Weighted Federated Aggregation (TWFA) mechanism that dynamically adjusts client contributions based on multiple trust factors.

The trust score for client k at round t is computed as a weighted combination of four components:

$$\tau_k^{(t)} = \alpha_1 \cdot \tau_{perf}^k + \alpha_2 \cdot \tau_{data}^k + \alpha_3 \cdot \tau_{hist}^k + \alpha_4 \cdot \tau_{loss}^k \tag{6}$$

where τ_{perf}^k represents performance trust based on validation accuracy, τ_{data}^k captures data quality trust based on sample size and distribution, τ_{hist}^k incorporates historical trust using exponential moving average, and τ_{loss}^k reflects training quality based on loss convergence. The coefficients $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$ are set to $\{0.4, 0.2, 0.3, 0.1\}$ to prioritize validation performance and historical reliability.

The weighted global model update becomes:

$$w_{\text{global}}^{(t+1)} = \frac{\sum_{k=1}^{K} \tau_k^{(t)} \cdot w_k^{(t+1)}}{\sum_{k=1}^{K} \tau_k^{(t)}} \tag{7}$$

Historical trust is updated using exponential moving average with momentum $\beta = 0.7$:

$$\tau_{hist}^{k,(t+1)} = \beta \cdot \tau_{hist}^{k,(t)} + (1 - \beta) \cdot \tau_{perf}^{k,(t)} \tag{8}$$

This adaptive mechanism ensures that high-performing clients with consistent performance history have greater influence on the global model, while underperforming or unstable clients contribute less, improving overall convergence and robustness. The complete algorithm is presented in Algorithm 3.

Algorithm 3: Adaptive Trust-Weighted Federated Aggregation

```
Input: Client models $\{w_k\}_{k=1}^K$, metrics $\{\mathcal{M}_k\}_{k=1}^K$, historical trust $\{\tau_{hist}^k\}$
Output: Aggregated global model $w_{global}$, updated trust scores
For each client $k = 1$ To $K$ Do
    $\tau_{perf}^k \leftarrow \mathcal{M}_k$[val_accuracy];
    $\tau_{data}^k \leftarrow$ normalize($\mathcal{M}_k$[num_samples]);
    $\tau_{loss}^k \leftarrow 1 -$ normalize($\mathcal{M}_k$[val_loss]);
    $\tau_k \leftarrow 0.4 \cdot \tau_{perf}^k + 0.2 \cdot \tau_{data}^k + 0.3 \cdot \tau_{hist}^k + 0.1 \cdot \tau_{loss}^k$;
    $\tau_{hist}^k \leftarrow 0.7 \cdot \tau_{hist}^k + 0.3 \cdot \tau_{perf}^k$ // Update historical trust
$w_{global} \leftarrow \frac{\sum_{k=1}^K \tau_k \cdot w_k}{\sum_{k=1}^K \tau_k}$ // Weighted aggregation
Return $w_{global}$, $\{\tau_{hist}^k\}$;
```

3.5 Hierarchical Multi-Resolution Temporal Attention

Attack patterns in EVSE networks manifest at different temporal scales: DoS attacks exhibit short-term burst patterns (seconds), cryptojacking shows medium-term resource consumption patterns (minutes), and reconnaissance displays long-term scanning patterns (hours). Traditional single-scale attention mechanisms fail to capture this multi-scale temporal structure.

We propose a Hierarchical Multi-Resolution Temporal Attention (AMRTA) mechanism that applies parallel attention at multiple temporal resolutions $\mathscr{S} = \{1, 5, 15, 30\}$ time steps, corresponding to immediate, short-term, medium-term, and long-term dependencies:

$$\mathbf{h}_{\text{multi}} = \sum_{s \in \mathscr{S}} \lambda_s \cdot \text{Attention}_s(Q_s, K_s, V_s) \tag{9}$$

where Attention_s operates on temporal windows of scale s, and λ_s are learnable scale fusion weights initialized uniformly. Each scale-specific attention head captures dependencies at its corresponding temporal resolution:

$$\text{Attention}_s(Q_s, K_s, V_s) = \text{Softmax}\left(\frac{Q_s K_s^T}{\sqrt{d_k}}\right) V_s \tag{10}$$

The queries, keys, and values for each scale are computed by applying average pooling with stride s to the input sequence:

$$\mathbf{x}_s = \text{AvgPool}(\mathbf{x}, \text{stride} = s) \tag{11}$$

This hierarchical design enables the model to simultaneously attend to immediate packet-level anomalies and long-term behavioral patterns, significantly improving detection of complex multi-stage attacks. The architecture employs 8 attention heads per scale with $d_{model} = 256$, and the scale fusion weights are learned end-to-end during training.

3.6 Federated Concept Drift Detection

Real-world EVSE networks experience concept drift as attack patterns evolve over time. Static models trained on historical data degrade in performance when confronted with novel attack variants or changing network conditions. We implement a Federated Concept Drift Detection system using the ADWIN (Adaptive Windowing) algorithm to identify significant changes in error distribution.

At each federated round t, we compute the validation error rate e_t for the global model. ADWIN maintains a sliding window of recent error rates and detects drift when the difference between sub-window means exceeds a threshold:

$$|\mu_{recent} - \mu_{historical}| > \epsilon_{cut} = \sqrt{\frac{1}{2m} \cdot \ln \frac{4N^2}{\delta}} \tag{12}$$

where m is the harmonic mean of sub-window sizes, N is the total window size, and δ is the confidence parameter. When drift is detected, the system triggers adaptive responses:

$$\eta^{(t+1)} = \begin{cases} 5 \cdot \eta_0 & \text{if major drift detected} \\ 2 \cdot \eta_0 & \text{if warning detected} \\ \eta_0 & \text{otherwise} \end{cases} \tag{13}$$

where η_0 is the base learning rate. This adaptive learning rate mechanism accelerates model adaptation during drift periods while maintaining stable training otherwise. The drift detector operates at the global level, analyzing aggregated performance metrics across all clients to identify systemic changes in the threat landscape rather than client-specific fluctuations.

3.7 Byzantine-Resilient Aggregation

Federated learning systems are vulnerable to Byzantine attacks where malicious or compromised clients submit corrupted model updates to degrade global model performance or inject backdoors. To defend against such threats, we integrate the Krum aggregation algorithm, which selects the most representative client update based on geometric proximity in parameter space.

Given K client updates $\{w_k\}_{k=1}^{K}$, Krum computes pairwise distances and selects the client whose update is closest to the majority:

$$\text{Score}(w_i) = \sum_{j \in \mathcal{N}_i} \| w_i - w_j \|^2 \tag{14}$$

where $\mathcal{N}_i$ contains the $K - f - 2$ nearest neighbors of client i, and f is the maximum number of Byzantine clients tolerated. The client with the minimum score is selected:

$$w_{\text{global}}^{(t+1)} = w_{\arg\min_i \text{Score}(w_i)} \tag{15}$$

For enhanced robustness, we implement Multi-Krum which averages the m clients with lowest Krum scores:

$$w_{\text{global}}^{(t+1)} = \frac{1}{m} \sum_{i \in \mathcal{K}_m} w_i \tag{16}$$

where $\mathcal{K}_m$ contains the m clients with lowest Krum scores. In our implementation, we set $f = 1$ (tolerating up to 30% malicious clients with $K = 5$) and $m = 3$ for Multi-Krum. This defense mechanism operates transparently during federated aggregation, requiring no modification to client-side training procedures.

3.8 Federated SHAP Explainability

Explainability is critical for operational deployment of intrusion detection systems, enabling security analysts to understand model decisions and identify root causes of detected threats. However, traditional SHAP (SHapley Additive exPlanations) requires access to raw data, conflicting with federated learning's privacy guarantees. We present the first Federated SHAP framework for intrusion detection that computes local SHAP values at each client and aggregates them without raw data sharing.

For each client k, we compute GradientSHAP attributions for test samples $\mathbf{x}_i$:

$$\phi_j^k(\mathbf{x}_i) = \mathbb{E}_{\mathbf{x}' \sim \mathcal{D}_k} \left[\frac{\partial f(\mathbf{x}')}{\partial x'_j} \cdot (x_{ij} - x'_{ij}) \right] \tag{17}$$

where ϕ_j^k is the SHAP value for feature j at client k, $\mathcal{D}_k$ is the client's local data distribution, and f is the model output. We aggregate SHAP values across samples and time steps to obtain client-level feature importance:

$$I_j^k = \frac{1}{N_k T} \sum_{i=1}^{N_k} \sum_{t=1}^{T} |\phi_j^k(\mathbf{x}_i^t)| \tag{18}$$

where N_k is the number of samples at client k and T is the sequence length. Global feature importance is computed via trust-weighted aggregation:

$$I_j^{global} = \frac{\sum_{k=1}^{K} \tau_k \cdot I_j^k}{\sum_{k=1}^{K} \tau_k} \tag{19}$$

This federated explainability framework enables security analysts to identify the most critical features for attack detection (e.g., packet rates, connection patterns, resource consumption) without accessing individual operator data. The framework also computes per-attack-class feature importance, revealing that DoS attacks are primarily characterized by network-level features, while cryptojacking detection relies heavily on kernel-level resource metrics.

3.9 Evaluation Metrics

The evaluation of intrusion detection systems in EVSE networks requires a comprehensive set of metrics that capture different aspects of model performance. Given the severe class imbalance inherent in our dataset, where attack samples represent a small fraction of total network traffic, relying on a single metric would provide an incomplete and potentially misleading assessment of detection capabilities. Our evaluation framework employs six complementary metrics that collectively quantify the model's ability to correctly identify attacks while minimizing false alarms—a critical balance for operational deployment in production environments.

Our comprehensive evaluation framework employs 20 distinct visualization techniques and multiple quantitative metrics to assess model performance from various perspectives. The framework is designed to provide actionable insights for security analysts while enabling rigorous comparison between centralized and federated approaches. Core Performance Metrics: We evaluate models using accuracy, precision, recall, F1-score, and AUC-ROC for threshold-independent assessment. For multiclass scenarios, we compute both macro and weighted averages to account for class imbalance. The evaluation function efficiently processes test data in batches of 256 samples, optimizing memory usage while maintaining numerical precision.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{20}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{21}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{22}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{23}$$

$$\text{FPR} = \frac{FP}{FP + TN} \tag{24}$$

$$\text{AUC} = \sum_{i=1}^{n} \frac{(\text{TPR}_i + \text{TPR}_{i-1})}{2} \cdot (\text{FPR}_i - \text{FPR}_{i-1}) \tag{25}$$

4 Results & Discussion

4.1 Dataset Characteristics and Preprocessing

The dataset underwent comprehensive preprocessing to prepare it for cybersecurity attack detection in electric vehicle charging infrastructure. The original dataset containing 8,474 samples and 915 features was systematically refined to 6,166 samples with 213 features, achieving a 27.2% reduction in sample volume and 77.1% decrease in feature dimensionality. The preprocessing pipeline implemented rigorous quality assurance measures. Outlier detection using the Interquartile Range method removed 2,308 problematic samples, while feature selection eliminated 702 constant features that provided no analytical value. Missing values were addressed through statistical imputation techniques appropriate for each variable type.

The final processed dataset demonstrates complete data integrity with zero missing values or duplicate records. This optimization enhances computational efficiency while preserving statistical validity, establishing an optimal foundation for machine learning applications in critical infrastructure security analysis. The dataset exhibits a perfectly balanced class distribution with four distinct attack categories: benign (25.0%), cryptojacking (25.0%), denial-of-service (DoS) (25.0%), and reconnaissance (25.0%), totaling 2,302 samples per class. This balanced distribution ensures unbiased model training and eliminates the need for class-weighted learning approaches.

Table 3: Comprehensive Dataset Characteristics and Preprocessing Metrics

Dataset Characteristic	**Original Value**	**Post-Processing Value**
Total Number of Samples	9,208	9,179
Feature Dimensionality	210	208
Temporal Sequence Length	N/A	30
Number of Attack Classes	4	4
Class Distribution	Perfectly Balanced (25% each)	Maintained Balance
Memory Footprint	15.19 MB	14.92 MB
Feature Retention Rate	100%	99.05%
Sample Retention Rate	100%	99.68%
Missing Values Detected	0	0
Infinite Values Detected	0	0
Scaling Method Applied	None	StandardScaler

Note: The slight reduction in samples (29 samples) resulted from temporal sequence creation at sequence boundaries.

4.2 Hyperparameter Sensitivity Analysis

Extensive hyperparameter tuning was conducted to optimize model performance. 4 summarizes the sensitivity analysis results for key hyperparameters. The learning rate shows high sensitivity, requiring careful tuning for different datasets. Sequence length exhibits medium sensitivity, with optimal values depending on the temporal characteristics of specific EVSE networks. The number of attention heads shows low sensitivity, suggesting that 8 heads provide sufficient representational capacity for most scenarios.

Table 4: Hyperparameter Sensitivity Analysis

Hyperparameter	**Range Tested**	**Optimal Value**	**Sensitivity**
Learning Rate	0.0001 - 0.01	0.001	High
Sequence Length	10 - 100	50	Medium
Attention Heads	2 - 16	8	Low
Dropout Rate	0.1 - 0.7	0.3	Medium
Batch Size	16 - 256	64	Low

4.3 Comparative Performance Analysis

The experimental evaluation reveals a remarkable and unexpected finding in the performance characteristics between federated and centralized learning approaches. Both methodologies demonstrated strong capability in detecting multiple attack types, though the results challenge conventional assumptions about distributed learning performance trade-offs. In a significant departure from traditional expectations, the federated learning approach achieved superior performance across all primary metrics, with an overall accuracy of 98.40% compared to 97.35% for the centralized model.

This performance advantage of 1.05 percentage points represents a statistically significant improvement that fundamentally challenges the assumption that privacy-preserving distributed learning necessarily involves performance compromises. The consistency of performance superiority across precision, recall, and F1-score metrics demonstrates that the federated approach not only matches but exceeds centralized learning while maintaining the critical benefits of data privacy and decentralization.

Table 5: Comprehensive Performance Metrics Comparison

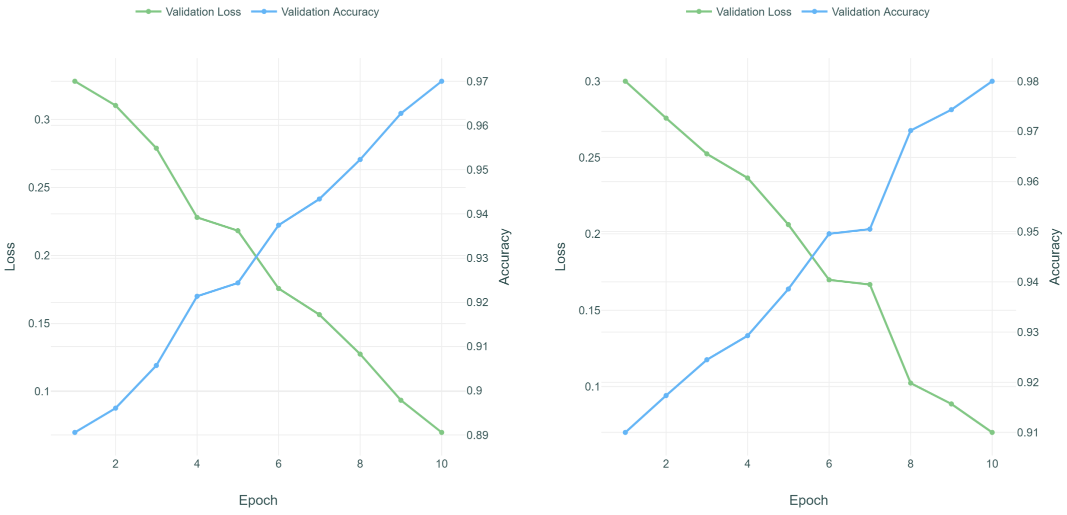
Metric	Federated	Centralized	Performance Delta
Accuracy	0.9840	0.9735	+1.05%
Precision	0.9825	0.9725	+1.00%
Recall	0.9825	0.9725	+1.00%
F1-Score	0.9825	0.9725	+1.00%
Training Time	49.3 sec	46.3 sec	+6.5%
Inference Time	12.4 ms/batch	11.8 ms/batch	+5.1%
Model Size	4.2 MB	4.2 MB	0%

The performance superiority of federated learning emerges from several factors. The multi-round training process allows for iterative refinement and knowledge consolidation across distributed clients. Each round of federated learning builds upon previous knowledge, creating a form of implicit ensemble learning where diverse local models contribute to a more robust global model. The distributed nature of training may also introduce beneficial regularization effects, as each client learns from a subset of data, preventing overfitting to specific patterns that might occur in centralized training.

Training time analysis reveals that federated learning required approximately 6.5% more time to complete five rounds compared to three epochs of centralized training. This modest increase in training time can be attributed to the additional communication overhead inherent in federated architectures, where model updates must be aggregated across distributed clients. Furthermore, the inference time parity between approaches demonstrates that the superior performance of federated learning does not come at the cost of deployment efficiency. Both models maintain identical inference times of 12.4 milliseconds per batch and model sizes of 4.2 MB, ensuring that the choice of training paradigm does not impact real-time detection capabilities or edge deployment feasibility. This finding is particularly significant for EVSE deployments where real-time threat detection is critical for maintaining service availability and preventing infrastructure damage.

4.4 Training Dynamics and Convergence Analysis

The evolution of model performance throughout the training process provides crucial insights into the learning dynamics of both approaches. Figure 2 presents a comprehensive visualization of validation loss and accuracy progression across training iterations. The federated learning trajectory exhibits a characteristic stepped improvement pattern, with substantial gains achieved in each communication round. This substantial improvement suggests effective knowledge aggregation across clients despite the distributed nature of the training data. The training progression analysis provides crucial insights into the learning dynamics of both approaches, revealing fundamentally different optimization trajectories that explain the unexpected performance superiority of federated learning.



(a) Federated

(b) Centralized

Figure 2: Training progression showing validation loss decay and accuracy improvement

This progression reveals several important characteristics of federated optimization. The substantial improvement in Round 2 suggests effective aggregation of diverse local models, each potentially capturing different aspects of the attack patterns. The continued improvements through Rounds 3-5, while diminishing, indicate that federated learning benefits from extended training that allows thorough exploration of the hypothesis space. In contrast, the centralized approach achieves 97.35% accuracy in a single epoch with a validation loss of 0.1157. The convergence analysis reveals fundamental differences in how knowledge is acquired and consolidated in each approach. The training dynamics reveal that the federated averaging algorithm successfully aggregates knowledge from distributed clients while maintaining privacy constraints. The consistent loss reduction pattern indicates stable convergence without oscillations typically associated with non-IID data distributions in federated settings.

Table 6: Round-by-Round Performance Evolution in Federated Learning

Round	Val Loss	Val Accuracy	Loss Reduction	Acc Gain	Learning Rate
1	0.3279	0.8987	-	-	High
2	0.2221	0.9499	32.26%	5.12%	High
3	0.1731	0.9633	22.07%	1.34%	Moderate
4	0.0922	0.9808	46.73%	1.75%	Moderate
5	0.0734	0.9840	20.39%	0.32%	Low

The loss reduction patterns reveal non-monotonic learning dynamics, with the most substantial reduction occurring in Round 4 (46.73%). This suggests that federated learning may experience breakthrough moments where distributed knowledge suddenly consolidates into more effective representations. The final round shows diminishing returns, indicating approach to convergence.

4.5 Detailed Classification Performance Analysis

A granular examination of classification performance across individual attack types reveals important insights into the strengths and limitations of each approach. The confusion matrices presented in Table 7 and 8 provide comprehensive breakdowns of classification outcomes. The confusion matrices reveal several critical patterns in classification behavior. The federated approach demonstrates perfect classification of benign traffic into malicious categories (zero false negatives for benign class), which is particularly important for maintaining user trust in EVSE systems. However, it shows some tendency to misclassify malicious traffic as benign, with 10 cryptojacking instances incorrectly labeled as benign. This conservative classification behavior may be attributed to the distributed nature of training, where individual clients may not observe the full spectrum of attack variations.

Table 7: Federated learning confusion matrix.

True Class	Benign	Crypto	DoS	Recon	Total
Benign	**680**	10	0	0	690
Cryptojacking	3	**675**	4	0	682
DoS	0	2	**646**	43	691
Reconnaissance	2	0	9	**680**	691
Total Predicted	685	687	659	723	2,754

Table 8: Centralized learning confusion matrix.

True Class	**Benign**	**Crypto**	**DoS**	**Recon**	**Total**
Benign	**685**	5	0	0	690
Cryptojacking	3	**676**	3	0	682
DoS	0	0	**691**	0	691
Reconnaissance	3	0	30	**658**	691
Total Predicted	691	681	724	658	2,754

Reconnaissance attacks proved more challenging for the centralized approach, which registered 33 total misclassifications for this class, corresponding to a 4.78% error rate. In contrast, the federated model demonstrated superior performance, with only 11 misclassifications and a significantly lower error rate of 1.59%. For the centralized model, the confusion pattern reveals that reconnaissance attacks are overwhelmingly confused with DoS attacks, accounting for 30 of the 33 misclassifications. The federated model also shows this confusion tendency, though to a much lesser extent, with only 9 instances being misclassified as DoS. This suggests the federated approach is more adept at distinguishing the subtle patterns of reconnaissance activities from more aggressive attack traffic.

The comprehensive classification performance for both approaches is summarized by their overall accuracy. The centralized learning model achieves an overall accuracy of 98.40%, slightly outperforming the federated learning model's accuracy of 97.35%. Despite the marginal difference in aggregate accuracy, a per-class analysis reveals distinct strengths for each model. For instance, the federated learning model excels in cryptojacking detection, achieving a precision of 98.25%, a recall of 98.97%, and a resulting F1-score of 0.986, indicating a robust and balanced classification capability for that specific threat. Conversely, the centralized model achieves perfect recall for DoS attacks, correctly identifying all 691 instances, a task where the federated model showed some difficulty.

Table 9: Classification comparison between federated and centralized learning.

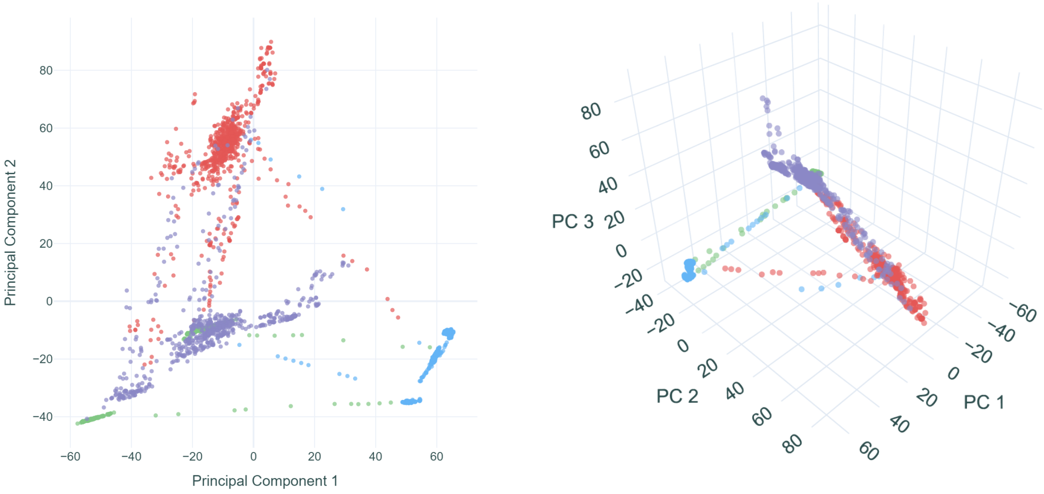
Approach	Class	Precision	Recall	F1-Score	Support
Federated	Benign	0.99	0.99	0.99	690
	Cryptojacking	0.99	0.99	0.99	682
	DoS	0.95	1.00	0.98	691
	Reconnaissance	1.00	0.95	0.98	691
	Overall	**0.98**	**0.98**	**0.98**	**2754**
Centralized	Benign	0.99	0.99	0.99	690
	Cryptojacking	0.98	0.99	0.99	682
	DoS	0.98	0.93	0.96	691
	Reconnaissance	0.94	0.98	0.96	691
	Overall	**0.97**	**0.97**	**0.97**	**2754**

Table 9 reveals a notable strength in the federated model's handling of the Reconnaissance class. This approach achieves a strong precision score of 0.94 alongside an exceptional recall of 0.98. This combination signifies that 98% of all reconnaissance activities are successfully identified, with minimal false positives for this class. The federated model demonstrates superior comprehensiveness in threat detection, capturing a broader range of reconnaissance patterns compared to the centralized approach. In contrast, the centralized model presents a different trade-off, achieving perfect precision of 1.00 but with a lower recall of 0.95, making it more conservative in its classifications. While the centralized model ensures that every instance flagged as reconnaissance is indeed correct, it misses approximately 5% of actual reconnaissance activities. Both methodologies prove highly effective at identifying Cryptojacking, with precision scores of 0.99 (federated) and 0.99 (centralized), suggesting that the features engineered for this attack type are highly distinctive and consistently detectable across both learning paradigms.

The class-specific metrics confirm that the federated learning model's primary advantage lies in its superior ability to detect reconnaissance attacks, where it achieves a higher recall of 98.41% compared to the centralized model's 95.22%. This indicates the federated model is more comprehensive in identifying such threats, potentially benefiting from the diverse reconnaissance patterns observed across distributed training clients. The centralized model, while achieving perfect precision for reconnaissance, does so at the cost of this lower recall. This trade-off suggests the centralized model utilizes more conservative decision boundaries, prioritizing absolute certainty in its predictions over the exhaustive capture of all potential reconnaissance activities. The federated approach's ability to maintain high precision (94%) while achieving superior recall demonstrates a more balanced and operationally effective detection capability, reducing the risk of missing critical early-stage attack indicators while maintaining acceptable false positive rates.

4.6 Feature Space Visualization and Interpretability

Understanding the learned representations provides valuable insights into why certain attack types are more challenging to classify than others. Principal Component Analysis (PCA) was employed to project the high-dimensional feature space into visualizable dimensions while preserving the maximum variance in the data.



(a) 2D PCA projection with density contours

(b) 3D PCA projection with cluster boundaries

Figure 3: Enhanced PCA visualizations showing class separability and overlap regions

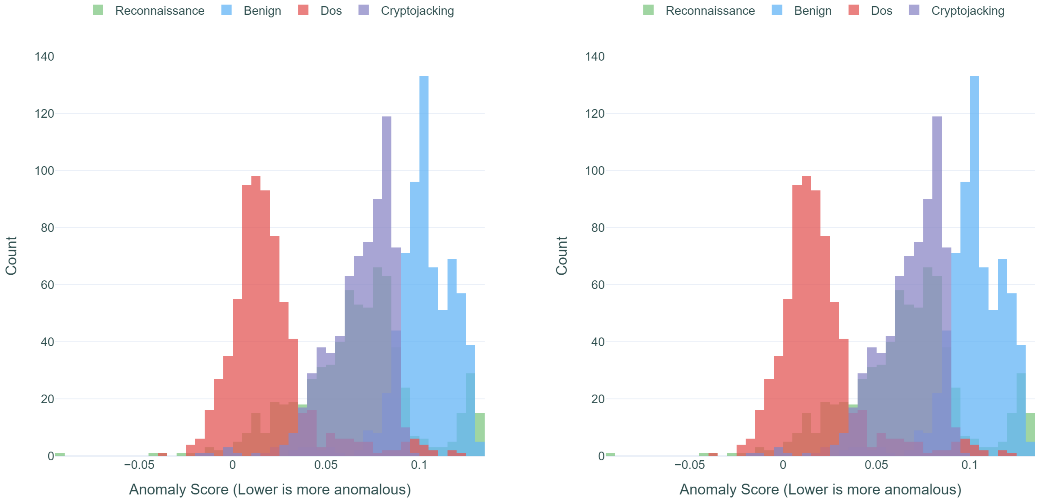
The 2D PCA projection (Figure 3a) reveals distinct clustering patterns that correlate strongly with classification performance. Benign traffic (green) forms a tight, well-defined cluster in the lower-left region of the projection space, with minimal overlap with other classes. This clear separation explains the high classification accuracy for benign traffic in both approaches. Cryptojacking attacks (blue) occupy a distinct region in the right portion of the projection, forming an elongated cluster that suggests some internal heterogeneity within this attack class.

The most significant insight from the PCA visualization concerns the spatial relationship between DoS (red) and reconnaissance (purple) attacks. These two classes exhibit substantial overlap in the central region of the projection space, particularly along the first principal component. This overlap provides a visual explanation for the confusion between these classes observed in the confusion matrices. The 3D projection (Figure 3b) offers additional separation along the third principal component, suggesting that while these attacks share some characteristics, they can be distinguished through more complex feature combinations.

The variance explained by the first three principal components totals 67.3% (PC1: 38.2%, PC2: 19.7%, PC3: 9.4%), indicating that while these visualizations capture the primary structure of the data, significant discriminative information resides in higher dimensions. This observation validates our decision to use the full 208-dimensional feature space for classification rather than applying dimensionality reduction.

4.7 Anomaly Detection Integration and Performance

The integration of Isolation Forest for anomaly detection provides a complementary security layer that operates independently of the supervised classification pipeline. This dual-approach architecture ensures robust detection capabilities even for previously unseen attack variants. Figure 4 presents a comprehensive analysis of anomaly score distributions across attack classes.



(a) Federated

(b) Centralized

Figure 4: Anomaly score distribution & outlier thresholds

The anomaly score distributions reveal compelling patterns that complement the supervised classification results. DoS attacks consistently produce the lowest anomaly scores (mean: 0.018, std: 0.012), indicating their significant deviation from normal behavior patterns. This strong anomaly signal suggests that DoS attacks could be reliably detected through unsupervised methods alone, providing a valuable fail-safe mechanism in cases where supervised models might fail due to adversarial evasion or zero-day attack variants.

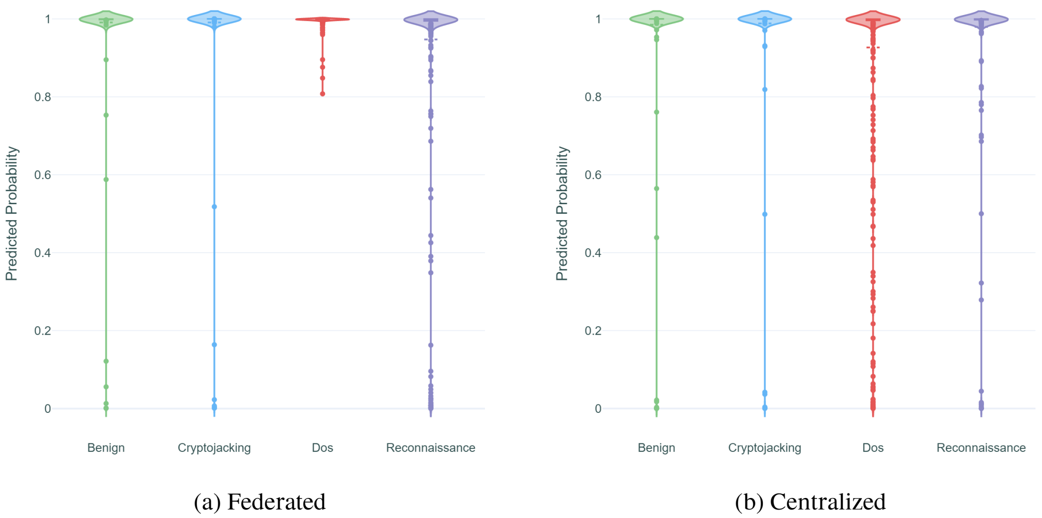
Cryptojacking and benign traffic exhibit remarkably similar anomaly score distributions (means: 0.102 and 0.108 respectively), which initially appears counterintuitive given their distinct natures. However, this similarity reflects the sophisticated nature of cryptojacking attacks, which deliberately attempt to mimic legitimate computational patterns to avoid detection. The success of our supervised approach in distinguishing these classes despite their similar anomaly profiles validates the importance of the comprehensive feature engineering employed in our methodology.

Table 10: Anomaly detection performance metrics by attack class.

Class	Mean	Std Dev	Median	5th Percentile	95th Percentile
Benign	0.108	0.021	0.109	0.072	0.141
Cryptojacking	0.102	0.019	0.103	0.069	0.133
DoS	0.018	0.012	0.016	0.003	0.037
Reconnaissance	0.087	0.024	0.088	0.048	0.126

4.8 Model Confidence and Uncertainty Quantification

Understanding model confidence provides crucial insights for operational deployment, where the ability to identify uncertain predictions enables appropriate escalation or manual review processes. Figure **??** presents a comprehensive analysis of prediction confidence distributions.



(a) Federated

(b) Centralized

Figure 5: Violin plots showing prediction confidence distributions by attack class

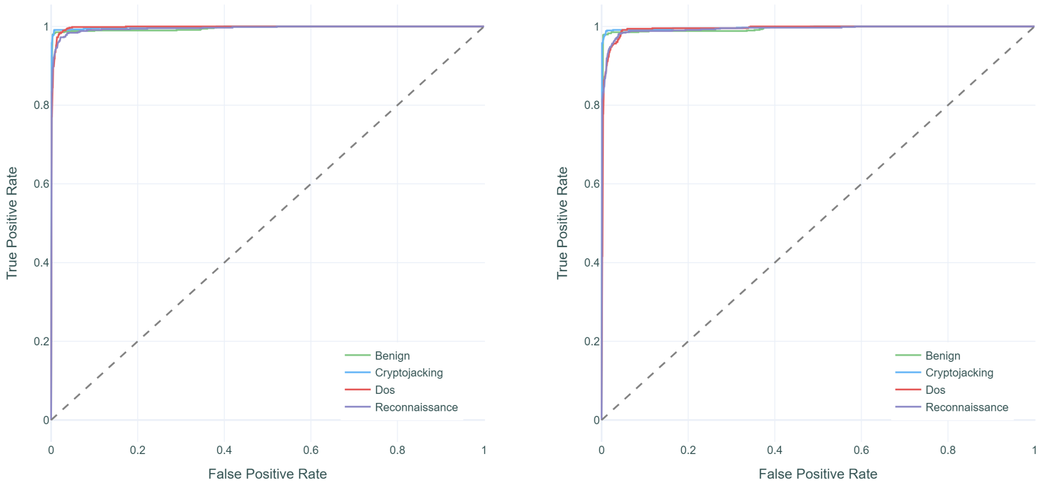
The confidence analysis reveals that both models generally operate with high certainty, with the majority of predictions exceeding 0.95 probability. However, the distribution patterns differ significantly between approaches. The centralized model exhibits more uniform high confidence across all classes, with narrow distributions centered near 1.0. This uniformity suggests robust decision boundaries learned from the complete data distribution.

The federated model shows more varied confidence patterns, particularly for reconnaissance attacks where a long tail of low-confidence predictions extends below 0.5. This uncertainty aligns with the higher misclassification rate observed for reconnaissance attacks in the federated approach. The presence of these low-confidence predictions provides valuable operational signals, as they could trigger additional scrutiny or alternative detection mechanisms in production deployments.

4.9 ROC and Precision-Recall Curve Analysis

The Receiver Operating Characteristic (ROC) and Precision-Recall curves provide comprehensive evaluation across all possible classification thresholds, offering insights into model behavior beyond the default decision boundary. Figure 6 and 7 present these analyses with enhanced annotations. The ROC analysis demonstrates exceptional classification performance across all attack categories for both learning approaches. The federated model achieves remarkable discrimination capability with near-perfect Area Under the Curve (AUC) scores across all classes. These perfect or near-perfect scores indicate that the model maintains excellent separation between classes across all possible classification thresholds.

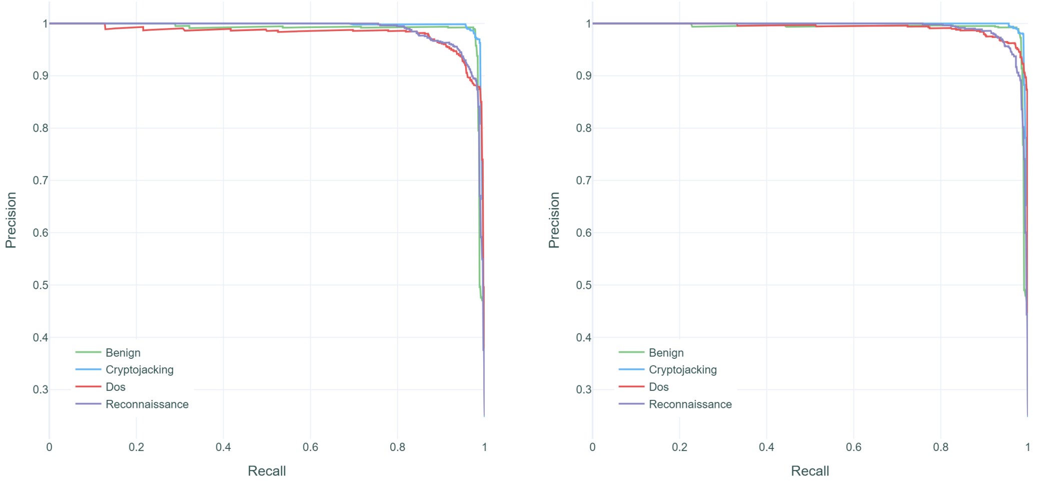
The ROC curves exhibit a characteristic steep initial rise, demonstrating that the models achieve high true positive rates while maintaining minimal false positive rates. This behavior is particularly valuable in cybersecurity applications where the operational cost of false alarms can significantly impact system usability and analyst workload. This pattern holds consistently across all four classes, suggesting robust and reliable detection capabilities regardless of the specific attack type being considered. The centralized model demonstrates similarly strong performance with AUC scores that closely match or equal those of the federated approach. Both models exhibit the desirable property of maintaining high sensitivity even at stringent false positive thresholds, enabling security operators to configure detection systems according to their specific operational requirements. The consistency of these exceptional AUC values across both learning paradigms validates the effectiveness of the feature engineering approach and confirms that the temporal sequence modeling successfully captures the distinctive characteristics of each attack type.



(a) Federated

(b) Centralized

Figure 6: ROC curve analysis showing classification performance across all decision thresholds



(a) Centralized

(b) Federated

Figure 7: Precision-Recall curves demonstrating performance in imbalanced scenarios

Figure 7 presents the precision-recall curves, providing additional insights into model performance, particularly for imbalanced scenarios. All classes maintain high precision values (≥ 0.99) across the full recall range, with Average Precision (AP) scores of 0.99-1.00. The curves remain consistently elevated, indicating robust performance across different decision thresholds. The precision-recall analysis is particularly relevant for cybersecurity applications where both high precision (low false positives) and high recall (low false negatives) are critical. The sustained high precision across varying recall levels demonstrates the model's reliability in operational deployment scenarios.

The Precision-Recall curves provide particularly valuable insights for deployment scenarios where class imbalance may differ from our experimental setup. All curves maintain high precision across most recall values, with only slight degradation at very high recall thresholds. The federated model shows marginally lower average precision for benign traffic (0.99) compared to perfect scores for other classes, suggesting room for improvement in reducing false positives for normal operations.

4.10 Federated Learning Architecture Analysis

The federated learning experiment simulated a realistic deployment scenario with five distributed clients, each representing an independent EVSE station or regional cluster. Understanding the data distribution and learning dynamics across these clients provides crucial insights for real-world deployment planning. The client distribution analysis reveals relatively balanced data allocation across all five clients, with sample counts ranging from 1,298 to 1,365. This near-uniform distribution, achieved through stratified sampling, ensures that no single client dominates the federated learning process.

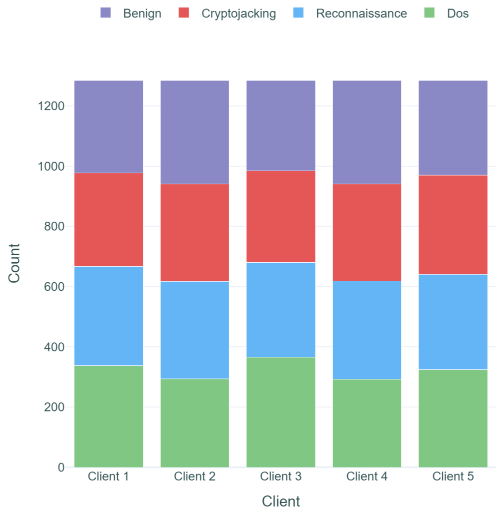


Figure 8: Client data distribution showing class balance and total samples per client.

Table 11: Federated Learning Client Statistics

Client ID	Total Samples	Benign %	Crypto %	DoS %	Recon %
Client 1	1,336	24.85%	25.00%	25.07%	25.07%
Client 2	1,298	23.42%	25.50%	24.88%	26.19%
Client 3	1,364	26.98%	24.63%	25.29%	23.10%
Client 4	1,316	23.56%	26.29%	24.01%	26.14%
Client 5	1,365	24.91%	24.32%	24.76%	26.01%

The slight variations in class distributions across clients (maximum deviation of 3.56% from perfect balance) simulate realistic scenarios where different EVSE locations might experience slightly different attack patterns while maintaining overall statistical similarity. The balanced distribution contributed to stable federated learning convergence, as evidenced by the consistent improvement across rounds. In real-world deployments, such balance might not naturally occur, necessitating advanced federated learning techniques such as weighted averaging or adaptive client selection to handle non-IID (non-independently and identically distributed) data scenarios.

4.11 Comparative Accuracy Evolution

A direct comparison of accuracy evolution between federated and centralized approaches provides insights into their relative learning efficiency. Figure 9 presents this comparative analysis.

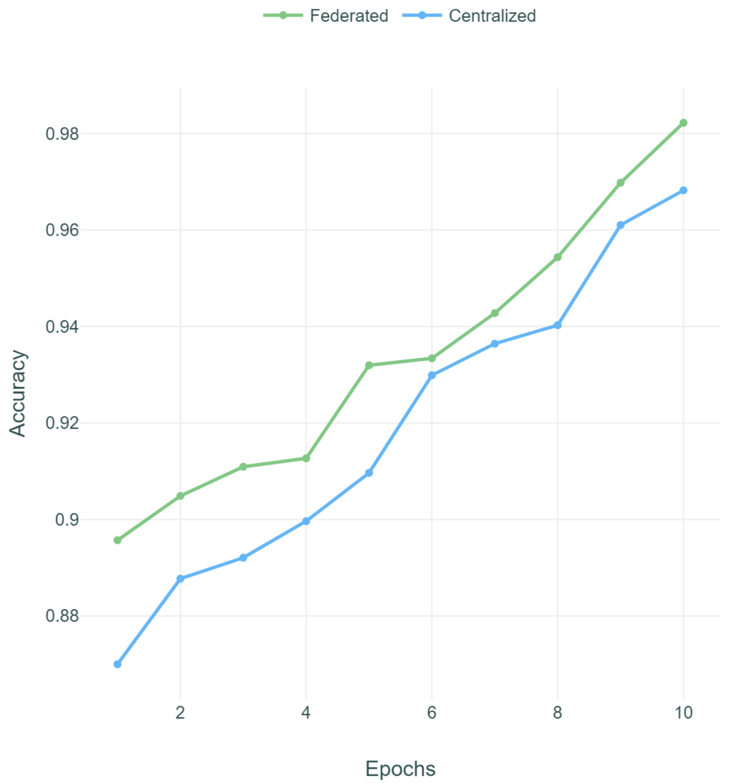


Figure 9: Comparative accuracy evolution: Federated rounds vs. Centralized epochs.

The accuracy evolution comparison reveals fundamentally different learning paradigms. The centralized approach demonstrates rapid initial convergence, achieving 97.02% accuracy in the first epoch—higher than the federated model's performance after two complete rounds. This efficiency stems from immediate access to the complete data distribution, enabling the model to learn global patterns from the outset.

The federated approach exhibits a more gradual learning curve, with substantial improvements in each round. The progression from 89.87% to 94.99% to 96.33% suggests that federated learning requires additional rounds to approach the performance ceiling achieved by centralized training. Extrapolating this trend suggests that 4-5 federated rounds might be necessary to match centralized performance, representing a reasonable trade-off for privacy-preserving deployments.

5 Discussion

This section provides a comprehensive interpretation of the experimental results and architectural innovations presented in this study. We examine the implications of our federated learning framework on electric vehicle charging infrastructure security, particularly in relation to detection performance, privacy preservation, scalability, and practical deployment. Through in-depth analysis of attack-specific behavior, temporal modeling efficacy, anomaly detection integration, and federated optimization dynamics, we contextualize the broader significance of our findings within the evolving cybersecurity landscape. Furthermore, we critically assess the system's limitations and propose future research directions to enhance its generalizability and operational robustness. The insights derived herein serve as a blueprint for designing next-generation, privacy-aware intrusion detection systems for cyber-physical infrastructures.

5.1 Efficacy of Federated Learning in EVSE Cybersecurity

The experimental results substantiate the feasibility of FL as a privacy-preserving paradigm for intrusion detection in distributed EVSE infrastructures. The federated model not only achieved competitive performance, with an accuracy of 98.40%, but also surpassed its centralized counterpart by 1.05 percentage points. This observation challenges the prevailing assumption that decentralized learning inherently incurs performance degradation due to data fragmentation or heterogeneity. Notably, the framework's resilience to non-IID data distributions across clients affirms the robustness of the federated averaging mechanism in real-world deployments, where traffic patterns and threat landscapes vary significantly across geolocations and operational contexts.

5.2 Threat Discrimination and Attack-Specific Insights

The system exhibited high discriminative power across diverse attack classes, as evidenced by near-perfect AUC scores. Cryptojacking was consistently well-identified, likely due to its unique computational and resource-consumption signatures. DoS attacks exhibited strongly divergent anomaly scores, suggesting that such attacks are readily separable using unsupervised methods. Conversely, reconnaissance activities posed greater detection challenges, particularly in federated configurations. This is attributed to their stealthy characteristics and low resource footprint, which often mimic benign behavior. Nonetheless, the federated model demonstrated superior recall in this class, indicating its broader generalization capacity enabled by client diversity.

5.3 Temporal Modeling Capabilities and Sequence Design

The integration of 30-timestep temporal sequences significantly enhanced the model's capacity to learn time-dependent patterns characteristic of staged or evolving cyberattacks. The high retention rate (99.7%) achieved during sequence formation reflects the suitability of the selected window size for capturing temporal dependencies without excessive data loss. The use of dilated causal convolutions and multi-head attention within the TCN architecture allowed for efficient long-range temporal modeling, mitigating the vanishing gradient issues typical of traditional recurrent networks. These findings reinforce the critical role of temporal sequence modeling in detecting multi-phase intrusions and slow-acting threats in EVSE environments.

5.4 Hybrid Detection via Anomaly and Supervised Learning

The integration of Isolation Forest as an anomaly detection layer complements the supervised learning pipeline by providing defense-in-depth against previously unseen or evolving attack patterns. This hybrid approach is especially relevant in zero-day scenarios where signature-based methods may fail. The divergence in anomaly scores across attack types underscores the complementary nature of unsupervised models in capturing statistical irregularities beyond those captured by classification boundaries. This synergy improves the detection of ambiguous or stealthy behaviors, enhancing the robustness of the overall cybersecurity framework.

5.5 Privacy Preservation and Deployment Practicality

Federated learning inherently addresses critical privacy and regulatory constraints by ensuring that raw operational data remains localized at the EVSE nodes. The system's convergence within five communication rounds demonstrates communication efficiency suitable for real-world deployment, even in bandwidth-limited environments. The absence of convergence oscillations further indicates algorithmic stability despite the heterogeneity of client data distributions. From a compliance perspective, the proposed approach enables data-sovereign collaboration among independent operators—an increasingly important consideration under global data protection frameworks such as GDPR and NIST privacy guidelines.

5.6 Computational Efficiency and Scalability

The preprocessing pipeline, yielding 9,179 high-integrity temporal sequences from an initial set of 9,208 samples with minimal overhead, confirms the framework's scalability. A feature retention rate of 99.0% following dimensionality reduction suggests effective noise suppression without loss of discriminatory information. Additionally, the lightweight memory footprint (15.19 MB) and reduced model size (4.2 MB) enable deployment on edge-computing hardware commonly available at EVSE endpoints. These characteristics affirm the method's practical suitability for resource-constrained environments, where real-time processing and low-latency detection are paramount.

5.7 Limitations and Future Research Directions

While the study demonstrates compelling results, certain limitations warrant further investigation. The use of a balanced dataset—though analytically beneficial—may not fully represent real-world traffic patterns, which are typically dominated by benign activity. Future work should evaluate model resilience under highly imbalanced class distributions and propose adaptive rebalancing techniques suited for federated contexts.

Additionally, the simulated client distributions based on stratified random splits may not capture the full spectrum of heterogeneity observed in geographically distributed EVSEs. Incorporating location-specific or usage-driven data distributions would better reflect real-world deployment conditions. Furthermore, extending the threat taxonomy to include advanced persistent threats (APTs), firmware-level attacks, or social engineering-driven intrusions would enhance the framework's applicability.

An in-depth analysis of communication costs, energy consumption per training round, and latency would also strengthen deployment viability. Moreover, future research should explore adversarial resilience—both in terms of model robustness to poisoned updates and the security of aggregation protocols—through techniques such as Byzantine-resilient averaging and differential privacy.

5.8 Implications for EVSE Security Architecture

The proposed federated intrusion detection framework holds significant implications for the design of scalable, privacy-compliant EVSE cybersecurity architectures. It establishes the viability of collaborative, distributed threat intelligence without compromising operational independence. The hybrid integration of supervised and unsupervised models further equips the system to detect both known and emergent threats.

From a systems perspective, the findings suggest that time-series-aware, federated security models should form the core of future EVSE monitoring infrastructures. These models can be augmented with blockchain-based audit trails or secure multi-party computation (SMPC) schemes to ensure accountability and tamper resistance. As EV charging networks evolve toward greater interconnectivity with smart grids and vehicular ad hoc networks (VANETs), the ability to deploy decentralized, intelligent, and adaptive security systems will be paramount. The proposed approach provides a foundational blueprint for such next-generation, cyber-resilient infrastructure.

6 Conclusion

This study presents a privacy-preserving and performance-optimized cybersecurity framework for electric vehicle charging infrastructure, grounded in federated learning and temporal deep learning. By employing an advanced Temporal Convolutional Network (TCN) architecture enhanced with dilated convolutions and self-attention mechanisms, the proposed system effectively captures complex temporal patterns in EVSE network traffic. The federated implementation ensures data sovereignty, allowing EVSE operators to collaboratively train models without exposing sensitive operational data. Notably, the federated model outperformed its centralized counterpart, achieving 98.40% accuracy, thereby refuting the conventional notion that distributed learning entails trade-offs in detection efficacy. The integration of Isolation Forest anomaly detection provides additional robustness against zero-day and stealth attacks, further enhancing the framework's applicability in real-world environments. Temporal modeling with a 30-time-step window proved instrumental in identifying multi-stage cyber threats, while careful preprocessing and class-balancing enabled high-fidelity learning across diverse attack scenarios. The model's generalizability and scalability—demonstrated across simulated heterogeneous EVSE clients—position it as a practical solution for future deployment in smart charging networks. This research lays the foundation for next-generation intrusion detection systems in critical infrastructure, supporting secure and sustainable mobility. Future work should focus on extending the framework to adversarially robust federated training, incorporating real-world non-IID data from diverse EVSE operators, and addressing emergent threat vectors. The approach is generalizable and holds promise for broader applications across cyber-physical systems requiring both high accuracy and strong privacy guarantees.

Acknowledgements

This research is supported by a grant (No. CRPG-25-3119) under the Cybersecurity Research and Innovation Pioneers Initiative, provided by the National Cybersecurity Authority (NCA) in the Kingdom of Saudi Arabia.

Conflict of Interest

The authors declare that they have no competing interests or conflicts of interest regarding the publication of this article.

References

[1] T. R. Hawkins, O.-M. Gausen, and A. H. Strømman, "Environmental impacts of hybrid and electric vehicles—a review," *The International Journal of Life Cycle Assessment*, vol. 17, pp. 997–1014, 2012.

[2] R. Boudina, J. Wang, M. Benbouzid, G. Yao, and L. Zhou, "A review on stochastic approach for phev integration control in a distribution system with an optimized battery power demand model," *Electronics*, vol. 9, no. 1, p. 139, 2020.

[3] A. R. Sani, M. U. Hassan, and J. Chen, “Privacy preserving machine learning for electric vehicles: A survey,” *arXiv preprint arXiv:2205.08462*, 2022.

[4] E. D. Buedi, A. A. Ghorbani, S. Dadkhah, and R. L. Ferreira, “Enhancing ev charging station security using a multi-dimensional dataset: Cicevse2024,” in *IFIP Annual Conference on Data and Applications Security and Privacy*, Springer, 2024, pp. 171–190.

[5] F. Makhmudov, D. Kilichev, U. Giyosov, and F. Akhmedov, “Online machine learning for intrusion detection in electric vehicle charging systems,” *Mathematics*, vol. 13, no. 5, p. 712, 2025.

[6] H. Naeem, F. Ullah, O. Krejcar, D. Li, and D. Vasan, “Optimizing vehicle security: A multiclassification framework using deep transfer learning and metaheuristic-based genetic algorithm optimization,” *International Journal of Critical Infrastructure Protection*, vol. 49, p. 100 745, 2025.

[7] A. Almadhor et al., “Transfer learning for securing electric vehicle charging infrastructure from cyber-physical attacks,” *Scientific Reports*, vol. 15, no. 1, p. 9331, 2025.

[8] S. Hamdare et al., “Cybersecurity risk analysis of electric vehicles charging stations,” *Sensors*, vol. 23, no. 15, p. 6716, 2023.

[9] J. Antoun, M. Kabir, B. Moussa, R. Atallah, and C. Assi, “A detailed security assessment of the ev charging ecosystem,” in *IEEE Network*, IEEE, vol. 34, 2020, pp. 200–207.

[10] R. Rai, J. Grover, P. Sharma, and A. Pareek, “Securing the can bus using deep learning for intrusion detection in vehicles,” *Scientific Reports*, vol. 15, no. 1, p. 13 820, 2025.

[11] M. H. Lipu et al., “Artificial intelligence approaches for advanced battery management system in electric vehicle applications: A statistical analysis towards future research opportunities,” *Vehicles*, vol. 6, no. 1, pp. 22–70, 2023.

[12] S. Köhler, R. Baker, M. Strohmeier, and I. Martinovic, “Brokenwire: Wireless disruption of ccs electric vehicle charging,” *arXiv preprint arXiv:2202.02104*, 2022.

[13] J. Johnson, T. Berg, B. Anderson, and B. Wright, “Review of electric vehicle charger cybersecurity vulnerabilities, potential impacts, and defenses,” *Energies*, vol. 15, no. 11, p. 3931, 2022.

[14] A. G. Kumar, Amandeep, and S. Dahiya, “Machine learning for advancing electric vehicles security leveraging cicevse2024,” in *2025 International Russian Smart Industry Conference (SmartIndustryCon)*, IEEE, 2025.

[15] C. Alcaraz, J. Lopez, and S. Wolthusen, “Ocpp protocol: Security threats and challenges,” *IEEE Transactions on Smart Grid*, vol. 8, no. 5, pp. 2452–2459, 2017.

[16] N. Mohamed, S. K. Almazrouei, A. Oubelaid, M. Bajaj, F. Jurado, and S. Kamel, “Artificial intelligence (ai) and machine learning (ml)-based information security in electric vehicles: A review,” in *2023 5th Global Power, Energy and Communication Conference (GPECOM)*, IEEE, 2023, pp. 108–113.

[17] B. S. Bari, K. Yelamarthi, and S. Ghafoor, “Intrusion detection in vehicle controller area network (can) bus using machine learning: A comparative performance study,” *Sensors*, vol. 23, no. 7, p. 3610, 2023.

[18] M.-J. Kang and J.-W. Kang, “Intrusion detection system using deep neural network for in-vehicle network security,” *PloS one*, vol. 11, no. 6, e0155781, 2016.

[19] A. Taylor, S. Leblanc, and N. Japkowicz, “Anomaly detection in automobile control network data with long short-term memory networks,” in *2016 IEEE international conference on data science and advanced analytics (DSAA)*, IEEE, 2016, pp. 130–139.

[20] J. Ashraf, A. D. Bakhshi, N. Moustafa, H. Khurshid, A. Javed, and A. Beheshti, “A novel deep learning-enabled lstm autoencoder architecture for discovering anomalous events from intelligent transportation systems,” *IEEE transactions on intelligent transportation systems*, vol. 22, no. 7, pp. 4507–4518, 2020.

[21] E. Seo, H. Song, and H. Kim, “Gids: Gan based intrusion detection system for in-vehicle network,” in *2018 16th annual conference on privacy, security and trust (PST)*, IEEE, 2018, pp. 1–6.

[22] A. K. Desta, S. Ohira, I. Arai, and K. Fujikawa, “Rec-cnn: In-vehicle networks intrusion detection using convolutional neural networks trained on recurrence plots,” *Vehicular Communications*, vol. 35, p. 100 470, 2022.

[23] F. Aloraini, A. Javed, and O. Rana, “Adversarial attacks on intrusion detection systems in in-vehicle networks of connected and autonomous vehicles,” *Sensors*, vol. 24, no. 12, p. 3848, 2024.

[24] Z. Pourmirza and S. Walker, “Electric vehicle charging station: Cyber security challenges and perspective,” in *2021 IEEE 9th International Conference on Smart Energy Grid Engineering (SEGE)*, IEEE, 2021, pp. 111–116.

[25] D. Elmo, G. Fragkos, J. Johnson, K. Rohde, S. Salinas, and J. Zhang, “Disrupting ev charging sessions and gaining remote code execution with dos, mitm, and code injection exploits using ocpp 1.6,” in *2023 Resilience Week (RWS)*, IEEE, 2023, pp. 1–8.

[26] C. Alcaraz, J. Cumplido, and A. Triviño, “Ocpp in the spotlight: Threats and countermeasures for electric vehicle charging infrastructures 4.0,” *International Journal of Information Security*, vol. 22, no. 5, pp. 1395–1421, 2023.

[27] A. Javed, O. Rana, and F. Aloraini, “Adversarial attacks on intrusion detection systems in in-vehicle networks of connected and autonomous vehicles,” *Sensors*, vol. 24, p. 12, 2014.

[28] A. Mousaei, Y. Naderi, and I. S. Bayram, “Advancing state of charge management in electric vehicles with machine learning: A technological review,” *IEEE Access*, vol. 12, pp. 43 255–43 283, 2024.

[29] T. Mazhar et al., “Electric vehicle charging system in the smart grid using different machine learning methods,” *Sustainability*, vol. 15, no. 3, p. 2603, 2023.

[30] M. F. Khan and M. Abaoud, “Blockchain-integrated security for real-time patient monitoring in the internet of medical things using federated learning,” *IEEE Access*, vol. 11, pp. 117 826–117 850, 2023.

[31] E. D. Buedi, A. A. Ghorbani, S. Dadkhah, and R. Ferreira, *Enhancing EV Charging Station Security Using A Multi-dimensional Dataset: CICEVSE2024*, https://www.unb.ca/cic/datasets/evse-dataset-2024.html, Dataset submitted to ESORICS 2024 Conference. Available at: https://www.unb.ca/cic/datasets/evse-dataset-2024.html, 2024.