

Smart grid false data injection detection through federated learning with deep learning models

Raseel Alshamasi^a and Dina M. Ibrahim^{a*}

^aDepartment of Information Technology, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia

CHRONICLE

Article history:

Received December 18 2025

Received in Revised Format

December 26 2025

Accepted February 2 2026

Available online February 2

2026

Keywords:

Deep learning (DL)

Smart grid

Federated learning

Security

Privacy

False Data Injection (FDI)

attack

ABSTRACT

The security of smart grids is seriously threatened by false data injection (FDI) attacks. Falsified data is maliciously injected into the grid's measurement and control systems as part of these attacks, which might seriously disrupt the power supply and jeopardize system integrity. In the context of smart grids, it is also imperative to address the issue of consumer privacy and the protection of their sensitive data. The main objective of this work is to provide a collaborative framework based on federated learning to detect various FDI dangers while protecting SG's resources and privacy. We have implemented several technologies that provide a good solution in order to accomplish this goal. Using a dataset designed to replicate attacks on the power system environment, we used federated learning to locally train models using the data stored on the sensors. The best model should then be chosen by comparing the outcomes. These outcomes demonstrate the potential of our framework, which has used mixed models to repel attacks, short-circuit faults, and maintain lines with a 98% accuracy rate during the federated learning phase.

© 2026 by the authors; licensee Growing Science, Canada

1. Introduction

The smart grid (SG) is an advanced power network that enables the two-way exchange of energy and information between customers and suppliers. The incorporation of information and communication technology into power networks allows for widespread control, automation, and connectivity, spanning from energy producing facilities to the point of consumption. Nevertheless, the advancement of wireless connections, the heightened level of autonomy, and escalating softwarization and virtualization trends have increased the vulnerability to attacks and potential for threats in SGs. Furthermore, given the constant stream of up-to-date data and the presence of online tools for managing energy usage, it is imperative to address the issue of customer privacy and the protection of their confidential information in the context of SGs.

To mitigate potential threats and weaknesses in developing power networks, there is a growing need to further research security and privacy mechanisms. Furthermore, there has been a growing utilization of machine intelligence and machine learning (ML) methods in various aspects of SGs. ML models are currently the dominant approach for detecting attacks and analyzing threats. Nevertheless, even though these algorithms exhibit exceptional precision and dependability, ML systems are susceptible to a category of hostile actions known as adversarial machine learning (AML) attacks (Mirzaee et al., 2022). your findings with previous works or published results of other researchers. Additional section is allowed as long as the main section structure is intact. Smart grid security and dependability are seriously threatened by false data injection attacks (FDI). Falsified data is maliciously injected into the grid's measurement and control systems as part of these attacks, which might seriously disrupt the power supply and jeopardize system integrity. Because smart grids rely on cutting-edge information and communication technologies, which create new attack surfaces, they are especially vulnerable as cyber-physical systems. By creating attack vectors that do not alter the residuals in state estimation equations, FDI might evade conventional bad data detection techniques and become difficult to detect (Lin et al., 2022; Hu et al., 2023). Successful FDI can have serious repercussions, including the possibility of blackouts, financial losses, and poor operational decisions. The complexity of FDI is increasing along with smart grid technologies, which calls for the creation of sophisticated detection and

* Corresponding author

E-mail d.husseini@qu.edu.sa (D. M. Ibrahim)

ISSN 1929-5812 (Online) - ISSN 1929-5804 (Print)

2026 Growing Science Ltd.

doi: 10.5267/j.dsl.2026.2.004

prevention systems to guarantee the safety and dependability of these vital infrastructure systems (Drayer et al., 2019; Zhang et al., 2020).

In order to improve the communication between client devices and servers, the study intends to create a distributed model utilizing the Federated Learning (FL) methodology. By removing and resolving the issue of different numbers of customers, this will address the issue of clients not using FL on the SG platform. Power System Attack, which includes a variety of SG attacks, is the appropriate dataset for the research. The main contributions in this paper are:

- To propose and develop a model based on FL.
- To improve process efficiency, the FL model will be implemented in the SG system.
- To eliminate the need for central servers in processes involving the handling of personal information.
- To improve privacy in SG.
- To defend the SG from attacks by FDI.

2. Related Studies

The increasing digitization of power grids has expanded the attack surface for cyber threats, particularly False Data Injection Attacks (FDIA), which manipulate state estimation to destabilise grid operations. In parallel, Distributed Denial of Service (DDoS) and replay attacks pose significant risks to the availability and integrity of smart grid communications. In response, a substantial body of research has applied deep learning (DL) and federated learning (FL) techniques to detect anomalies in smart grid environments. This review synthesises recent contributions, organised by methodological approach.

2.1 Hybrid Deep Learning Models for Attack Detection

Several studies have adopted hybrid deep learning architectures to capture both the spatial and temporal characteristics of grid measurement data. Niu et al. (2019) proposed a convolutional neural network-long short-term memory (CNN-LSTM) framework for FDIA detection using the IEEE 39-bus system and the NSL-KDD dataset. While the model achieved over 90% accuracy for high-magnitude attacks, it demonstrated limited sensitivity to low-power intrusions and assumed constrained attacker knowledge. Similarly, Diaba and Elmusrati (2023) applied a CNN-LSTM model to the CICIDS-2017 dataset for DDoS detection, reporting 99.7% accuracy. Despite outperforming random forest and support vector machine baselines, the study acknowledged computational complexity and limited generalisability due to single-dataset validation. Abdallah et al. (2021) also employed a CNN-LSTM hybrid for anomaly detection in software-defined networks (SDNs), though their work focused on network infrastructure rather than power systems directly. Expanding on temporal modelling, Elsaedy et al. (2021) employed a deep restricted Boltzmann machine (RBM) to detect replay and DDoS attacks in smart city environments. The model processed multivariate time-series data and outperformed conventional classifiers in Wilcoxon–Holm significance tests. However, its high computational demand and reliance on synthetic datasets constrain real-world deployment. In a related study, Elsaedy et al. (2020) utilised a deep CNN for replay attack detection, achieving statistical significance via the Wilcoxon signed-rank test, though the model required extensive training data and computing resources.

2.2 Autoencoder and Generative Adversarial Network Approaches

Autoencoder-based methods have been widely explored for semi-supervised and unsupervised FDIA detection. Zhang et al. (2020) developed a generative adversarial network-autoencoder (GAN-AE) framework for distribution systems, achieving 97.85% accuracy on the IEEE 123-bus system. The model effectively detected imperceptible attacks but relied heavily on labelled data, limiting scalability. Siniosoglou et al. (2021) proposed MENSEA, an autoencoder-GAN intrusion detection system for Modbus/TCP and DNP3 protocols. Validated on IEEE 14-bus and 30-bus systems, MENSEA achieved an accuracy of 0.8835 in a hydro plant scenario. Nonetheless, the system exhibited limited adaptability to zero-day attacks and required substantial labelled datasets. Ali and Li (2019) introduced a multilevel autoencoder integrated with multiple kernel learning (MKL) for DDoS attack detection. Evaluated on the UNB ISCX and UNSW-NB15 datasets, the model achieved 97% accuracy and demonstrated superior precision compared to six alternative machine learning approaches. However, the authors acknowledged the model's structural complexity, constrained assessment, and lack of validation on real-world attack data as key limitations.

2.3 Adversarial Vulnerabilities and Robustness

The robustness of DL-based intrusion detection systems to adversarial manipulation has emerged as a critical concern. Sayghe et al. (2020) examined the vulnerability of multilayer perceptron (MLP) models to L-BFGS and Jacobian-based Saliency Map Attacks (JSMA) using IEEE 14-bus system data. Their results indicated that JSMA was more effective at evading detection; at a delta value of 0.2, model accuracy was 87.65% and recall was 80.02%. The study highlighted that increasing perturbation

magnitude reduced detection accuracy, underscoring the need for adversarially robust architectures. As the authors note, experiments were conducted in simulated environments, limiting generalisability to real-world power grid deployments.

2.4 Federated Learning for Privacy-Preserving Detection

In response to data privacy concerns, several researchers have proposed federated learning frameworks for decentralised attack detection. Zhao et al. (2021) introduced a decentralised FDIA detection framework in which sensors train local models and transmit updates to a central server. While this approach preserves data privacy and demonstrated improved efficiency, the study acknowledged that limited sensor computing power constrains local model training. Additionally, the framework assumes sensor reliability, which may not hold in practice. Al-Quraan et al. (2023) proposed FedTrees, a tree-based FL framework for power system monitoring. Tested on the Tetouan dataset, the Light Gradient Boosting Machine (LGBM) model outperformed LSTM, achieving a lower mean absolute error of 0.017. However, FedTrees is designed specifically for tree-based models, limiting its flexibility for neural network architectures, and assumes evenly distributed data among clients, a condition that may not reflect real-world heterogeneity. Jithish et al. (2023) and Li et al. (2022) have also contributed to the growing body of FL research for smart grid security. Li et al. (2022) proposed a secure federated deep learning approach for FDIA detection, while Jithish et al. (2023) explored distributed anomaly detection using FL. Both studies highlight the potential of FL to mitigate privacy risks, though challenges related to communication efficiency and heterogeneous client capabilities persist.

2.5 Convolutional Neural Networks and Image-Based Representations

Convolutional neural networks have been applied both to raw measurement data and to image-based representations of grid states. VS (2021) proposed a CNN-based multi-label classification model for FDIA detection, tested on IEEE 14-bus and 30-bus systems. The model demonstrated superior accuracy and precision in capturing power flow correlations but incurred high computational costs and required extensive training data. Mohammadpourfard et al. (2021) introduced a two-stage DL model that converts power system data into images for classification. Validated on IEEE 57-bus systems, the approach achieved high accuracy; however, its computational intensity and limited evaluation across diverse datasets constrain its readiness for real-time deployment.

2.6 Threat Attribution and Traditional Machine Learning

Beyond deep learning, traditional machine learning methods continue to be refined for threat attribution in industrial control systems. Zhang et al. (2024) developed a threat attribution and reasoning framework for industrial Internet of Things (IIoT) environments, employing support vector machines (SVM), decision trees, random forests, XGBoost, and k-nearest neighbours (KNN). Using power system data, random forests achieved the highest accuracy (0.95), followed by KNN (0.92), decision trees and XGBoost (0.87 each), and SVM (0.54). While the study contributes to asset-centric security ontology development, the performance disparity across classifiers suggests that model selection remains highly context-dependent.

2.7 Classical and Signal Processing-Based Detection

Prior to the widespread adoption of deep learning, classical detection methods formed the foundation of smart grid security research. He et al. (2017) proposed an intelligent DL-based detection mechanism combining a state vector estimator (SVE) with deep-learning-based identification (DLBI). The system achieved a 95.69% detection rate with a 3.47% false positive rate but struggled to detect attacks that closely mimicked normal operational patterns. Drayer and Routtenberg (2019) employed graph signal processing for FDIA detection, while Hu et al. (2023) applied expectation-maximisation algorithms. Although these methods offer interpretability advantages, they generally underperform relative to DL-based approaches on high-dimensional data.

2.8 Synthesis and Research Gaps

Collectively, the literature demonstrates that hybrid DL models, particularly CNN-LSTM architectures, and autoencoder-based methods have significantly advanced the accuracy of FDIA and DDoS detection in smart grids. However, several persistent gaps remain. First, the majority of studies rely on simulated datasets (e.g., IEEE bus systems, NSL-KDD, CICIDS-2017) or synthetic attack data, limiting ecological validity. Second, computational complexity and resource requirements present substantial barriers to real-time deployment, particularly for resource-constrained edge devices. Third, while federated learning offers a promising pathway for privacy-preserving detection, current frameworks face challenges related to client heterogeneity, communication overhead, and model generalisation. Fourth, the vulnerability of DL-based intrusion detectors to adversarial evasion attacks remains underexplored in power system contexts. Finally, few studies have addressed the integration of detection systems with existing grid control infrastructure or evaluated long-term performance under evolving attack strategies.

Table 1

Comparative of some of the best studies on cyber-attack security intelligence on smart grids.

Study	Models	Dataset	Attack Type	Result	
(Niu et al., 2019)	CNN, LSTM	NSL-KDD from KDD Cup 99	FDI	Accuracy=90%	
(He et al., 2017)	SVE, DLBI	NA	FDI	Accuracy=95.69%.	
(Elsaeidy et al., 2020)	deep CNN	-Soil management node dataset -Environmental monitoring node dataset.	Reply	Accuracy =98.0416% False positive rate =5.6633% Sensitivity =97.6422% Specificity=97.6422% Precision=97.6422%	
(Elsaeidy et al., 2021)	RBM	The environmental dataset, The smart river dataset, and the smart soil dataset.	Reply, DDoS	-High Accuracy rates: 98.37% for the environmental dataset. 98.13% for the smart river dataset. 99.51% for the smart soil dataset.	
(Ali & Li, 2019)	MKL, MKLDR	-The UNB ISXC D 2012 dataset. -The UNSW-NB15 dataset.	DDoS	Accuracy=97%	
(Sayghe et al., 2020)	MLP	NA	Evasion	Accuracy= 87.65% Recall= 80.02 % Precision=100%	
(Siniosoglou et al., 2021)	MENSA, DL	Produced from normal and malicious Modbus/TCP and DNP3 flows	FDI	MENSA: Accuracy=0.8835 TPR= 0.8715 FPR= 0.1134 F1 score= 0.7498.	DL: Accuracy= 0.9646 TPR= 0.7349 FPR= 0.0189 F1 score= 0.7349
(Zhao et al., 2021)	FL	IID and non-IID	FDI	FL with IID: Accuracy=0.9750 Precision=0.9690 Recall=0.9613 F1-score=0.9651.	FL with non-IID: Accuracy=0.9735 Precision=0.9566 Recall=0.9561 F1-score=0.9606.
(Diaba et al., 2023)	CNN, LSTM, GRU	CICIDS-2017 dataset, Friday WorkingHours Afternoon DDoS dataset	DDoS	Accuracy rate of 99.7%.	
(Al-Quraan et al., 2023)	FedTrees framework LSTM, LGBM	Tetouan power consumption dataset	Cyberattacks	LSTM: MAE is 0.02 MAPE 3.04%.	LGBM: MAE 0.017 MAPE 2.69%
(Zhang et al., 2020)	GAN	Practical power system	FDI	Precision=91.17% Recall=92.26%, Accuracy=91.70%	
(VS, 2021)	CNN	NA	FDI	Accuracy=98.31% Precision=99.48% Recall=99.17% F1 Source=99.29%	
(Mohammadpourfard et al., 2021)	CNN, GAF, RP	Empirical load data	Cyber attacks	RP-CNN: Bus 2: Accuracy=0.92% Bus 6: Accuracy=0.96% Bus 43: Accuracy=0.98%	
(Zhang et al., 2024)	SVM, RFs, decision trees, XGBoost, KNN.	Power System Attack Datasets from Mississippi State University and Oak Ridge National Laboratory	Cyber attacks	SVM= 0.54 RFs= 0.95 Decision trees= 0.87 XGBoost= 0.87 KNN= 0.92.	

2.8 Critical analysis of Literature review:

After examining the relevant literature, it is evident that SGs are complex systems that face multiple challenges, requiring them to operate efficiently and securely.

- The objective of these seven studies (Elsaeidy et al., 2021; VS, 2021; Mohammadpourfard et al., 2021; Li et al., 2022; Jithish et al., 2023)) was to address FDI attacks using FL, as demonstrated in (Zhao et al., 2021). Additionally, the studies in (Niu et al., 2019; Zhang et al., 2020; He et al., 2017; Siniosoglou et al., 2021; VS, 2021) examined FDI attacks without FL, with the best outcome observed in (Bus 2=0.92, Bus 6=0.96, Bus 43=0.98).
- Three approaches, referenced as (Ali & Li, 2019; Elsaedy et al., 2021; Diaba et al., 2023) are currently experiencing DDoS attacks. Among these approaches, (Elsaeidy et al., 2021) has shown exceptional performance with high accuracy rates. Specifically, it achieved an accuracy rate of 98.37% for the environmental dataset, 98.13% for the smart river dataset, and 99.51% for the smart soil dataset.

- Two papers, referenced as (Elsaeidy et al., 2020; Elsaedy et al., 2021), encountered reply attacks. Paper (Elsaeidy et al., 2021) achieved exceptional results with high accuracy rates: 98.37% for the environmental dataset, 98.13% for the smart river dataset, and 99.51% for the smart soil dataset.
- In references (Mohammadpourfard et al., 2021; Li et al., 2022), the authors utilized anomaly detection techniques to combat cyberattacks using FL. References (Al-Quraan et al., 2023; Mohammadpourfard et al., 2021) showcase impressive results without FL. Specifically, reference (Mohammadpourfard et al., 2021) reports admirable results with Bus 2 achieving a score of 0.92, Bus 6 achieving a score of 0.96, and Bus 43 achieving a score of 0.98.

The challenges inherent in the implementation of SGs are as follows: the vulnerability of SGs to cyber-attacks presents a significant risk, resulting in a significant increase in operational costs and compromising the overall stability of the system. The challenges in improving cybersecurity in SGs include the accuracy of detecting and identifying cyber-attacks, as well as the computational burden of detecting and identifying attacks and classifying the sources of threats. The reliability of SGs is crucial for their stability. Defects can undermine the dependability of SG systems, and it is vital to tackle the problems related to monitoring, detecting, and categorizing defects in this field. The collection of sensitive data by SGs raises significant privacy concerns. Unauthorized interception or modification of crucial information transmitted through SG systems can violate users' privacy.

The adoption of SG technology has raised concerns regarding potential privacy hazards stemming from the surveillance, consolidation, and examination of consumers' live energy usage data by energy providers. The literature review indicates that there are many issues with SG security. The suggested strategy makes use of federated learning techniques driven by deep learning and machine learning to address these issues. This technology seeks to detect instances of FDI attacks and analyze and classify trends in SG data. The literature review indicates that there are many issues with SG security. The suggested strategy makes use of federated learning techniques driven by deep learning to address these issues. This technology seeks to detect instances of false data injection (FDI) attacks and analyze and classify trends in SG data.

3. Methodology

The literature review indicates that in order to accomplish the goals, more FL techniques need to be used in SG. DL-based FL approaches are used in the suggested method. The objective of this methodology is to detect bogus data injection assaults while identifying and classifying trends in SG data. As demonstrated, there are numerous crucial processes involved in utilizing DL to improve security in SG, as shown in Fig. 1. This section describes the five phases of the approach used in this study: (1) collection of the dataset, (2) local model buildings, (3) simulating federated learning (4) aggregating and averaging local models, (5) revising and disseminating the global model, and (6) model evaluation.

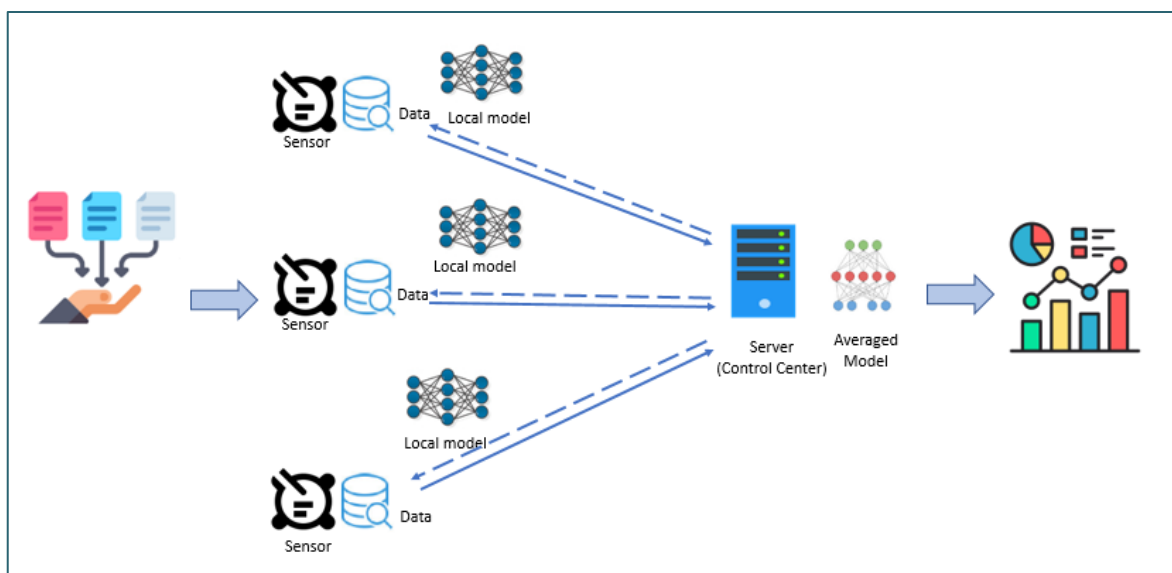


Fig. 1. The research methodology

3.1 Collection of Dataset

Power System Attack Datasets from Oak Ridge National Laboratory and Mississippi State University, particularly the dataset from 4/15/2014 (Power, 2025), are the recommended dataset for this investigation. The FL model will be trained using this dataset. There are three datasets in this collection. This dataset contains three datasets. The dataset is derived from an initial set comprising 15 sets, each containing 78377 samples, 37 power system event scenarios, and 128 features in each scenario. The multiclass datasets are in ARFF format and designed for seamless integration with Weka. The remaining datasets are also

available in CSV format which is compatible with Weka. The 37 scenarios are categorized into natural events (8), no events (1), and attack events (28). The datasets were randomly sampled at one percent and grouped into binary, three-class, and multiclass datasets. By using a variety of samples, FL can improve its accuracy in distinguishing between normal and abnormal behavior.

3.2 Local Models Buildings

In order to attain the best levels of accuracy, precision, recall, and F1-score, we train the models using four recommended DL models for every client. These relate to gaining a more thorough comprehension of a classifier's performance as opposed to depending only on its overall accuracy. Following a comprehensive examination of numerous tests, it was concluded that four models, CNN, LSTM, CNN+LSTM, and MLP, are suitable for datasets and produce noteworthy outcomes. Therefore, it is essential to perform a comparative analysis of models in order to generate better results for FL with SG.

3.2.1 Convolutional Neural Network (CNN)

A One particular kind of machine learning model that belongs to the class of DL techniques is a convolutional neural network (CNN). It is especially good at deciphering visual information. CNNs, sometimes referred to as convnets, use concepts from linear algebra, more especially, convolution operations, to identify patterns and extract characteristics from images. CNNs can be used to process audio and other signal data types, however they are mainly used for image processing. The connection patterns seen in the visual cortex of the human brain serve as the model for the CNN design. The ability to see and comprehend visual information depends on this area of the brain. CNNs are able to handle entire images with ease because they use artificial neurons that are arranged deliberately to interpret visual data. CNNs are widely used in computer vision applications, such as picture recognition and object detection, because of their great effectiveness in object identification. Medical image analysis, facial recognition software, and driverless cars are a few common application scenarios. Unlike CNNs, earlier neural network iterations usually needed to use segmented or lower-resolution input images to analyze visual data in a fragmented manner. With its comprehensive image identification methodology, CNN outperforms a traditional neural network in a variety of image-related tasks and, to a lesser extent, in voice and audio processing (Wu J., 2017).

3.2.2 Long Short-Term Memory (LSTM)

A particular class of recurrent neural networks (RNNs) called long short-term memory (LSTM) networks is able to recognize and preserve long-term associations in sequential input. Sequential data, including time series, text, and speech, can be handled and examined by LSTMs. Information can be selectively retained or discarded as needed thanks to the use of memory cells and gates, which control the flow of information. This successfully avoids the vanishing gradient issue that traditional RNNs frequently face. LSTMs are widely used in a variety of fields, including time series forecasting, audio identification, and natural language processing (Van Houdt et al., 2020).

3.2.3 Mixed Models (CNN+LSTM)

The Compared to either CNN or LSTM alone, the CNN–LSTM hybrid model improves classification accuracy for binary problems via a number of mechanisms:

- Enhanced feature extraction: While the LSTM element understands temporal relationships, the CNN element effectively extracts spatial properties from the input data. Because of this combination, the model may use both temporal and geographical information, which leads to a more complete understanding of the data.
- Compared to individual models, the hybrid model shows higher resilience to noise. Research indicates that while both CNN and LSTM can struggle with high noise levels, the CNN–LSTM hybrid performs better in challenging scenarios. This resilience is particularly useful in real-world applications where data could be hacked or noisy.
- Improved performance metrics: Research shows that the hybrid model consistently achieves higher accuracy rates in a variety of applications.
- The hybrid model can maintain context throughout time thanks to the LSTM's ability to recall previous inputs, which is crucial for tasks where the order of events affects the classification outcome. The model's ability to produce accurate predictions based on trends in historical data is enhanced by this temporal awareness.
- Mechanisms for attention: Some CNN–LSTM hybrid implementations include attention techniques that improve classification performance by allowing the model to focus on relevant features during processing. This improves the capacity to identify crucial components of the data that enable accurate classification.

Combining CNNs and LSTM networks into a hybrid model offers significant advantages for binary classification tasks by combining their skills in temporal sequence learning and spatial feature extraction, which improves accuracy and robustness compared to using either model alone (Lilhore et al., 2023; Abdallah et al., 2021; Shang et al., 2023).

3.2.4 Multilayer Perceptron (MLP) Classifier

A multilayer perceptron (MLP) classifier is an artificial neural network known for its efficacy in handling intricate classification tasks. An MLP classifier comprises multiple layers of interconnected artificial neurons called perceptrons. The typical architecture of these networks consists of an input layer, one or more hidden layers, and an output layer. Every neuron within the network receives input signals, applies a non-linear activation function, and then transmits the transformed output to the subsequent layer. This process persists until reaching the ultimate layer, which generates the classification output.

The training process of an MLP classifier consists of two primary steps: forward propagation and backpropagation. During the process of forward propagation, the input data are transmitted through the network, and the outputs of each neuron are calculated. Subsequently, the computed outputs are contrasted with the precise labels, and an error metric, such as cross-entropy loss, is calculated. During the backpropagation step, the error is retroactively transmitted through the network, and the neurons' weights are modified using gradient descent optimization. This iterative process persists until the network reaches a desirable level of performance (Bikku et al., 2020).

Prior to deploying the model to sensors, the suggested approach seeks to train the dataset. Even though the training process is real-time, our suggested model still has to be validated in practical settings. The parameters from the initial global model are sent to each operational sensor so that local models can be built using them. Using local data from each client, the regional models are first trained. The trained local model is then produced independently for every client.

3.3 Simulating Federated Learning

We used the April 15, 2014 (Power, 2025) Power System Attack Datasets from Oak Ridge National Laboratory and Mississippi State University for this investigation. Data gathering in a central location is not required for FL, based on the information presented in Chapter Two. Rather, the data is stored where the training process takes place, and a global model is developed using the trained models. We split the chosen dataset into multiple segments, each of which represents a distinct client in the SG environment prior to the training phase, in order to recreate this process using the dataset. 80% of Power System Attack datasets are used for training, and 20% are used for testing. The dataset has been split up across 15 clients at the same time. The dataset was then further separated into three and five clients.

3.3.1 Aggregating and Averaging Local Models

Once the models are created locally, sensors work together to convey the local model parameters to the server, allowing them to take part in the federated process without disclosing any personal information. After then, all of the regional models' parameters are gathered throughout the aggregation procedure. The federated average procedure, which creates a new global model by adding the regional models and dividing the total by the number of models received, determines the average based on the aggregate.

3.3.2 Revising and Disseminating the Global Model

Once the average has been calculated, the client's local data is used to create a new global model. The updated global model is distributed to all clients, regardless of their past participation in the learning process, in order to improve the regional models' applicability.

3.4 Model Evaluation

A different set of SG data that was not used for training is used to assess the trained model. The purpose of this evaluation is to determine how well the model classifies unexpected packets as harmful or benign. Metrics from a confusion matrix (CM) are mostly used to assess DL models for classification issues, like the one this research looks at. This analysis's cross-tabulation shows how frequently a model correctly classifies a given data sample with its corresponding accurate label. The model looks for the right kind of data sample. The aforementioned forecast is appropriately acknowledged and contrasted with its factual equivalent. The CM is used to evaluate the model's data categorization accuracy. In the field of anomaly or intrusion detection, a CM can be used to confirm the accuracy of a model's ability to:

- Identifying assaults or anomalies, commonly referred to as True Positives (TP)
- Accurately determine whether regular traffic is present, commonly known as True Negatives (TN).
- False Negatives (FN), which are frequently the result of misinterpreting abnormal traffic as normal, happen.
- Normal traffic may be regarded as abnormal; this is known as False Positives (FP).

All TP and TN would be correctly classified by an ideal model, with no misclassification between the two traffic classifications. Of course, achieving this goal is not possible in practice. Nonetheless, FP and FN rates must be kept to a

minimum. The computation of certain important metrics is made possible by a CM system. The numbers that follow are (Zhang et al., 2020; Wang et al., 2020; Wang et al., 2022):

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

$$False\ Positive\ Rate = \frac{FP}{FP + TN} \quad (2)$$

$$Precision = \frac{TP}{FP + TP} \quad (3)$$

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

$$F1\text{-Score} = 2 * \frac{Recall * Precision}{Recall + Precision} \quad (5)$$

$$Loss = L(y, f(x)) = -[y * \log(f(x)) + (1 - y) * \log(1 - f(x))] \quad (6)$$

where:

- L represents the Binary Cross-Entropy Loss function
- y is the true binary label (0 or 1)
- $f(x)$ is the predicted probability of the positive class (between 0 and 1)

3.5 Experimental Setup

The full experimental configuration to ensure reproducibility can be summarized in Tables 2, 3, and 4. Table 2 reports the federated learning (FL) hyperparameters: the number of global communication rounds RRR, local epochs per round EEE, the client participation fraction CCC, and the local batch size BBB. It also specifies the server aggregation rule (FedAvg by default), the client-sampling policy (uniform without replacement), evaluation cadence, and random seeds, collectively defining the FL protocol and its stochasticity.

Table 2
Federated Learning (FL) Hyperparameters

Parameter	Symbol	Value (default)	Range / Options	Notes
Communication rounds	R	100	50–300	Global aggregations (FedAvg)
Local epochs per round	E	3	1–10	Client-side passes per round
Client fraction per round	C	0.5	0.2–1.0	Fraction of clients sampled each round
Batch size	B	64	32–256	Local mini-batch size
Aggregation	,	FedAvg	FedProx, Scaffold	Server-side update rule
Client sampling	,	Uniform (w/o replacement)	Stratified, Weighted	Sampling policy per round
Evaluation frequency	,	Each round	Every k rounds	Global model validation cadence
Seed(s)	,	3 seeds (e.g., 7, 11, 13)	1–5	For reproducibility

Table 3 details the model architectures used in our study, CNN, LSTM, MLP, and the hybrid CNN–LSTM, listing the layer order and key hyperparameters (e.g., filter counts, kernel sizes, LSTM units, dropout). These definitions standardize capacity across models and clarify how temporal and spatial features are captured; the output layer uses KKK units with softmax, where KKK is the number of classes.

Table 3
Model Architectures (layer-by-layer layer-by-layer specs for CNN, LSTM, MLP, Hybrid CNN–LSTM)

Model	Layer stack (in order)	Hyperparameters	Output layer
CNN	Conv1D → ReLU → MaxPool → Conv1D → ReLU → GlobalAvgPool → Dropout → Dense	Filters: 32, 64; Kernel: 3, 3; Pool: 2; Dropout: 0.3	Dense(units=3, activation=softmax)
LSTM	LSTM → Dropout → LSTM (optional) → Dense	Units: 128 (then 64 optional); Dropout: 0.3	Dense(3, softmax)
MLP	Flatten/FC → Dense → ReLU → Dropout → Dense → ReLU → Dropout → Dense	Hidden units: 256, 128; Dropout: 0.3	Dense(3, softmax)
Hybrid (CNN– LSTM)	Conv1D → ReLU → MaxPool → Conv1D → ReLU → LSTM → Dropout → Dense	CNN filters: 32, 64; kernel: 3; pool: 2; LSTM units: 128; Dropout: 0.3	Dense(3, softmax)

Table 4 provides the training configuration common to centralized and FL settings: optimizer (Adam by default), learning rate and decay, loss function, optional class weighting for imbalance, early-stopping criterion, maximum epochs for the centralized baseline, and hardware/framework notes (mixed precision optional). Together, these tables document the exact choices that govern learning dynamics, model capacity, and optimization, enabling precise replication and fair comparison across models and between centralized and federated rules.

Table 4
Training Configuration

Setting	Value (default)	Alternatives / Range	Notes
Optimizer	Adam	SGD (momentum 0.9), AdamW	$\beta_1=0.9, \beta_2=0.999$
Learning rate	1e-3	5e-5 – 5e-3	Cosine/Step decay optional
Weight decay	1e-5	0 – 1e-3	Use for AdamW/regularization
Loss	Cross-entropy	Focal loss	Class imbalance handling
Class weights	Off (default)	On	If classes are imbalanced
Early stopping	Patience = 10	5–20	Monitor val loss
Max epochs (centralized)	50	30–200	Non-FL baseline training
Hardware	1× GPU (e.g., T4/V100) or CPU	,	Record CUDA/cuDNN versions
Mixed precision	Off	On (fp16)	Depends on hardware
Framework	PyTorch 2.x / TensorFlow 2.x	,	Specify exact version

4. Results And Discussion

After Following that, we use the four recommended DL models for every customer to train the models. We will present and talk about the DL model findings. The four models are the main focus of our attention at the moment. In terms of accuracy, recall, F1-Score, loss, and precision, the suggested approach is shown to be effective by the models that have been suggested. It has been noted that models perform better when working with fewer datasets, like the example of fifteen clients. In the scenario involving fifteen clients, the Mixed Models Classifier produced the best average performance.

In the three-client scenario, federated learning (FL) surpasses the centralized ("No FL") baseline in all models and metrics is displayed in Table 5 and Fig. 2. The hybrid Mix Models classifier achieves the highest ranking, attaining Precision/Recall/F1/Accuracy scores of 0.93/0.89/0.91/0.95 in a federated learning context, compared to 0.89/0.82/0.90/0.90 in a centralized setting, with a reduced loss of 0.07 against 0.09. LSTM ranks second (0.89/0.87/0.88/0.93; loss 0.09) and MLP ranks third (0.84/0.82/0.83/0.86; loss 0.09), both demonstrating steady improvements across centralized training (accuracy +0.05 and +0.06, respectively).

CNN has the lowest performance yet derives the most advantage from federated learning in terms of accuracy (0.86 compared to 0.77, loss 0.20 versus 0.25). The bar chart reflects the table: for each model, the FL bars surpass the No-FL bars in precision, recall, F1, and accuracy, maintaining the ranking (Hybrid > LSTM > MLP > CNN). In this experiment, FL demonstrates superior generalization and reduced loss compared to training on a singular centralized dataset, while having just three clients.

Table 5
The average of all models for centralized vs. baseline FL in the three-client's scenario.

The Model	Precision		Recall		F1-score		Accuracy		Loss		Rank	
	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL
<i>CNN Classifier</i>	0.77	0.68	0.76	0.66	0.77	0.68	0.86	0.77	0.20	0.25	4 th	4 th
<i>LSTM Classifier</i>	0.89	0.74	0.87	0.75	0.88	0.72	0.93	0.88	0.09	0.11	2 nd	2 nd
<i>Mix Models Classifier</i>	0.93	0.89	0.89	0.82	0.91	0.9	0.95	0.9	0.07	0.09	1 st	1 st
<i>MLP Classifier</i>	0.84	0.8	0.82	0.78	0.83	0.79	0.86	0.8	0.09	0.12	3 rd	3 rd

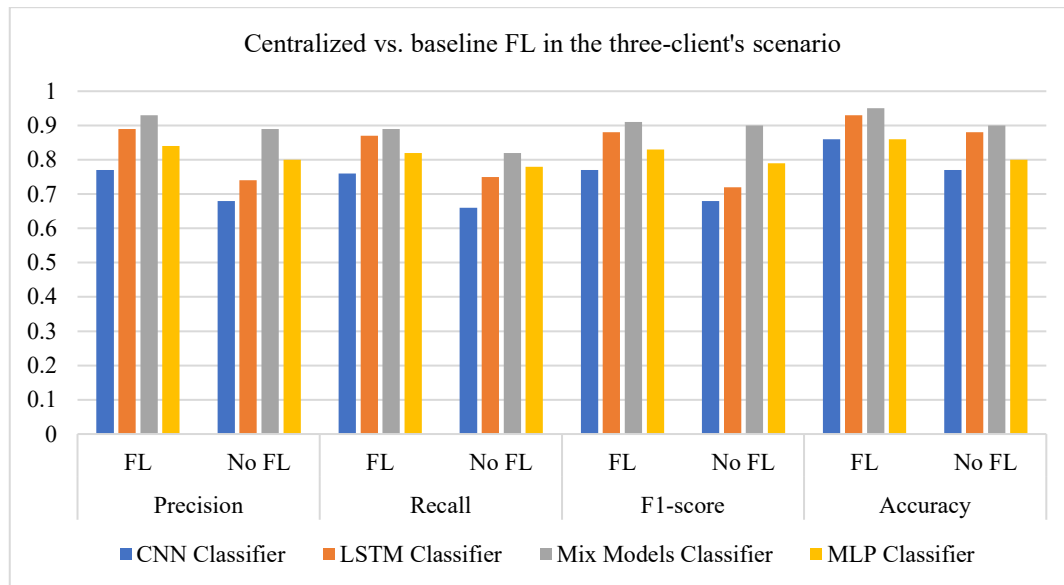


Fig. 2. The average of precision, recall, F1-score, accuracy, and loss for all the models for centralized vs. baseline FL in the three-client's scenario.

In the five-client scenario, illustrated in Table 6 and Fig. 3. FL consistently outperforms the centralized ("No FL") baseline for every model and metric, with uniformly lower losses. The Mix Models classifier remains best overall, Precision/Recall/F1/Accuracy = 0.90/0.89/0.90/0.94 under FL vs. 0.83/0.81/0.82/0.86 centrally, and the lowest loss (0.08 vs. 0.14). LSTM is a close second (0.85/0.85/0.85/0.92, loss 0.12), showing a large loss drop from the centralized run (0.29). CNN improves across the board with FL (0.75/0.64/0.69/0.84, loss 0.32 vs. 0.41), while MLP remains weakest (0.78/0.77/0.77/0.78, loss 0.11), though still better than centralized (0.71/0.70/0.72/0.71, loss 0.21). The bar chart mirrors the table: for precision, recall, F1, and accuracy, FL bars exceed No-FL bars for each model, preserving the ranking (Hybrid > LSTM > CNN > MLP for accuracy/robustness, with MLP last overall).

Table 6

The average of all models for centralized vs. baseline FL in the five-client's scenario.

The Model	Precision		Recall		F1-score		Accuracy		Loss		Rank	
	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL
<i>CNN Classifier</i>	0.75	0.68	0.64	0.6	0.69	0.63	0.84	0.77	0.32	0.41	3 rd	3 rd
<i>LSTM Classifier</i>	0.85	0.79	0.85	0.77	0.85	0.79	0.92	0.86	0.12	0.29	2 nd	2 nd
<i>Mix Models Classifier</i>	0.9	0.83	0.89	0.81	0.9	0.82	0.94	0.86	0.08	0.14	1 st	1 st
<i>MLP Classifier</i>	0.78	0.71	0.77	0.7	0.77	0.72	0.78	0.71	0.11	0.21	4 th	4 th

Furthermore, in the scenario with fifteen clients shown in Table 7 and Fig. 4, FL distinctly outperforms the centralized ("No FL") baseline across all models and metrics, exhibiting consistently lower losses and the most robust outcomes overall among all client counts.

Table 7

The average of all models for centralized vs. baseline FL in the fifteen-client's scenario.

The Model	Precision		Recall		F1-score		Accuracy		Loss		Rank	
	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL
<i>CNN Classifier</i>	0.88	0.81	0.86	0.77	0.87	0.8	0.93	0.87	0.11	0.21	3 rd	3 rd
<i>LSTM Classifier</i>	0.91	0.83	0.88	0.81	0.89	0.81	0.94	0.88	0.10	0.22	2 nd	2 nd
<i>Mix Models Classifier</i>	0.94	0.88	0.92	0.84	0.93	0.86	0.96	0.89	0.05	0.09	1 st	1 st
<i>MLP Classifier</i>	0.89	0.8	0.87	0.79	0.88	0.79	0.91	0.83	0.07	0.10	4 th	4 th

The Mix Models classifier has superior performance with Precision/Recall/F1/Accuracy metrics of 0.94/0.92/0.93/0.96 and a Loss of 0.05, compared to the centralized model's metrics of 0.88/0.84/0.86/0.89 (Loss = 0.09). LSTM ranks second (0.91/0.88/0.89/0.94, Loss = 0.10) and demonstrates a significant improvement over the centralized model (0.83/0.81/0.81/0.88, Loss = 0.22). CNN enhances its performance to 0.88/0.86/0.87/0.93 (Loss 0.11) from a centralized

baseline of 0.81/0.77/0.80/0.87 (Loss 0.21). MLP retains its fourth position while significantly outperforming centralized methods (0.89/0.87/0.88/0.91, Loss 0.07) compared to their performance (0.80/0.79/0.79/0.83, Loss 0.10). The bar chart reflects the table. FL bars consistently surpass No-FL bars in precision, recall, F1, and accuracy, maintaining the hierarchy (Hybrid > LSTM > CNN > MLP) and suggesting that scaling to 15 clients optimizes generalization and dependability.

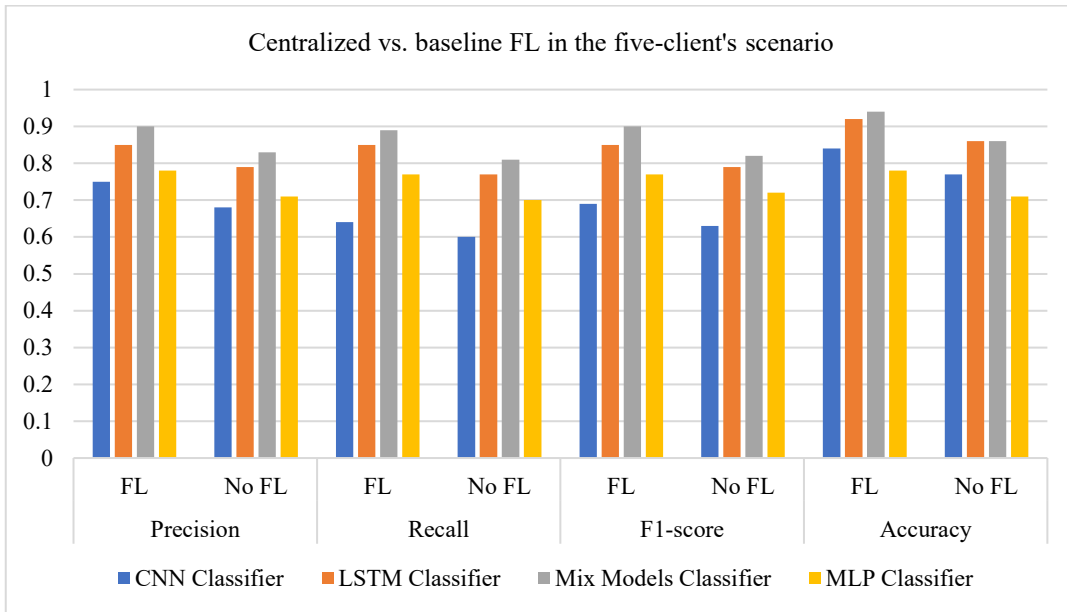


Fig. 3. The average of precision, recall, F1-score, accuracy, and loss for all the models for centralized vs. baseline FL in the five-client’s scenario.

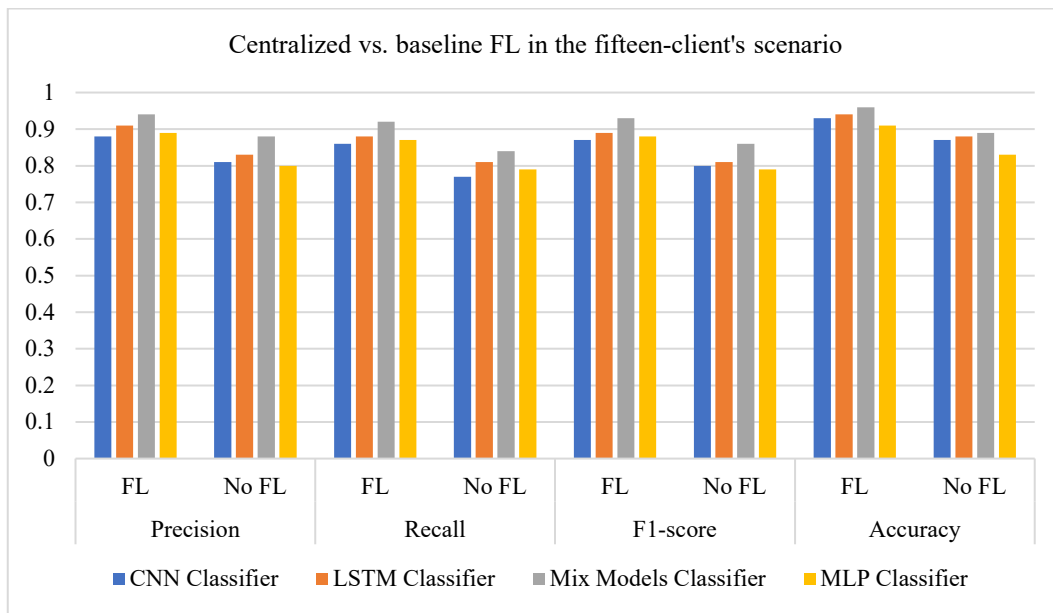


Fig. 4. The average of precision, recall, F1-score, accuracy, and loss for all the models for centralized vs. baseline FL in the fifteen-client’s scenario.

Averaged across all client scenarios, federated learning (FL) outperforms the centralized (“No FL”) baseline for every model and metric, while also reducing loss, as shown in Table 8. The best of precision, recall, F1-score, accuracy, and loss for all the models are illustrated in Figure 5. The Mix Models classifier is best overall, Precision/Recall/F1/Accuracy = 0.91/0.90/0.91/0.95 with Loss = 0.07, beating centralized (0.85/0.82/0.83/0.87, Loss = 0.11). LSTM ranks second (0.87/0.86/0.86/0.93, Loss = 0.10) with a large loss drop from centralized (0.21). MLP is third (0.82/0.80/0.81/0.82, Loss = 0.09) vs. centralized (0.74/0.73/0.74/0.75, Loss = 0.14). CNN remains fourth but still gains noticeably under FL (0.79/0.71/0.75/0.87, Loss = 0.21) compared with centralized (0.72/0.66/0.69/0.80, Loss = 0.29). The bar chart mirrors the table: FL bars are higher for precision, recall, F1, and accuracy across all models, preserving the ranking (Hybrid > LSTM > MLP > CNN).

Overall, FL delivers ~5–8 percentage-point average accuracy gains per model alongside meaningful loss reductions, indicating better generalization and robustness when data remain decentralized. None of the three clients' experiments produced adequate results across all models, according to the condensed tables. Each client provided a large amount of data, which negatively affected the findings and caused them to drop. Out of all the studies conducted in each situation, the accuracy results obtained from five clients were the most satisfying. Furthermore, across all models, fifteen clients consistently produced better results. This confirms that precision, recall, F1-score, and accuracy all improve with an increase in the number of clients.

Table 8

The average of all models in all scenarios for centralized vs. baseline FL.

The Model	Precision		Recall		F1-score		Accuracy		Loss		Rank	
	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL	FL	No FL
<i>CNN Classifier</i>	0.79	0.72	0.71	0.66	0.75	0.69	0.87	0.80	0.21	0.29	4 th	4 th
<i>LSTM Classifier</i>	0.87	0.80	0.86	0.78	0.86	0.80	0.93	0.87	0.10	0.21	2 nd	2 nd
<i>Mix Models Classifier</i>	0.91	0.85	0.9	0.82	0.91	0.83	0.95	0.87	0.07	0.11	1 st	1 st
<i>MLP Classifier</i>	0.82	0.74	0.80	0.73	0.81	0.74	0.82	0.75	0.09	0.14	3 rd	3 rd

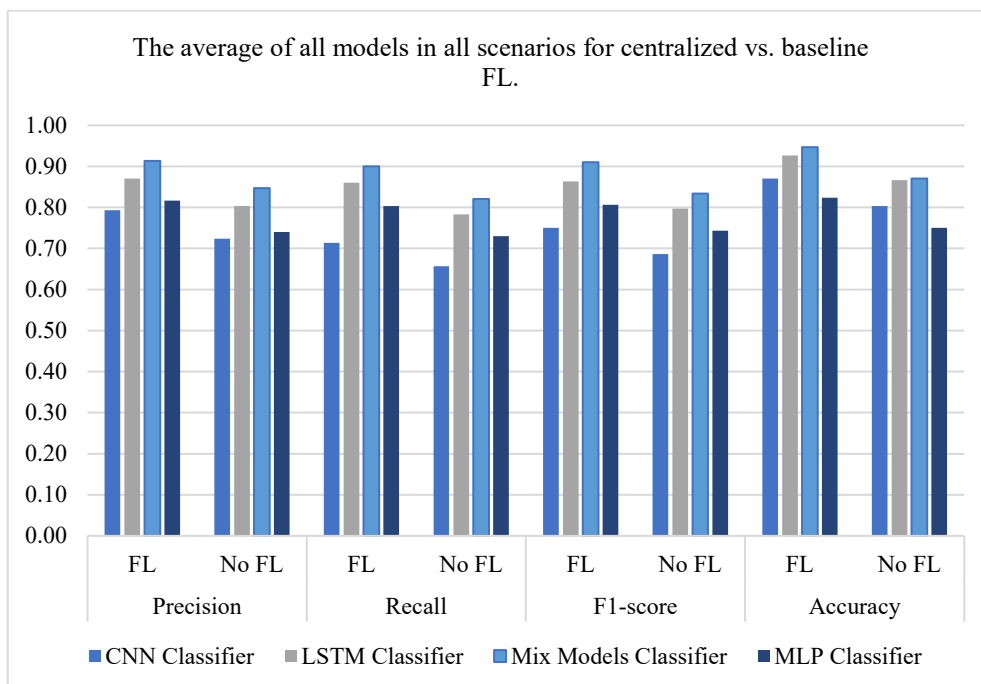


Fig. 5. The best of precision, recall, F1-score, accuracy, and loss for all the models for centralized vs. baseline FL.

The Mixed Models Classifier was the most successful model, based on the average and outcomes tables. For the DL models, the Mix Models yield excellent results.

According to the findings, our paradigm might be applied to situations other than FDI. There are several kinds of situations that can be used:

1. The term "short-circuit fault" describes a circumstance in which two points in a power line are directly connected, disrupting the regular flow of electricity. This can occur at many positions along the line, and a percentage range indicates the precise location.
2. In order to undertake line maintenance, one or more relays on a particular line must be temporarily disabled.
3. A form of attack known as remote tripping command injection occurs when a relay receives a command that causes a breaker to open. Only after an attacker has successfully penetrated the exterior security measures can this action be carried out.
4. Relay configuration modification (Attack): The relays are initially configured using a distance protection mechanism. But to make the relay function useless, the attacker modifies the settings. The relay will thus stop reacting to actual errors or orders.

5. Conclusions And Future Work

Because SGs integrate advanced information and communication technologies (ICT) with traditional electrical power systems, security is a major concern. Although this integration increases efficiency and reliability, it also reveals significant flaws that hackers and other hostile actors could exploit.

Recently, FL has tackled a number of issues brought on by centralized DL. Therefore, the fundamental ideas used in DL should be applied in FL to control issues pertaining to sensitive or personal data. For smart power systems to detect and lessen the effects of security breaches, DL approaches are crucial. SG systems can increase their resistance to a variety of attacks and contribute to the security of SG infrastructure by utilizing the potential of DL algorithms: Predictive analytics, anomaly detection, intrusion detection, protection against hostile machine learning assaults, and blockchain integration

In three, five, and fifteen-client configurations, FL consistently surpasses the centralized (“No FL”) baseline across all models (CNN, LSTM, MLP, and the hybrid Mix/CNN-LSTM) and metrics (Precision, Recall, F1, Accuracy), while producing reduced losses. The hybrid Mix model consistently outperforms all others in every configuration, followed by LSTM, then MLP, with CNN ranked last, this hierarchy remains steady across all figures and tables. Performance enhances with an increasing number of clients, with the 15-client setup yielding the most robust findings and the most significant disparity over the centralized baseline, signifying improved generalization from heightened client diversity. When results are aggregated across all situations, FL consistently outperforms centralized training for each architecture, validating that privacy-preserving FL attains utility comparable to, and frequently exceeding, that of centralized methods without the necessity of aggregating raw data. These findings suggest implementing the hybrid CNN-LSTM in a ≥ 15 -client FL setup to optimize privacy while achieving superior detection accuracy and robustness in smart-grid scenarios.

The availability, integrity, and confidentiality of the electrical infrastructure can be ensured by using DL to strengthen SG systems' resistance to various types of attacks. In order to remedy the current shortcoming, the study used the FL on SG. We performed our investigation over three client divisions: three, five, and fifteen, using a total of four DL models: CNN, LSTM, Mixed models, and MLP. When used by the fifteen clients, the Mixed models produced the most remarkable and accurate results, with an accuracy rate of 98%. As far as we are aware, no previous study used mixed models as a model to handle the SG attacks. We intend to carry out the following initiatives in the future to enhance our models and experiments:

- To increase the research' accuracy, we want to bring the clientele up to thirty or fifty. This is due to the fact that accuracy would be enhanced by having more samples from datasets.
- Implement more DL models. in order to ascertain the best results that may be derived from the dataset used in FL.
- Use multiple optimizers and evaluate each one's performance.
- Apply every research model to a different dataset with more samples.
- Integration of blockchain technology to improve control over sensitive or personal data issues.

Acknowledgments

The authors gratefully acknowledge Qassim University, represented by the Deanship of Graduate Studies and Scientific Research, on the financial support for this research under the number (QU-J-PG-2-2025-52682) during the academic year 1446 AH / 2024 AD.

References

- Abdallah, M., An Le Khac, N., Jahromi, H., & Delia Jurcut, A. (2021, August). A hybrid CNN-LSTM based approach for anomaly detection systems in SDNs. *In Proceedings of the 16th International Conference on Availability, Reliability and Security* (pp. 1-7). <https://doi.org/10.1145/3465481.3469190>
- Ali, S., & Li, Y. (2019). Learning multilevel auto-encoders for DDoS attack detection in smart grid network. *IEEE Access*, 7, 108647-108659. <https://doi.org/10.1109/ACCESS.2019.2933304>.
- Al-Quraan, M., Khan, A., Centeno, A., Zoha, A., Imran, M. A., & Mohjazi, L. (2023). FedraTrees: A novel computation-communication efficient federated learning framework investigated in smart grids. *Engineering Applications of Artificial Intelligence*, 124, 106654. <https://doi.org/10.1016/j.engappai.2023.106654>
- Bikku, T. (2020). Multi-layered deep learning perceptron approach for health risk prediction. *Journal of Big Data*, 7(1), 50. <https://doi.org/10.1186/S40537-020-00316-7>
- Diaba, S. Y., & Elmusrati, M. (2023). Proposed algorithm for smart grid DDoS detection based on deep learning. *Neural Networks*, 159, 175-184. <https://doi.org/10.1016/j.neunet.2022.12.011>
- Drayer, E., & Routtenberg, T. (2019). Detection of false data injection attacks in smart grids based on graph signal processing. *IEEE Systems Journal*, 14(2), 1886-1896. <https://doi.org/10.1109/JSYST.2019.2927469>
- Elsaeid, A. A., Jagannath, N., Sanchis, A. G., Jamalipour, A., & Munasinghe, K. S. (2020). Replay attack detection in smart cities using deep learning. *IEEE Access*, 8, 137825-137837. <https://doi.org/10.1109/ACCESS.2020.3012411>

- Elsaedy, A. A., Jamalipour, A., & Munasinghe, K. S. (2021). A hybrid deep learning approach for replay and DDoS attack detection in a smart city. *IEEE Access*, 9, 154864-154875. <https://doi.org/10.1109/ACCESS.2021.3128701>
- He, Y., Mendis, G. J., & Wei, J. (2017). Real-time detection of false data injection attacks in smart grid: A deep learning-based intelligent mechanism. *IEEE Transactions on Smart Grid*, 8(5), 2505-2516. <https://doi.org/10.1109/TSG.2017.2703842>
- Hu, P., Gao, W., Li, Y., Wu, M., Hua, F., & Qiao, L. (2023). Detection of false data injection attacks in smart grids based on expectation maximization. *Sensors*, 23(3), 1683. <https://doi.org/10.3390/S23031683>
- Jithish, J., Alangot, B., Mahalingam, N., & Yeo, K. S. (2023). Distributed anomaly detection in smart grids: a federated learning-based approach. *IEEE Access*, 11, 7157-7179. <https://doi.org/10.1109/ACCESS.2023.3237554>
- Li, Y., Wei, X., Li, Y., Dong, Z., & Shahidehpour, M. (2022). Detection of false data injection attacks in smart grid: A secure federated deep learning approach. *IEEE Transactions on Smart Grid*, 13(6), 4862-4872. <https://doi.org/10.1109/TSG.2022.3204796>
- Lilhore, U. K., Dalal, S., Faujdar, N., Margala, M., Chakrabarti, P., Chakrabarti, T., & Velmurugan, H. (2023). RETRACTED ARTICLE: Hybrid CNN-LSTM model with efficient hyperparameter tuning for prediction of Parkinson's disease. *Scientific Reports*, 13(1), 14605. <https://doi.org/10.1038/s41598-023-41314-y>
- Lin, X., An, D., Cui, F., & Zhang, F. (2023). False data injection attack in smart grid: Attack model and reinforcement learning-based detection method. *Frontiers in Energy Research*, 10, 1104989. <https://doi.org/10.3389/FENRG.2022.1104989/BIBTEX>
- Mirzaee, P. H., Shojafar, M., Cruickshank, H., & Tafazolli, R. (2022). Smart grid security and privacy: From conventional to machine learning issues (threats and countermeasures). *IEEE Access*, 10, 52922-52954. <https://doi.org/10.1109/ACCESS.2022.3174259>
- Mohammadpourfard, M., Genc, I., Lakshminarayana, S., & Konstantinou, C. (2021, October). Attack detection and localization in smart grid with image-based deep learning. In *2021 IEEE international conference on communications, control, and computing technologies for smart grids (SmartGridComm)* (pp. 121-126). IEEE. <https://doi.org/10.1109/SmartGridComm51999.2021.9631994>
- Niu, X., Li, J., Sun, J., & Tomsovic, K. (2019, February). Dynamic detection of false data injection attack in smart grid using deep learning. In *2019 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)* (pp. 1-6). IEEE. <https://doi.org/10.1109/ISGT.2019.8791598>
- Power system. Accessed: August. 26, 2025. [Online]. Available: <https://www.kaggle.com/datasets/bachirbarika/power-system>
- Sayghe, A., Zhao, J., & Konstantinou, C. (2020, August). Evasion attacks with adversarial deep learning against power system state estimation. In *2020 IEEE Power & Energy Society General Meeting (PESGM)* (pp. 1-5). IEEE. <https://doi.org/10.1109/PESGM41954.2020.9281719>
- Shang, L., Zhang, Z., Tang, F., Cao, Q., Pan, H., & Lin, Z. (2023). CNN-LSTM hybrid model to promote signal processing of ultrasonic guided lamb waves for damage detection in metallic pipelines. *Sensors*, 23(16), 7059. <https://doi.org/10.3390/S23167059>
- Siniosoglou, I., Radoglou-Grammatikis, P., Efstathopoulos, G., Fouliras, P., & Sarigiannidis, P. (2021). A unified deep learning anomaly detection and classification approach for smart grid environments. *IEEE Transactions on Network and Service Management*, 18(2), 1137-1151. <https://doi.org/10.1109/TNSM.2021.3078381>
- Van Houdt, G., Mosquera, C., & Nápoles, G. (2020). A review on the long short-term memory model. *Artificial intelligence review*, 53(8), 5929-5955.
- VS, A. (2021). Multi Label Deep Learning classification approach for False Data Injection Attacks in Smart Grid. *KSII Transactions on Internet & Information Systems*, 15(6). <https://doi.org/10.3837/tiis.2021.06.013>
- Wang, Q., Ma, Y., Zhao, K., & Tian, Y. (2022). A comprehensive survey of loss functions in machine learning. *Annals of Data Science*, 9(2), 187-212. <https://doi.org/10.1007/s40745-020-00253-5>
- Wang, S., Bi, S., & Zhang, Y. J. A. (2020). Locational detection of the false data injection attack in a smart grid: A multilabel classification approach. *IEEE Internet of Things Journal*, 7(9), 8218-8227. <https://doi.org/10.1109/JIOT.2020.2983911>
- Wu, J. (2017). Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. *Nanjing University. China*, 5(23), 495.
- Zhang, S., Shi, P., Du, T., Su, X., & Han, Y. (2024). Threat attribution and reasoning for industrial control system asset. *International Journal of Ambient Computing and Intelligence (IJACI)*, 15(1), 1-27. <https://doi.org/10.4018/IJACI.333853>
- Zhang, Y., Wang, J., & Chen, B. (2020). Detecting false data injection attacks in smart grids: A semi-supervised deep learning approach. *IEEE Transactions on Smart Grid*, 12(1), 623-634. <https://doi.org/10.1109/TSG.2020.3010510>
- Zhao, L., Li, J., Li, Q., & Li, F. (2021). A federated learning framework for detecting false data injection attacks in solar farms. *IEEE Transactions on Power Electronics*, 37(3), 2496-2501. <https://doi.org/10.1109/TPEL.2021.3114671>

