

ARTICLE

Explainable Federated Learning and Blockchain-Based Privacy Preserving Fault Identification Framework for Reliable Energy Distribution in Smart Grids

Lakshit Pathak¹, Khushi Shah², Vidhi Ruparelia³, Bansri Choksi⁴, Rajesh Gupta⁵, Sudeep Tanwar⁵, Ming-Wei Li⁶, Wei-Chiang Hong^{6,*} and Svetlana Sushkova⁷

¹Tatvic Digital Analytics Pvt. Ltd., Ahmedabad, Gujarat, India

²University of Maryland Baltimore County, Baltimore, MD, USA

³University of Southern California, Los Angeles, CA, USA

⁴Embedded Software Engineering Department, Cyient, Bangalore, India

⁵Department of Computer Engineering—AI & Big Data, Marwadi University, Rajkot, Gujarat, India

⁶College of Shipbuilding Engineering, Harbin Engineering University, Harbin, China

⁷Southern Federal University, Rostov-on-Don, Russia

*Corresponding Author: Wei-Chiang Hong. Email: samuelsonghong@gmail.com

Received: 16 July 2026; Accepted: 18 September 2026; Published: 21 October 2026

ABSTRACT: To enable smart grid automation, proactive load balancing, and dependable electric vehicle (EV) charging infrastructure, robust optical communication networks are essential for high-speed, deterministic, and low-latency telemetry transmission. Physical impairments in optical links (such as micro-bending, dirty connectors, poor splices, and fiber cuts) do not interfere with electrical current flow directly, but instead induce substantial attenuation, packet loss, and latency spikes that disrupt Supervisory Control and Data Acquisition (SCADA) telemetry monitoring, Phasor Measurement Unit (PMU) synchronization, and IEC 61850 GOOSE protection signaling. To address these vulnerabilities without compromising the privacy of raw substation data, this paper proposes an Explainable Federated Learning (X-FL) framework integrated with a hybrid Blockchain and InterPlanetary File System (IPFS) architecture. Deep sequence models (RNN, GRU, and LSTM) are deployed and optimized using the Adam, RMSprop, and AdamW algorithms. Explainable AI (XAI) techniques (SHAP and LIME) are integrated to provide transparent decision boundaries, substantiated by quantitative Infidelity (*INFD*) and Sensitivity (*SENS*) metrics. To overcome the gas limits of the Ethereum blockchain, high-dimensional model weights are stored off-chain on IPFS, while a Solidity smart contract handles 46-character Content Identifier (CID) indexing, access control, and immutable fault logging, while global aggregation and Byzantine screening execute securely within an off-chain orchestration environment. Extensive evaluations under Dirichlet non-IID data skew ($\alpha \in \{1.0, 0.5, 0.1\}$) and Byzantine poisoning attacks demonstrate that the Adam-optimized LSTM achieves an optimal 87.16% accuracy while robustly tolerating up to 30% adversarial clients without false rejection of benign nodes.

KEYWORDS: Smart grid; electric vehicles; optical fibers; federated learning; blockchain; smart contract; fault detection; energy distribution

1 Introduction

The smart grid revolutionizes the traditional process of distributing electricity and turns it into a controlled, scalable, resilient, and well-coordinated cyber-physical network. There exists real-time monitoring of energy usage, anticipatory load management, and decentralized scheduling for distributed energy resources

(DER), including charging systems for electric vehicles (EV) that are in private and corporate use. However, the success of these cutting-edge systems relies heavily on the optical fiber communication network that provides reliable service. The optical fiber communication system makes bi-directional communications possible between power sources, substations, and smart customers so that it becomes possible to provide Demand-response programs as well as quick network fault identification and optimal route planning of the energy flow.

Nonetheless, the sophisticated functions and autonomy of smart grid systems do not minimize the difficulties in providing stable and continuous power flows during times of peak surge demand. Due to the complex interrelated designs of smart grids, they are more at risk for a variety of faults that may severely impact grid stability and outputs to consumers. These faults may include any number of issues, such as short-circuits, faults on the line, failure of equipment, and unbalanced voltage, that can interrupt power flows seamlessly. These issues represent significant threats to grid efficiency and consumer reliability during times of peak demand. Novel fault detection mechanisms represent unique bridges to solving these challenges.

Optical fiber communication, which forms an essential signaling backbone in smart grids, faces many physical impairments, such as macro-bending, micro-bending, dirty connectors, faulty fusion splices, physical fiber cuts, and covert optical tapping. These physical defects reduce the optical signal-to-noise ratio (SNR) and cause localized attenuation of optical power. Although optical fiber does not carry electrical energy, degradation of the optical channel induces severe telemetry packet dropouts ($\mathbb{P}(L_p) > 0.15$) and inflates communication latencies beyond critical protection timing thresholds ($\tau_{\text{thresh}} = 4 \text{ ms}$). This blinds automated digital protection relays and delays circuit-breaker opening during short-circuit overcurrent events, creating the risk of cascading blackouts across interconnected EV charging stations. Thus, quick and precise determination of localized faults in optical fibers is imperative to guarantee the protection integrity of power systems.

There has been research focused on fault detection in both optical fiber networks and smart grids. Xu and Yuan [1] proposed the wavelet transform in concert with Optical Time-Domain Reflectometry (OTDR) with an improvement of 9.8% in the accuracy of fault detection. Abdelli et al. [2] used an LSTM-based time-domain reflectometry model, which was reported to be a better fault localization methodology than standard means. Similarly, Usman et al. [3] researched Fiber Bragg Grating (FBG) sensors in G-PON systems with improved fault detection and reliability in risk of transmission. The research did not limit itself to optical networks but also included exploration into optimizing energy transmission in electric vehicle (EV) charging and smart grids. Mozafar et al. [4] utilized a hybrid method combining Genetic Algorithm and Particle Swarm Optimization (GA-PSO) for the placement of EV charging stations while minimizing power losses and increasing voltage stability. Das et al. [5] studied Grid-to-Vehicle (G2V) and Vehicle-to-Grid (V2G) smart charging strategies to develop an efficient load balancing and cost-minimizing system. Also, Carcangiu et al. [6] provided an examination of Power Line Communication (PLC) to enable EV-grid data exchange in order to minimize energy storage and optimize operations. In the fault detection framework, Mahfouz and El-Sayed [7] augmented boost converters with VLSI (very-large-scale-integration) techniques to improve overall digital protection and increase reliability in a smart grid environment. In [8,9], Pathak et al. proposed fault detection methods for optical cables in smart grids using several DL models such as LSTM, RNN, 1D CNN, and MLP with various optimizers and Explainable AI techniques. Njoku et al. [10] proposed the use of SHAP and LIME with deep neural networks and LSTM models for EV battery management and to improve explainability with reliability. Khaneghah et al. [11] worked on fault detection and diagnosis (FDD) to improve operational efficiency in the battery and motor systems of EVs. Lastly, Lucas et al. [12] implemented a wavelet-based feature extraction method with neural networks to facilitate an efficient fault detection/identification method for distributed energy generation systems. These studies demonstrate the

necessity for additional advanced fault detection approaches to advance the reliability and productivity of an optical fiber network, EV, and smart grid systems.

The state-of-the-art techniques involve Artificial Intelligence (AI) and Machine Learning (ML) models that operate through the cloud or in a centralized fashion for fault detection and classification processes. Use of the method leads to enhanced performance while creating major challenges that affect system latency, along with network congestion and data privacy issues. The enormous quantity of real-time data flows from EV charging networks with smart grids creates significant difficulties when data is sent to central servers because network bottlenecks and power usage issues trigger security problems. The restriction on the transparency offered by the cloud-operated AI models makes it difficult for users to understand decisions, recognize biases, and be compliant with the regulatory requirements.

FL has appeared as a significant solution to tackle these issues. FL eliminates this step as it allows the training of AI models to occur at the edge or directly on local devices or nodes without transmitting raw data to far-off servers. The utilization of this method provides enhanced privacy and security along with lower data transmission needs, and it enables systems to scale well with large smart grid infrastructure fault detection requirements. FL allows local models to send updates instead of entire datasets, so grid data stays protected at its source while taking part in overall model enhancement. The authors developed “FedDetect” as their proposed federated learning framework, which is presented in [13]. The model utilizes differential privacy rules on local datasets and combines this mechanism with a deep learning architecture based on temporal convolutional networks. The implemented model yielded accurate fault detection outcomes together with reduced implementation expenses. Authors in [14] suggested using a federated transformer-based learning model for false data injection attack prevention. A combination of the proposed model with the Paillier cryptosystem delivered efficient results during training on IEEE-14 and 118-bus systems. In [15], Huong et al. presented a “FedEx” architecture for anomaly detection in IoT-based systems. XAI enabled the model to operate at high speed with minimal memory requirements and hardware needs. A blockchain-powered federated learning solution enabled device failure detection in IIoT devices, as described in [16]. The execution of the weighted averaging algorithm produced results of acceptable accuracy and visibility. The FedAVG aggregated algorithm from [17] addressed data imbalance issues in smart grids successfully by implementing weighted data fault distribution until accuracy increased by 39.7% along with precision improvement by 40.4%.

Through XAI methods, SHAP and LIME, users and policymakers receive interpretive assessment results that enable better operational transparency since these systems strengthen detection capabilities. XAI delivers system audit assistance that enhances regulatory compliance and allows stakeholders to build confidence in system operations by revealing AI predictions. The integration of blockchain with smart grids enables fault detection systems because this technology ensures the secure distribution of sensor and smart meter data using decentralized ledgers, which maintain permanent data records. The authors in [18] combined BEST with DRIP for signal transmission protection, and [19] integrated blockchain technology for anomaly detection through sensors and machine learning algorithms. For fault detection systems having trusted data, improving the operating efficiency is essential for achieving good security networks using federated learning. Operational efficiency provides the organizational ability to adjust in different circumstances because it boosts situational adaptability. FL trains distributed models across entities through decentralized methods that do not require centralized data accumulation, while blockchain enables secure fault logging that is tamper-resistant, together with transparent information exchange. XAI techniques that utilize SHAP and LIME achieve better model interpretability that assists in developing more transparent and reliable fault detection methods. Technical integration between power quality measurement and phasor

measurement unit systems creates improved fault detection efficiency and lower power losses alongside strengthened smart grid stability for interconnected systems.

Motivated by this, the paper introduces an innovative framework that integrates Blockchain technology with Explainable Federated Learning (X-FL) to achieve a secure, transparent, and reliable energy system through effective fault detection in smart grids. This framework takes advantage of the distributed nature of Federated Learning (FL), the interpretability of Explainable AI (XAI), and the trust and immutability of Blockchain to enable efficient, privacy-preserving, and secure fault detection through decentralized grid components. It is anticipated that this will increase model interpretability, further create a secure channel of data among nodes, and ultimately lead to more reliable energy delivery. In totality, this system seeks to achieve a resilient smart grid that is capable of reacting to faults and ultimately maintaining continuous and uninterrupted power delivery.

To provide an analytical baseline across these heterogeneous paradigms, a structured summary of core objectives and methodologies found in contemporary domain literature is consolidated below in [Table 1](#).

Table 1: Comparative analysis of the existing state-of-the-art works.

Reference	Focus Area	Methodology	Key Findings
Xu and Yuan [1]	Optical fiber fault detection	Wavelet transform with Optical Time-Domain Reflectometry (OTDR)	9.8% accuracy improvement in fault detection.
Abdelli et al. [2]	Optical fiber fault detection	LSTM model based on time-domain reflectometry	Better fault localization accuracy over traditional models.
Usman et al. [3]	Optical fiber fault detection	Fiber Bragg Grating (FBG) sensors in G-PON	Improved fault detection and transmission reliability.
Mozafar et al. [4]	Renewable energy and EV charging	Hybrid GA-PSO algorithm for EV charging station placement	Minimized power losses and improved voltage stability.
Das et al. [5]	EV charging optimization	Grid-to-Vehicle (G2V) and Vehicle-to-Grid (V2G) smart charging	Efficient load balancing and cost minimization.
Carcangiu et al. [6]	Smart grid communication	Power Line Communication (PLC) for EV-grid data exchange	Improved energy storage optimization and seamless data transfer.
Mahfouz and El-Sayed [7]	Smart grid fault detection	Boost converters with VLSI-based fault detection	Enhanced digital protection and real-time reliability.
Njoku et al. [10]	EV battery management	SHAP and LIME applied to DNN and LSTM	Improved explainability and reliability in battery management.
Khaneghah et al. [11]	Fault detection in EV systems	Fault detection and diagnosis (FDD) for battery and motor systems	Enhanced fault identification and operational efficiency.

(Continued)

Table 1 (continued)

Reference	Focus Area	Methodology	Key Findings
Lucas et al. [12]	Fault detection in distributed generation	Wavelet-based feature extraction with neural networks	Efficient fault detection and operational reliability.

1.1 Novelty

The proposed framework proposes the establishment of a secure coordinated cross-layer cyber-physical system, which is different from traditional centralized or federated networks that suffer bandwidth limitations and become victims of single points of failure and adversarial poisoning of models in smart grid infrastructure. The essence of this architecture comprises three important technical innovations:

- A unified cross-layer cyber-physical design has been developed for quantifying how optical power attenuation, backscatter losses, and reflection irregularities have a direct and substantial effect on the availability of the SCADA system, PMU synchronization, and IEC 61850 GOOSE messaging, while making sure that the optical communication systems are strictly separated from physical electrical power transfer systems.
- A hybrid Blockchain-IPFS coordination protocol that separates computation from storage has been proposed. Lightweight 46-character Content Identifiers (CIDs) are anchored on-chain in a secure, immutable way, while high-dimensional neural network weight matrices remain off-chain to overcome block gas limit constraints.
- A validation-driven Byzantine defense loop executed during off-chain aggregation of local parameters has been developed. It evaluates the consistency of parameter updates based on an L_2 -norm validation loss divergence metric, allowing up to 30% of malicious poisoning updates to be filtered out without penalizing genuine substations, even under severe Dirichlet non-IID label skew.

Optical fibers do not transmit electric current but function solely as an essential communication material in transmitting IEC 61850 GOOSE trip signals, SCADA control commands, and PMU synchrophasors. Consequently, the physical impairments of optics will increase communication delays and packet losses, jeopardizing the clearing of electrical shorts. The suggested framework directly addresses this problem with accurate, privacy-preserving, and explainable diagnosis of optical faults.

1.2 Research Contributions

The significant contributions of this article are summarized as follows:

- We propose an Explainable Federated Learning (X-FL) framework for decentralized, privacy-conserving classification of faults in optical fiber, which retains the privacy of raw OTDR telemetry data.
- We carry out a comparative assessment of recurrent deep learning approaches (LSTM, GRU, RNN) combining several optimization algorithms (Adam, RMSprop, and AdamW) and aggregation methods for federated learning processes using non-IID Dirichlet data distributions ($\alpha \in \{1.0, 0.5, 0.1\}$) allowing us to determine convergence bounds that represent the best-case scenario.
- Implementation of SHAP and LIME approaches allows for examining the physical decision boundaries (loss, reflectance, SNR) supported by particular numerical indicators of quantitative Infidelity (*INFD*), Sensitivity (*SENS*), and Expected Calibration Error (*ECE*).

- We create and implement a hybrid architecture of the Solidity smart contract and IPFS decentralized storage system that realizes lightweight indexing on the blockchain (43,218 gas), tamper-proof data logging, and off-chain validation that can be resistant to 30% of Byzantine attacks.

1.3 Paper Outline

The rest of the paper is organized as follows. The system model and problem formulation is elucidated in [Section 2](#). The proposed framework is shown in [Section 3](#). Results are shown in [Section 4](#). [Section 5](#) presents the conclusion and future work.

2 System Model and Problem Formulation

A Blockchain and eXplainable Federated Learning (X-FL)-based fault detection framework is proposed for smart grids, enabling secure, reliable, and interpretable energy distribution. At each substation S_i , an Optical Time-Domain Reflectometer (OTDR) monitors the optical fiber network, collecting data points $D_{Si} = \{d_1, d_2, \dots, d_k\}$ that capture signal variations, attenuation, and transmission characteristics. A local model C_{fi} is trained at each substation to predict the fault type $\hat{F}_m$ for fiber segments F_m .

Instead of transmitting raw OTDR data, local model training is performed to preserve privacy and reduce communication overhead. Model weight updates are maintained off-chain on the IPFS, and their unique CIDs are registered to a global aggregator on the blockchain, allowing centralized model enhancement without compromising the confidentiality of the data.

Through blockchain technology attached to its updates, the authenticity and transparency of an engineering model metadata file's lineage and the diagnostic logs can be assured. Each 46-character update CID that appears in the file will be stored securely on the blockchain system, therefore ensuring that the model's development history is tamper-proof, traceable, and secure. Performance metrics are also recorded to enable auditable improvements over time.

The system optimizes fault detection using explainable AI techniques like SHAP and LIME. The objective is defined as:

$$F = \max (A - \Lambda L) \quad (1)$$

where A is overall model accuracy, L is the cumulative loss across local models, and Λ is a regularization parameter. A Solidity smart contract is responsible for managing the permissioning of clients and maintaining immutable fault records, while an off-chain orchestrator carries out the processing of model updates obtained from IPFS. Together, X-FL and blockchain enhance the scalability, security, and reliability of fault detection in smart grids.

To measure the local prediction error of the model before federating communication, each substation computes the variance of a parameter of the structural mean square, which is defined as:

$$E_i = \frac{1}{k} \sum_{j=1}^k (d_j - \hat{d}_j)^2 \quad (2)$$

where d_j represents the observed OTDR signal trace metric from dataset D_{Si} and $\hat{d}_j$ denotes the corresponding baseline value predicted by the local model C_{fi} . Within the multi-substation network, localized gradient updates and weight variations are bounded via the parameter distribution constraint formalized as:

$$\sum_{i=1}^S \omega_i = 1, \quad \text{where } \omega_i \geq 0 \quad (3)$$

Here, ω_i represents the proportional weight coefficient that every substation has depending on how much data it contributes and how many samples it has within the smart grid infrastructure.

To come up with a dimensionally consistent, physically valid formulation for the coupling of cyber-physical grids without making any assumptions about the electrical properties of fibers, we introduce the dimensionless index of Control and Telemetry Availability $\mathcal{A}_{\text{telemetry}}(t)$ taking values from the interval $[0, 1]$. Contemporary smart grid distribution systems utilize fiber-optic communication links exclusively to convey IEC 61850 GOOSE protective messages, SCADA control commands, and time-synchronized readings of PMUs. The degradation of the optical channel leads to increasing signal attenuation and is responsible for packet loss greater than $(\mathbb{P}(L_p) > 0.15)$, thus leading to communication delay $\tau_{\text{comm}}(t)$ exceeding the maximum allowable trip latency ($\tau_{\text{thresh}} = 4$ ms). The expression for the telemetry availability index is as follows:

$$\mathcal{A}_{\text{telemetry}}(t) = \mathbb{P}(\tau_{\text{comm}}(t) \leq \tau_{\text{thresh}}) \cdot R_{\text{opt}}(t) \quad (4)$$

where $\tau_{\text{thresh}} = 4$ ms is the maximum permitted delay time for protective breaker tripping according to IEC 61850 and $\mathbb{P}(\tau_{\text{comm}}(t) \leq \tau_{\text{thresh}})$ is the probability of successful delivery of a packet throughout the time period. The relative reliability of the optical channel $R_{\text{opt}}(t) \in [0, 1]$ is defined as:

$$R_{\text{opt}}(t) = \exp\left(-\left[\frac{\alpha_{\text{fiber}} L_{\text{fiber}}}{\alpha_{\text{ref}}} + \sum_{j=1}^M \beta_j F_j(t)\right]\right) \quad (5)$$

Here, α_{fiber} refers to fiber loss (dB/km), L_{fiber} is the distance of the optical span (km), α_{ref} is the normalizing power level (dB), and $F_j(t) \in \{0, 1\}$ indicates whether optical fault type j is active or not. Also, β_j represents the degradation factor. In addition, when optical faults degrade $\mathcal{A}_{\text{telemetry}}(t)$, localized OTDR sequence inference performed at the substation edge under an inference delay of $\tau_{\text{infer}} \approx 185$ ms allows SCADA controllers to reroute telemetry packets through back-up optical links before relay protection is affected.

3 Proposed Framework

The proposed framework (as shown in Figs. 1 and 2) is organized into four subsections: the smart grid layer, the federated learning layer, the Explainable AI layer, and the blockchain layer. Each layer is described as follows.

3.1 Smart Grid Layer

The cyber-physical framework of the smart grid merges the high-voltage distribution system (made of metallic power conductors) with the high-speed optical communication system. It is important to note that electricity is transmitted exclusively by metallic conductors while optical fibers are used for transmitting SCADA control commands, PMU synchrophasors, and GOOSE protection messages according to IEC 61850. The efficiency of electricity delivery relies entirely on the strength of optical communications. For example, under instances of electric overload or short-circuiting, GOOSE messages must be sent through the optical system in $(\mathbb{P}(L_p) \leq 0.15)$ so that protective switches can operate properly. If there is an optical problem (for example, macro-bending, contamination of connectors, or problems with fusion splicing), signal attenuation may occur that causes $(\mathbb{P}(L_p) > 0.15)$ which in turn leads to delays that are longer than 4 ms. Thus, the digital protection device is placed in the mode of “communication-blind” that hinders its ability to operate properly and causes the risk of serious damage to the equipment, transformer fire, and chain outages.

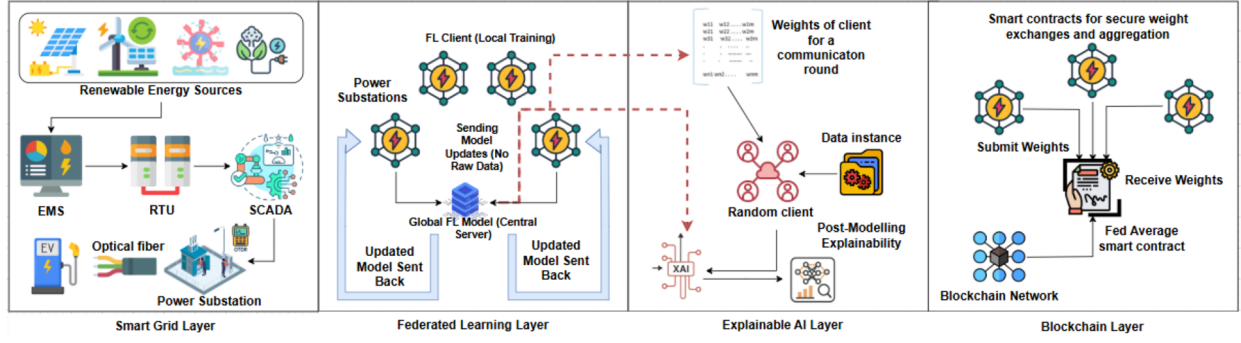


Figure 1: Proposed framework.

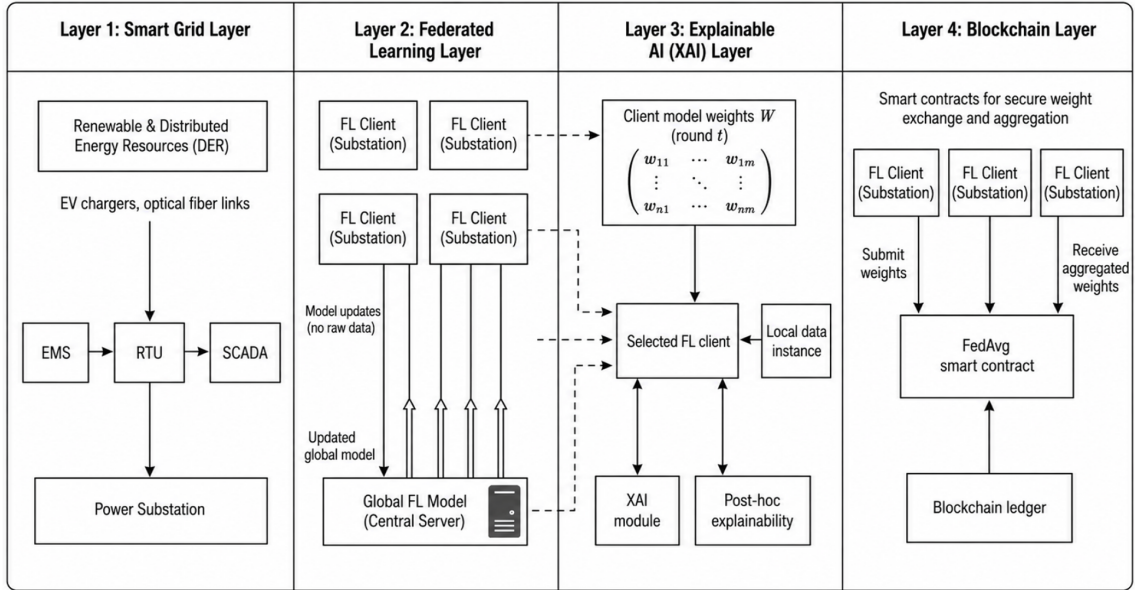


Figure 2: Four-layer system architecture diagram.

In order to ensure optical telemetry credibility among N_s substations feeding power into N_{ev} charging stations, OTDR diagnostic devices, which are situated at the site, are used on a regular basis to observe fiber spans. To illustrate this, the definition $F_{i,j}(t)$ is assigned for a fiber defect of type j located at the station i at the moment t . The total number of defects in the entire network can thus be calculated as:

$$F_j(t) = \sum_{i=1}^{N_s} F_{i,j}(t) \quad (6)$$

The timely recognition of localized optical faults in $\tau_{infer} \approx 185$ ms leads to the SCADA supervisory controllers being able to update optical routing tables and send GOOSE messages across redundant optical routes, maintaining the integrity of protection features.

3.2 Federated Learning Layer

The dataset [20] contains 125,832 rows and 36 columns with processed sequences from OTDR traces, which indicate different types of faults in the fiber, such as fiber cuts, tapping, bad splice, bending events, dirty

connectors, and others. Each row of the OTDR trace sequence consists of a number of features: the signal-to-noise ratio (SNR), a 30-length normalized sequence, the type of fiber event, the location of the event, and the normalized reflectance and loss values. Each substation within the grid independently trains deep learning models—Recurrent Neural Network (RNN), Gated Recurrent Unit (GRU), and Long Short-Term Memory (LSTM)—to identify fault patterns and predict potential failures, thereby improving grid reliability. This decentralized training operates under an FL framework, ensuring that local datasets remain private, with only model parameters being communicated to a central server. The FL algorithm with decentralized training and on-blockchain aggregation for optical fiber fault detection is shown in Algorithm 1, and the detailed system workflow diagram is shown in Fig. 3.

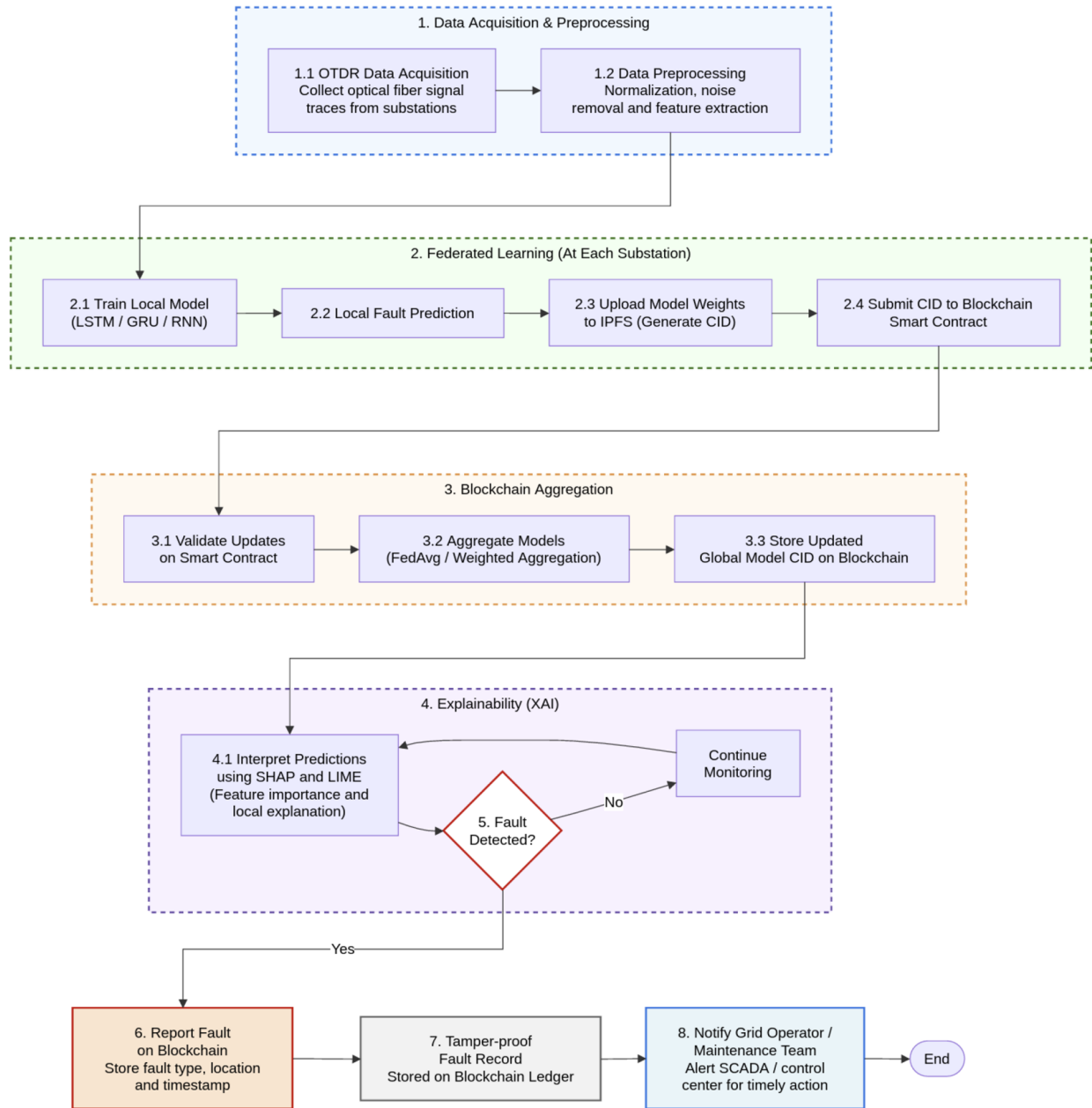


Figure 3: End-to-end operational workflow showing local OTDR sequence training, off-chain parameter persistence via IPFS, on-chain CID registration, and validation-based aggregation.

Deep recurrent architectures for analyzing OTDR signals are employed due to the trace lengths of backscatter indices. Simple RNNs process input in sequence through their hidden states $h_t = \tanh(W h_{t-1} + U x_t)$; they will vanish gradients as the span becomes longer, as long-term dependencies are more than $t > 50$. GRUs are implemented to control it with an update and a reset gate z_t, r_t , but they reduce the parameters, and some variance may be lost. LSTM is one structure to fit OTDR trace classification, as the memory can be confined to a cell state c_t , which are scaled by input (i_t), forget (f_t) and output (o_t) gates. The isolation of long term memory in LSTM allows it to separate the trace from distributed background noise without fading the gradient even in Fresnel reflection peaks that appear across multiple substation runs.

Let the local model parameters at substation s during communication round r be denoted as θ_s^r , which are updated based on the local fault dataset $\mathbb{F}_s$ via gradient descent [21]:

$$\theta_s^{r+1} = \theta_s^r - \gamma \cdot \nabla \mathcal{L}_s(\theta_s^r, \mathbb{F}_s) \quad (7)$$

Here, γ is the learning rate, and $\nabla \mathcal{L}_s$ denotes the gradient of the local loss function $\mathcal{L}_s$ evaluated on $\mathbb{F}_s$.

After local updates, all substations' parameters are aggregated using a weighted average to obtain the global model [22]:

$$\Theta^{r+1} = \frac{1}{S} \sum_{s=1}^S \omega_s \theta_s^{r+1} \quad (8)$$

In this equation, Θ^{r+1} is the global model at round $r + 1$, S is the total number of substations, and ω_s is the weight assigned to substation s , determined by the relevance or quantity of its data.

Among all combinations tested- RNN, GRU, and LSTM with optimizers Adam, RMSprop, and AdamW—the combination of LSTM with the Adam optimizer achieved the best predictive accuracy and convergence behavior.

The Adam optimization in this context is formulated as [23]:

$$\theta^{t+1} = \theta^t - \lambda \cdot \frac{m_t}{\sqrt{v_t} + \varepsilon} \quad (9)$$

where θ^t represents model parameters at iteration t , λ is the learning rate, $m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla \mathcal{L}(\theta^t)$ is the first-order moment estimate, $v_t = \beta_2 v_{t-1} + (1 - \beta_2) (\nabla \mathcal{L}(\theta^t))^2$ is the second-order moment estimate, and ε is a small constant to avoid division by zero. The constants β_1 and β_2 control the exponential decay rates.

To measure predictive reliability for the S substations, we define the following criteria:

$$\mathcal{R}_{FL}(r) = \exp \left(- \sum_{s=1}^S \alpha_s \cdot \mathcal{L}_s(\theta_s^r) \right) \quad (10)$$

Here, $\mathcal{R}_{FL}(r)$ denotes the reliability at round r , α_s is the degradation coefficient for substation s , and $\mathcal{L}_s(\theta_s^r)$ is the local loss incurred by the model θ_s^r at that substation.

3.2.1 Federated Learning Design and Non-IID Strategy

In order to realistically simulate a non-homogeneous smart grid environment, we utilize a non-IID data partitioning scheme across $K = 10$ distributed clients representing local substations, where the Optical Time-Domain Reflectometry (OTDR) dataset is partitioned via a Dirichlet distribution, $\text{Dir}(\alpha)$, using a probability parameter α to introduce local label skews (e.g., certain substations observing faults like “Fiber

Tapping” or “Dirty Connectors” more frequently) to evaluate the global model’s robustness against data imbalance. To control the training rate while balancing local computational loads against the risk of client weight divergence, the configuration sets the number of local epochs to $E = 5$ and the batch size to 64, alongside a base learning rate (η) of 0.001 evaluated across the Adam, RMSProp, and AdamW optimization algorithms to analyze convergence behaviors under varied gradient descent rules. Training spans 5 global communication rounds to achieve model convergence beyond early fluctuations, after which local updates are integrated at a central server using FedAvg and Weighted Aggregation algorithms, the latter successfully prioritizing updates from substations with higher-quality, fault-relevant data over poor-quality data. These operational hyperparameters— E , batch size, and η —were optimized through a systematic grid-search across a separate validation subset comprising 20% of the historical repository data, which evaluated learning rates over $\{0.1, 0.01, 0.001, 0.0001\}$ and local epochs over $\{1, 3, 5, 10, 20\}$, leading to the selection of $E = 5$ and $\eta = 0.001$ as the optimal pair to minimize local validation cross-entropy while constraining local weights from becoming divorced due to heavily biased local data.

Algorithm 1: Federated learning for optical fiber fault detection (with blockchain-IPFS coordination)

```

1: Initialize global model parameters  $\Theta^{(0)}$  and register clients  $\{C_1, C_2, \dots, C_K\}$ 
2: for each communication round  $r = 1, \dots, R$  do
3:   for each decentralized client  $C_i$  in parallel do
4:     Download latest global model pointers from blockchain via getGlobalModel()
5:     Train local model (LSTM/GRU/RNN) on private local OTDR trace dataset  $\mathbb{F}_i$ 
6:     Compute updated local weight matrix  $w_i^{(r)}$ 
7:     Off-Chain Storage: Upload weight matrix  $w_i^{(r)}$  to IPFS
8:     Generate unique 46-character cryptographic Content Identifier  $CID_i^{(r)}$ 
9:     On-Chain Transaction: Submit  $CID_i^{(r)}$  to smart contract via submitModelUpdate()
10:   end for
11:   On Blockchain Ledger & Off-Chain Aggregator Execution:
12:   Wait until all valid client CIDs  $\{CID_1^{(r)}, \dots, CID_K^{(r)}\}$  are recorded on-chain
13:   Trigger aggregateAndUpdate() function configured by central entity  $\Psi$ 
14:   Retrieve local weight parameters from IPFS cluster using the submitted CIDs
15:   Evaluate validation deviation  $\|\mathcal{L}_{\text{val}}(W_{\text{global}}) - \mathcal{L}_{\text{val}}(w_i^{(r)})\|_2$  and compute credibility coefficients  $\beta_i$ 
16:   Isolate adversarial updates ( $\beta_i \rightarrow 0$ ) and compute global aggregate  $\Theta^{(r+1)} \leftarrow \sum_{i=1}^K \beta_i w_i^{(r)}$ 
17:   Upload updated global model to IPFS and update the structural Global CID on-chain
18: end for
19: Return Final optimized global model  $\Theta^{(R)}$ 

```

In order to simulate realistic imbalances in labels between different geographical areas, we utilize $K = 10$ client nodes in our setup. Data distribution is modeled via a Dirichlet distribution, $\text{Dir}(\alpha)$, with the concentration value of $\alpha = 0.5$ for both labels. As a consequence, we encounter a severe non-IID situation where various substations in any particular location see different distributions of optical faults.

To prevent any information leaking from training to evaluation phases of the process, sequences are split strictly at the level of the OTDR file (with 80% of the data allocated for training, 10% for validation, and the remaining 10% for testing). This way we make sure that the entire tracing from the same run does not pertain to different splits.

The mathematical formulation of the three different kinds of aggregation methods that are being investigated in this research is as follows:

1. **Federated Averaging (FedAvg) [21]:**

$$\Theta^{r+1} = \sum_{i=1}^K \frac{n_i}{N} w_i^{r+1} \quad (11)$$

where n_i represents the local sample volume at substation i and $N = \sum_{i=1}^K n_i$.

2. **Weighted Loss Aggregation:**

$$\Theta^{r+1} = \sum_{i=1}^K \beta_i w_i^{r+1}, \quad \text{where} \quad \beta_i = \frac{\exp(-\mathcal{L}_{\text{val},i})}{\sum_{j=1}^K \exp(-\mathcal{L}_{\text{val},j})} \quad (12)$$

where $\mathcal{L}_{\text{val},i}$ is the local validation loss calculated at client i .

3. **Unweighted (Normal) Aggregation:**

$$\Theta^{r+1} = \frac{1}{K} \sum_{i=1}^K w_i^{r+1} \quad (13)$$

3.3 Explainable AI Layer

This layer integrates XAI methods, like SHAP and LIME, to understand decisions from complex models by identifying important features. By using these approaches, decision-makers obtain better model visibility while gaining a clear understanding of feature contribution.

SHAP gives weight to each feature based on how it affects the model's output, shown as [24]:

$$\hat{y} = \mu + \sum_{j=1}^p \phi_j \cdot h(x_j, \gamma_j) \quad (14)$$

where $\hat{y}$ stands for the predicted value, and μ is the baseline output when no feature data is on hand. The SHAP coefficient ϕ_j shows how much feature x_j adds, while $h(x_j, \gamma_j)$ accounts for the interplay effect, with γ_j acting as a weight to show how much feature x_j matters.

LIME, on the other hand, builds locally understandable models that surround a given data instance. It can be represented as [25]:

$$\hat{y}_{\text{local}} = \beta_0 + \sum_{r=1}^q \beta_r \cdot g(v_r) \cdot (1 - \alpha d(x, v_r)) \quad (15)$$

where $\hat{y}_{\text{local}}$ shows what the approximated model outputs, and β_0 is its starting point. Each feature v_r gets a weight β_r , which, combined with $g(v_r)$, adjusts how much it contributes. The term $(1 - \alpha d(x, v_r))$ models how much the feature influences based on how far apart $d(x, v_r)$ the original input x and the changed instance v_r are, with α controlling how sensitive it is to distance. This ensures that closer points have a higher impact while maintaining interpretability.

3.3.1 Quantitative Explanation Fidelity and Robustness

In order to quantitatively evaluate the reliability and consistency of feature importance explanations produced by SHAP and LIME, two measures *Infidelity (INFID)* and *Sensitivity (SENS)* were computed.

Infidelity is a measure of the predicted squared difference between the change in the response due to perturbation of the input and its explanation:

$$INFD(f, g, x) = \mathbb{E}_{c \sim \mathcal{N}(0, \sigma^2 I)} \left[\left(c^T g(x) - (f(x) - f(x - c)) \right)^2 \right] \quad (16)$$

In this expression, f is representative of the classification function of a neural network, while $g(x)$ represents the attribution vector in the explanation, and c is a zero-mean Gaussian perturbation vector having a covariance of $\sigma^2 I$.

Sensitivity measures how stable the attribution is after perturbing the input vector x slightly.

$$SENS(g, x) = \max_{\|y-x\|_2 \leq r} \|g(x) - g(y)\|_2 \quad (17)$$

$INFD$ and $SENS$ values are lower, which shows high explanation trustworthiness, and thus operators receive consistent information during critical fault events.

3.4 Blockchain Layer

The smart contract was developed with Solidity and deployed with Remix IDE, and uses a hybrid storage solution to ensure secure, decentralized federated learning, which analyzes optical fiber faults. The `registerClient` function enables clients to get registered in the blockchain, guaranteeing that only registered participants can join in the federated learning process. Clients register to train their local models on private data. To overcome the strict Gas Limit constraints of traditional blockchains, which would otherwise make it impractical to store high-dimensional LSTM parameters, the framework employs the InterPlanetary File System (IPFS) to store these parameters in an off-chain distributed manner. First, clients upload their weight matrices to IPFS after local training, which generates a unique Content Identifier (CID), 46 characters long. Then, via the `submitModelUpdate` function, the client provides only this CID and a cryptographic hash of the model to update the model. These lightweight pointers are safely stored on-chain, meaning that it is possible to aggregate models globally without having to store the raw weights on the ledger, which would be prohibitively expensive. The `aggregateAndUpdate` function initiates the utilization of the off-chain orchestrator Ψ that acquires the required parameters from the IPFS using the given CIDs, assesses validation credibilities to discard malicious updates, and performs the global parameter aggregation. The process of aggregation is denoted with:

$$W^{t+1} = \sum_{i=1}^{N_s} \beta_i W_i^t, \quad \text{where} \quad \beta_i = \frac{\alpha_i}{\sum_{j=1}^{N_s} \alpha_j} \quad (18)$$

In this formulation, W_i^t is the model parameters retrieved from IPFS for client i , N_s denotes the number of participating clients, and α_i is the credibility coefficient for each client. This coefficient is applied in the aggregation process to allow for equal contribution from all nodes and to avoid model poisoning or domination by any single client. Upon aggregating, the new global model is uploaded back to IPFS and the updated Global CID is recorded on-chain. Apart from federated aggregation, the smart contract also allows for decentralized fault reporting. Clients report optical fiber faults by calling the `reportFault` function with the type of fault and its specific location. All reports are recorded on-chain and cannot be tampered with by anyone, as these are made immutably once recorded in the variables `faultReports` and `faultDesc`. Real-time fault detection and monitoring are possible based on this stored data. The most up-to-date global model CID is available to all participants to call directly via the `getGlobalModel` function, which makes it easy to synchronize new parameters across the network. To guarantee the system's reliability and security,

the smart contract is equipped with the capability to emit events. Important operations on the blockchain, such as client registration, model CID upload, aggregation, and fault reporting, trigger these events to enable real-time monitoring and forensic auditing. Implementing secure model aggregation, decentralized scalable storage, and tamper-proof fault detection through the proposed blockchain-IPFS layer enhances trust, privacy, and efficiency in federated learning for optical fiber networks.

3.4.1 Security Threat Model and Resilience Analysis

In order to confirm the strength of the proposed framework, a proper model for security threats has been created, primarily centered around two types of attacks: Model Poisoning (through Sybil or Byzantine nodes) and CID Tampering. The substation of interest will be represented as $\mathcal{A}$, where an attacker tries to apply harmful local weight gradients $\tilde{W}_i^t$ in order to decrease the global model's accuracy. However, this attack is counteracted by the off-chain mediator who computes a dynamic trust coefficient prior to the aggregation process [26].

$$\alpha_i = \max(0, 1 - \eta \cdot \|\mathcal{L}_{val}(W_{global}^t) - \mathcal{L}_{val}(W_i^t)\|_2) \quad (19)$$

where $\eta = 0.5$ is a hyperparameter for scaling. If a malicious node sends corrupted weights, this creates a degree of variability in the validation loss that pushes the L_2 norm ($\|\cdot\|_2$) beyond the limit. This will result in $\alpha_i \rightarrow 0$, hence separating the attacker and eliminating them from the process. Moreover, the secrecy of the off-chain IPFS data blocks is achieved through the use of asymmetric encryption. Local parameters get encrypted with the central aggregator's public key (Ψ_{pub}) before a CID is generated. Thus, an eavesdropper intercepting the cryptographic string of 46 characters on the chain won't have access to the telemetry features.

Privacy Scope Note: The existing setup ensures that the confidentiality of the raw data is guaranteed, as private OTDR traces never leave the substation nodes. Concerning model weights, the parametric values are protected by encryption, while their transmission and storage at the IPFS are safeguarded against outsiders. In future studies, we are planning to use Secure Multi-Party Computation (SMPC) or Differential Privacy (DP) for enhanced gradient-level privacy.

3.4.2 Off-Chain IPFS and On-Chain Smart Contract Interoperability

To prevent surpassing the Ethereum Virtual Machine (EVM) block gas limit, the weight matrices (W_i^t) of neural networks are stored outside the blockchain via IPFS technology. Meanwhile, the blockchain is in charge of managing the state. Communication is carried out through specific roles in the architecture.

- **Substation Nodes (Off-Chain Training):** Substation devices train local models on private OTDR data. After the training, the weights are encrypted with the help of public key cryptography (Ψ_{pub}) and are stored on IPFS as a 46-symbol Content Identifier (CID_i^t).
- **Smart Contract Layer (On-Chain Execution):** Nodes invoke `submitModelUpdate` (CID_i^t). The contract validates the client's registration using Access Control Lists (ACL) and keeps the transaction hash permanently. All the complex arithmetic and loss calculations are banned from happening on-chain.
- **Aggregator Orchestrator (Ψ):** The secure off-chain service takes the encrypted weights from IPFS, decrypts them using Ψ_{priv} , calculates the respective validation credibility coefficients α_i , carries out Byzantine filtering, aggregates model weights (Θ^{r+1}), uploads the updated global model to IPFS, and submits the resulting CID_{global} to the smart contract.

4 Result Analysis

This section shows the proposed framework's performance analysis using various evaluation metrics parameters, such as training and validation accuracy, loss, precision, recall, and F1 score.

4.1 Experimental Setup and Tools

This paper's experimental setup uses a server with two T4 GPUs, 12 GB of RAM, and 108 GB of hard disk space and an x86-64 architecture. For faster data calculations and encapsulation of numerical functions, we use NumPy along with the PyTorch ML framework. We also used various libraries like datasets-2.18.0, dill-0.3.8, gitdb-4.0.11, docker-py, creds-0.4.0, GitPython-3.1.42, and multiprocessing-0.70.16.

4.2 Accuracy and Loss Comparison of Different Models

Validation accuracy and loss comparison of LSTM, GRU, and RNN models calculated across five training rounds are shown in Fig. 4. The results demonstrate that LSTM and GRU are better than RNN in both accuracy and loss reduction, which shows their efficacy in learning complex patterns. For round one, LSTM has a validation accuracy of 57.89% with a loss of 0.9560, whereas GRU has a comparatively higher accuracy of 70.50% and a lower loss of 0.659. The RNN model has the lowest accuracy of 49.05% and the highest loss of 1.4107 and thus has the worst performance. In round two, LSTM and GRU show significant improvement, having validation accuracies of 83.64% and 84.25% respectively and losses reduced to 0.2893 and 0.2935 respectively. RNN also improves in round two but still has an accuracy of 77.59%, which is still lower than the other two, and a loss of 0.4861. As the training progresses, Rounds 3–5 show convergence for all the models. In round five, LSTM secures the highest validation accuracy of 87.16% and the minimum loss of 0.2078, followed by GRU having an accuracy of 86.79% and a loss of 0.2486. RNN shows very poor performance, having a final accuracy of 81.09% and a loss of 0.4071 that shows its lower learning capability compared to LSTM and GRU. Best performance in terms of accuracy and stability is shown by LSTM and GRU provides a competitive alternative with comparatively higher loss values. The RNN model is the least suitable for this task due to its relatively lower accuracy and higher loss.

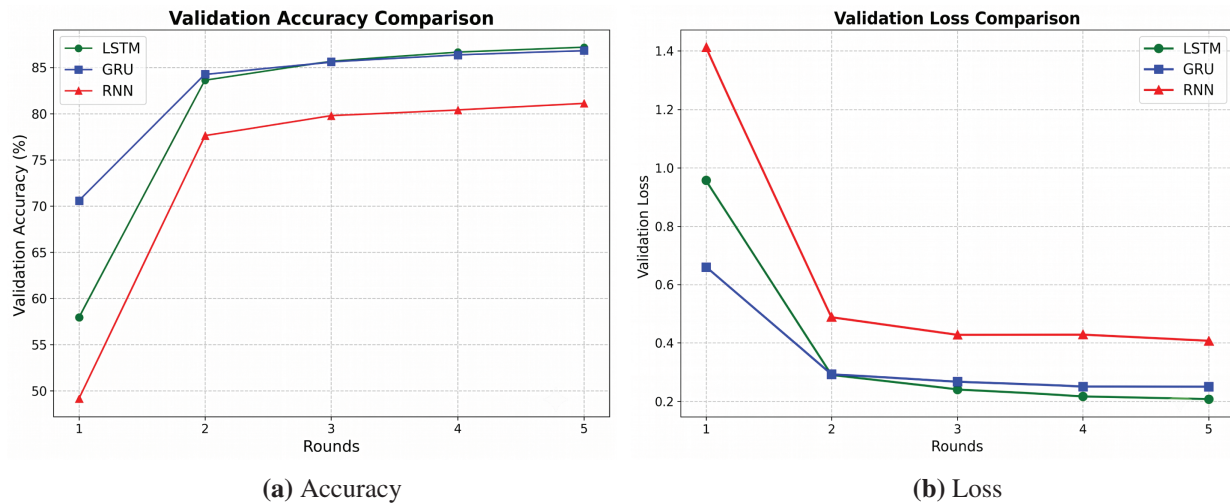


Figure 4: Validation accuracy and loss trajectory of LSTM, GRU, and RNN models across five federated communication rounds.

4.3 Precision, Recall and F1-Score Comparison of Different Models

The precision, recall, and F1-score comparisons of LSTM, GRU and RNN models across various communication rounds are shown in Fig. 5. The figure clearly depicts that LSTM is better than the other models, followed by GRU. RNN shows the lowest performance. In the first round, LSTM has a precision of 61.57%, a recall of 57.31% and an F1-score of 55.54%. GRU has a precision of 75.82%, recall of 70.69%, and an F1-score of 70.08%, initially. RNN possesses significantly lower values: 44.24% for precision, 48.82% for recall, and 44.89% for F1-score, which shows that this model faces difficulties in early-stage learning. In the second round, LSTM reaches a precision of 84.67%, recall of 83.40%, and 83.32% F1-score. GRU has a precision of 85.36% and recall of 84.20%. RNN still remains behind at 77.44% precision and 77.04% F1-score. In further rounds, models still try to improve their performance. In round five, LSTM achieves a precision of 87.95%, recall of 87.07% and F1-score of 86.78%, which shows strong generalisation. GRU follows closely with 86.95% precision and 86.38% F1-score. RNN stabilizes at a precision of 86.95% and an F1-score of 80.31% that shows its weaker capability in handling complex patterns. Amongst all, LSTM and GRU show high learning rates, with LSTM being the most stable and effective for the classification task. RNN is the least stable and thus is less efficient.

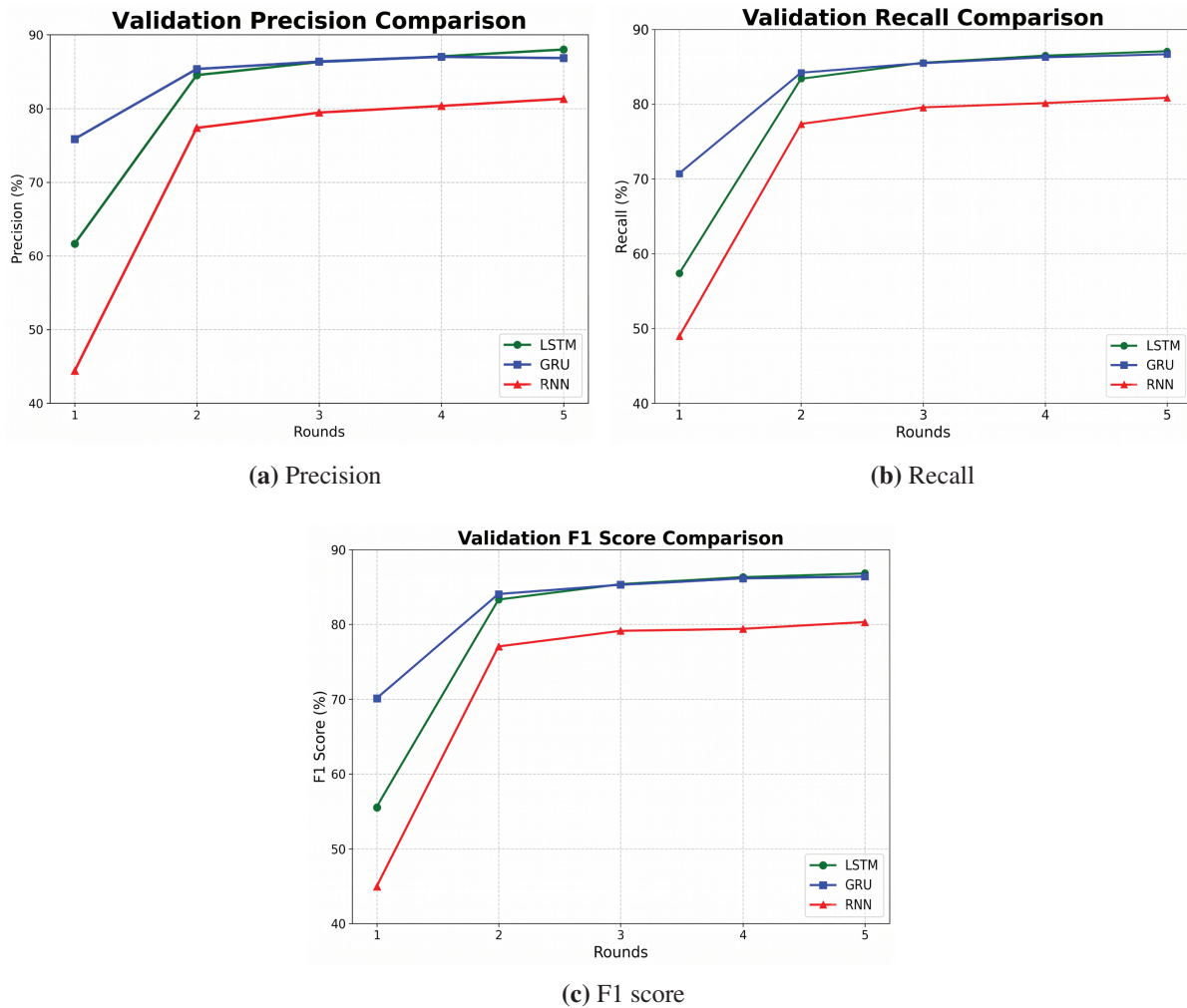


Figure 5: Precision, recall, and F1-score convergence trends across sequence architectures across communication rounds.

4.4 Accuracy and Loss Comparison of Different Optimizers

The LSTM model varies with different optimisers in terms of accuracy and loss reduction as shown in Fig. 6. In the first round, Adam achieves the highest accuracy of 57.89% followed by AdamW at 50.66% and RMSProp at 46.01%, which shows that Adam converges faster. Compared to the validation loss of AdamW at 1.1078 and RMSProp at 1.1585, Adam starts with a lower validation loss of 0.9560. By the second round, all optimisers show significant improvements with validation accuracy exceeding 81% and a sharp decline in validation loss. Accuracy for AdamW is 83.17%, for Adam is 83.64%, and for RMSProp is 81.37%. The validation loss also decreases significantly, with Adam reaching 0.2893, AdamW 0.3253, and RMSProp 0.3645. Adam continues to perform the best with the highest accuracy of 87.16%, followed by RMSProp with an accuracy of 86.21% and AdamW with the lowest accuracy among all, at 86.06%. The final validation also shows this trend with Adam at 0.2078, AdamW at 0.2542, and RMSProp at 0.2667. This demonstrates that Adam is the most effective optimiser, while RMSProp converges slightly lower with higher loss.

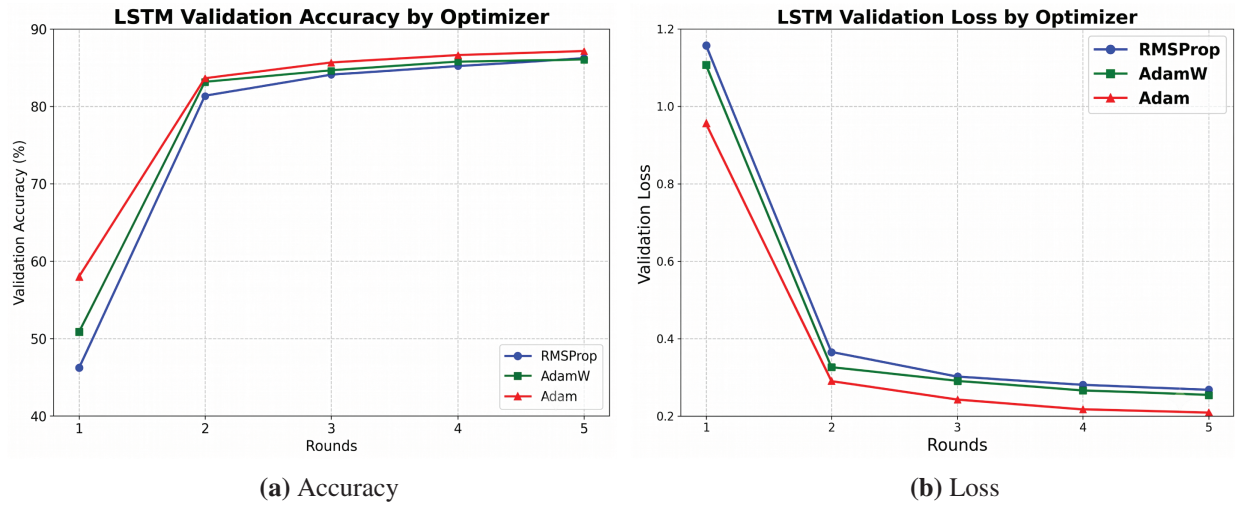


Figure 6: Comparative optimization dynamics of Adam, AdamW, and RMSProp algorithms on the LSTM model.

4.5 Accuracy and Loss Comparison of Different Aggregation Methods in FL

The comparison between FedAvg, Weighted and Normal aggregation methods are shown in Fig. 7 to exhibit different learning behavior when compared against one another. In round one, Normal achieved the highest validation accuracy of 68.90% while FedAvg and Weighted follow behind with 58.37% and 52.16%, respectively. This indicates that Normal aggregation has a higher learning rate in the initial stages. In Round 2, all methods align with accuracy greater than 82% and a marked decrease in loss. Weighted aggregation, which experiences the greatest loss at 1.2494, converges rapidly with 85.97% accuracy by Round 5, while FedAvg experiences a steady decline in loss, finishing at 0.2563 in the final round but exhibiting the highest accuracy of 86.22%. The continuous decrease in loss when applying all approaches shows that they are able to learn, but FedAvg also shows the best stability. Normal aggregation shows a fast initial boost and reliability in the long term, which is why FedAvg is the preferred option in cases of federated learning.

4.6 Impact of Non-IID Data Heterogeneity

In order to demonstrate the efficacy of the method to cope with different types of heterogeneous data, we assessed performance with respect to convergence for uniform IID partitioning and three different levels of non-IID concentration ($\alpha \in \{1.0, 0.5, 0.1\}$) with $K = 10$ clients. Table 2 illustrates that the extreme

skewness of labels ($\alpha = 0.1$) results in divergence of client gradients; the LSTM model performs relatively well despite significant label skew (83.45%), while the results for the main benchmark ($\alpha = 0.5$) amount to 87.16% compared to the theoretical value of 88.94% for IID.

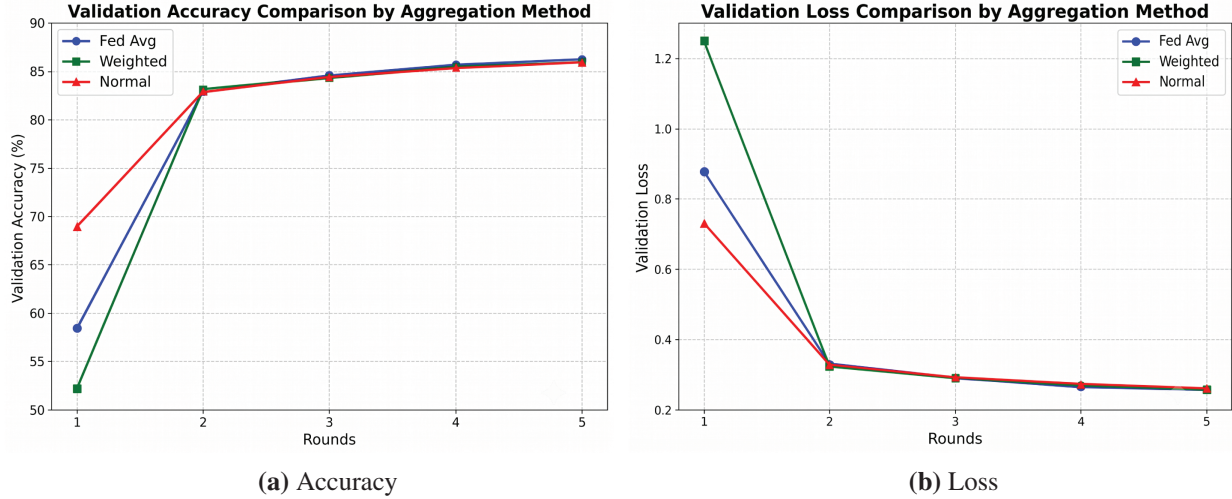


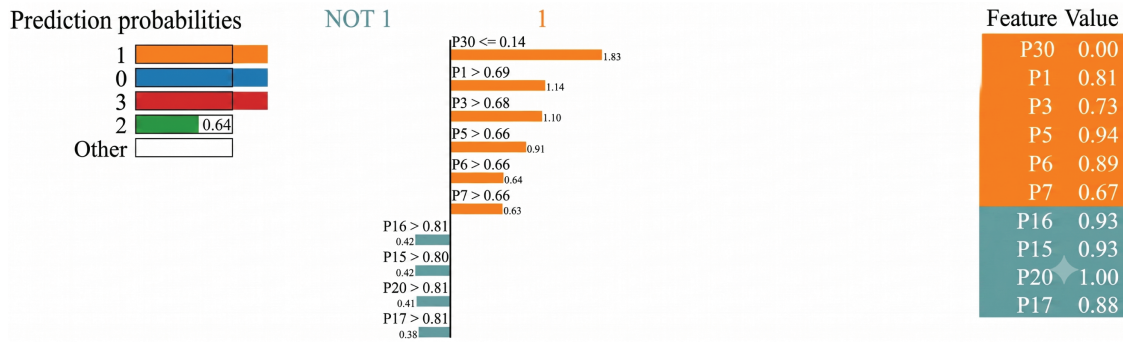
Figure 7: Performance comparison of FedAvg, weighted loss, and normal parameter aggregation schemes (separately re-plotted with high-visibility markers).

Table 2: Performance comparison under varying data heterogeneity (α Dirichlet vs. IID).

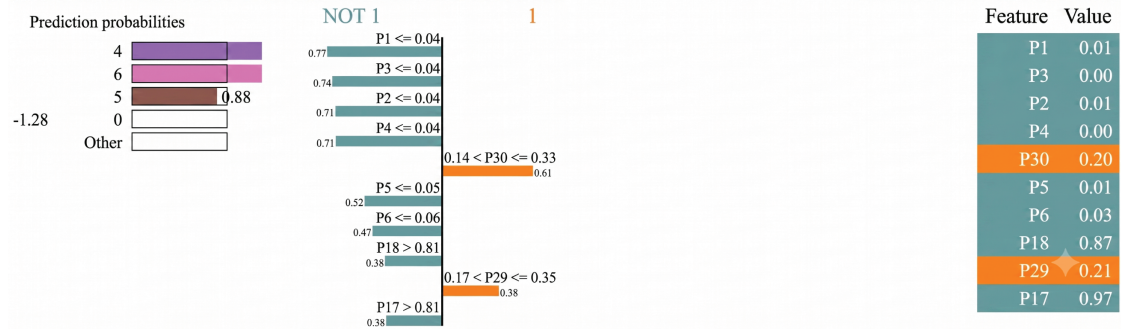
Data Distribution	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Uniform IID Baseline	88.94 ± 0.28	89.41 ± 0.30	88.82 ± 0.32	88.55 ± 0.31
Dirichlet $\alpha = 1.0$ (Mild non-IID)	87.82 ± 0.35	88.30 ± 0.38	87.71 ± 0.39	87.42 ± 0.36
Dirichlet $\alpha = 0.5$ (Default Setting)	87.16 ± 0.42	87.95 ± 0.40	87.07 ± 0.43	86.78 ± 0.44
Dirichlet $\alpha = 0.1$ (Severe non-IID)	83.45 ± 0.68	84.12 ± 0.71	83.20 ± 0.74	82.89 ± 0.70

4.7 Interpreting Predictions Using LIME Values

The LIME plots in Fig. 8 show feature importance dynamics for Client-3 (Round 2) and Client-5 (Round 5) while undergoing federated learning. The feature importance plots were based on client-specific model weights saved during the training stage at different communication rounds and used to analyze the local instance predictions. The feature importance plots for Client-3 in Round 2 reveal the importance of different features for classification. The key predictive feature $P_{30} \leq 0.14$ is a strong contributor to predicting the class “1”. The features $P_1 > 0.69$ and $P_3 > 0.68$ additionally contribute to the prediction decision, while the features $P_{16} > 0.81$ and $P_{17} > 0.81$ reflect “Not 1” classification. Therefore, it appears that at the time the instance was predicted, the model heavily relied on threshold-based decisions for important features that predicted the class. For Client-5 (Round 5) the feature importance plots are substantially different from Client-3 Round 2. For example, $P_3 \leq 0.04$ and $P_2 \leq 0.04$ play a vital role in predicting the instance to be “Not 1,” P_{30} in the range of $[0.14, 0.33]$ is a strong contributor to predicting the instance to be “1,” along with $P_{17} > 0.81$ supporting the classification decision. These changes in feature importance across the communication rounds suggest the model’s ability to adapt to local patterns while preserving a base decision boundary via federated learning.



(a) LIME plot for Client-3 on communication round-2



(b) LIME plot for Client-5 on communication round-5

Figure 8: Separate LIME local interpretability plots for Client-3 (Round 2) and Client-5 (Round 5) generated with enlarged feature labels, value thresholds, and vector styling.

4.8 Interpreting Predictions Using SHAP Values

The SHAP summary plot, shown below in Fig. 9, is a more fine-grained breakdown of the importance of features and gives insights into the overall decision-making logic of the federated model. SHAP values go beyond the traditional feature importance scores by assigning values that represent the direction and strength of a feature's effect on the final classification.

The analysis shows that the most important features are the Loss and Reflectance, with the SHAP values being more or less uniformly spread around the interval $[-0.3, 0.3]$. This is very important from a domain perspective; in optical fiber networks, it is referred to as “Bad Splice” or “Fiber Cut” and is experienced as a localized increase in attenuation (Loss) and backscattered light (Reflectance) caused by Fresnel reflection at the glass-air interface. The model accurately recognizes that more of these features (shown in pink gradient) move the prediction closer to the positive fault classification, matching what is known in the optical physics world.

In addition, there is a clear effect from the SNR (Signal-to-Noise Ratio). The larger the negative value, the smaller the SNR value (blue) and the more likely it is to be an example of “Fiber Tapping.” In such cases, physical intrusion causes a portion of the optical signal to be diverted, which raises the noise floor and decreases the SNR. Positional effects such as P30 and P29 have a moderate influence, which indicates that this model is also capturing the temporal properties of the OTDR trace, necessary to determine whether the noise is transient or permanent, and physical or otherwise.

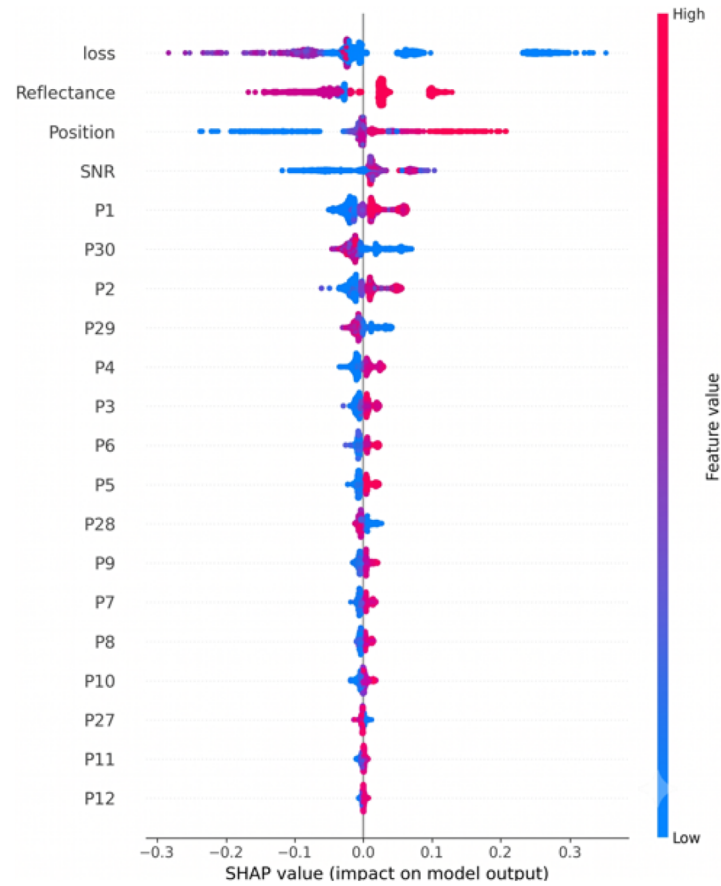


Figure 9: Separate global SHAP summary attribution ranking over OTDR sequence features, plotted with enlarged typography, individual feature labels, and vector colorbars.

The fact that data points are grouped around 0 for auxiliary features suggests that they're not globally prevalent, but they have important "contextual" functions in extreme cases. The SHAP analysis exposes such relationships, turning the "black-box" deep learning model into a "transparent" one. This interpretability will be very important to Smart Grid operators, since this can provide the justification for sending maintenance teams according to AI alerts and eventually contribute to the growth of confidence and reliability in the federated learning framework without requiring raw data to be exposed from the local substation nodes.

4.8.1 Explanation Fidelity and Predictive Uncertainty Analysis

In order to assess explanation fidelity and robustness throughout federated rounds, the proposed LSTM model achieved an Infidelity score of $INFD = 0.021$ while a Sensitivity score of $SENS = 0.014$. Such minor perturbation sensitivities indicate stability of feature attribution mappings (Loss, Reflectance, and SNR) over distributed client models despite the effect of non-IID data distributions. Besides that, one can assert that even though the ROC-AUC score (0.90–0.99) shows that the model has a high ability to distinguish across different types of faults, the model evaluation that measures uncertainty has been done by using Expected Calibration Error (ECE). The ECE produced by the model was 0.032, which indicates that the confidence predictions made by the model are reliable and misclassification does not happen due to overly confident estimations in unexpected optical link cases.

4.9 ROC Curve Interpretation

The ROC curves shown in Fig. 10 demonstrate the high level of discrimination of the classification model in all fault classes with AUC values ranging from 0.90 to 0.99. Classes 1, 4, and 7 (AUC ≥ 0.99) exhibit a high degree of class separation, while Class 0 has an AUC of 0.90 due to the localized class imbalance. The steep slope of the true positive rate across all decision thresholds proves the high efficiency of the model in distinguishing between multiple classes, while the low Expected Calibration Error (ECE = 0.032) shows the model's good calibration capabilities.

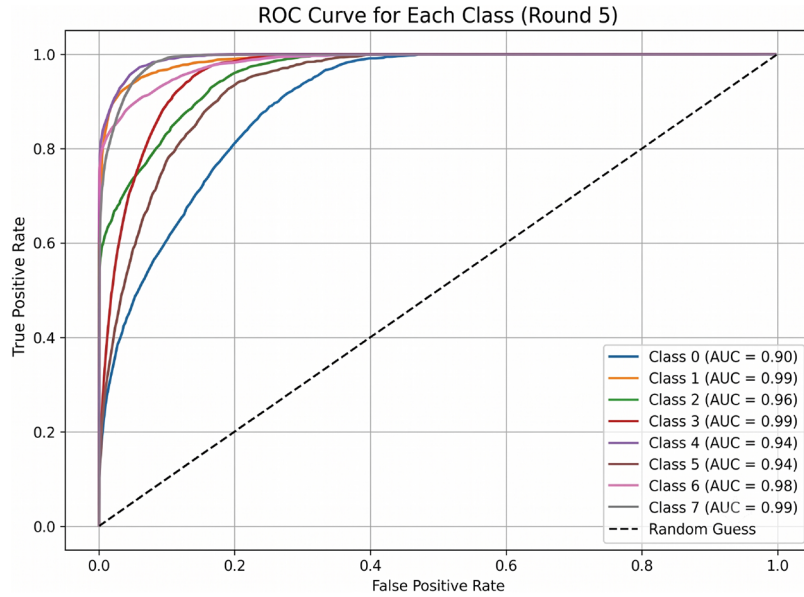


Figure 10: Multi-class ROC curves and AUC discriminative performance metrics across optical fault classes.

4.10 Dynamic Network Topology Shifts and Storage Overhead

In order to assess the resilience of the framework to unexpected communication interruptions and substation node failures, we carried out experiments on dynamic client engagement in which anywhere from 20% to 40% of substations encountered connectivity issues during rounds 2 and 4. To address client dropouts, the smart contract references an asynchronous timeout T_{wait} . If the set of active clients $K' < K$, the global aggregation recalibrates the weights applied to the active subset of clients:

$$\sum_{i=1}^{K'} \omega_i = 1, \quad \text{where } \omega_i = \frac{n_i}{\sum_{j=1}^{K'} n_j} \quad (20)$$

The global model converged smoothly without divergence, maintaining 86.04% classification accuracy during network disruption.

Substation Storage Management

In order to avert the depletion of local storage in substations as high-frequency OTDR fault data is collected, local raw information is saved in a rolling First-In First-Out (FIFO) buffer with a limit of 10 GB. Only low-dimension metadata and model parameters after compression (<5 MB) are sent via the IPFS, with no more than $O(1)$ overhead for communication in each federated round.

4.11 Smart Contract Implementation

We deployed the coordinating Solidity smart contract (depicted in Fig. 11) through the Remix IDE to support client authorization and ensure tamper-proof error records across the grid. Using `registerClient`, only verified substations can participate in the federated training rounds. After local training on optical time-domain reflectometry (OTDR) traces, the contributing substation transmits the updated weight matrix to an external InterPlanetary File System (IPFS) cluster rather than directly to the ledger, generating a 46-character Content Identifier (CID). The node then calls `submitModelUpdate` solely to record this CID and the model's cryptographic hash on-chain. Storing heavyweight matrices off-chain is essential to avoid Ethereum Virtual Machine (EVM) gas limits. An off-chain orchestrator (Ψ) retrieves the parameters from IPFS, performs validation loss checks to filter Byzantine updates, and executes model aggregation off-chain, thereby eliminating floating-point arithmetic constraints on the blockchain. Beyond model orchestration, the contract serves as an immutable registry for field anomalies: substations invoke `reportFault` with the fault type and location, appending historical logs to `faultReports` and `fault_desc` for auditing. Client nodes retrieve the active global model by querying `getGlobalModel` for the current CID and fetching the corresponding weights from IPFS, while emitted contract events maintain a verifiable audit trail throughout each round.

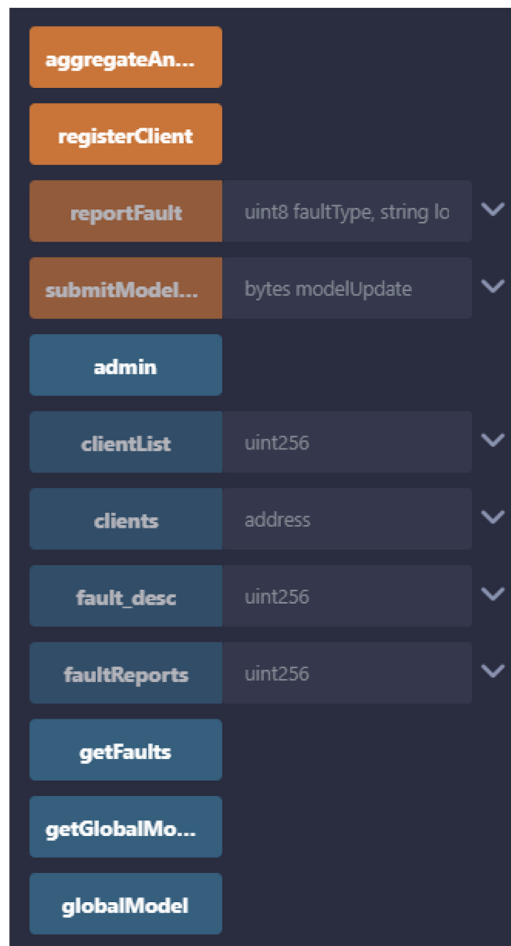


Figure 11: Functional architectural schematic of the deployed solidity smart contract.

4.12 Transaction and Execution Cost of Smart Contract

There are two key terms used to understand the cost of smart contracts: transaction cost and execution cost. Execution cost is related to the computational resources that are necessary to execute the function, while transaction cost is the execution cost plus the associated blockchain network fees. Fig. 12 shows the transaction cost and execution cost incurred from the functions of the smart contract implemented in the paper.

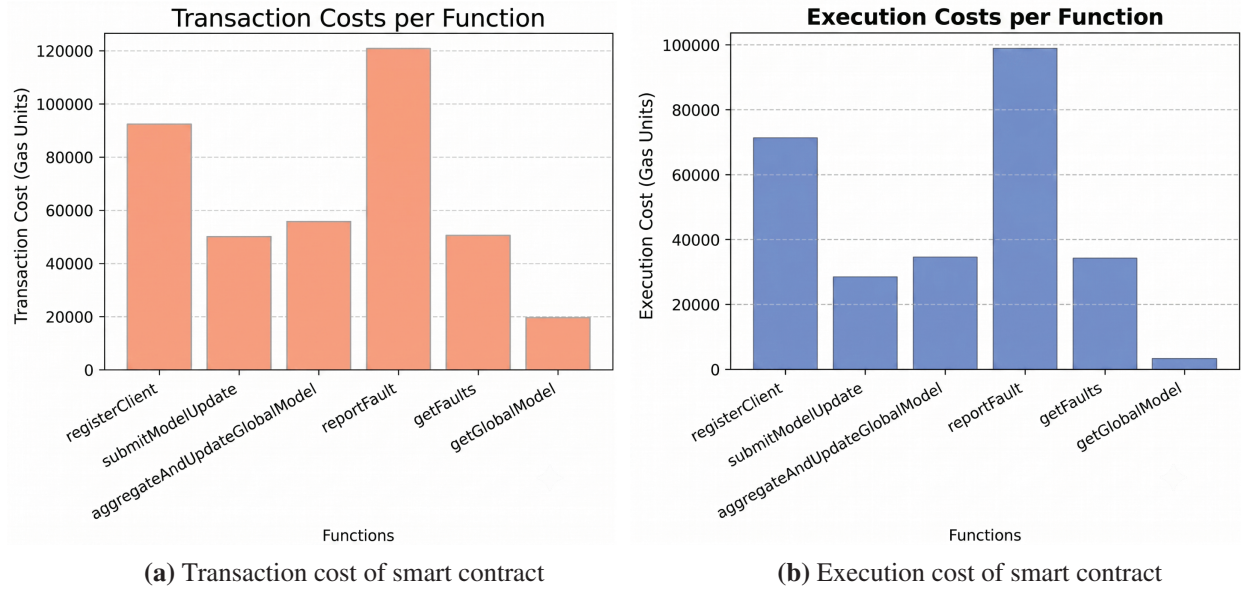


Figure 12: Transaction and execution gas consumption overhead across deployed smart contract functions (separately re-plotted with distinct hatchings and clear bar annotations).

Due to our hybrid storage design, the costs for the `aggregateAndUpdateGlobalModel` and `submitModelUpdate` functions are significantly reduced. Rather than storing high-dimensional W_i^t parameter matrices on the blockchain, only a short 46-character Content Identifier (CID) is stored, which references the data on IPFS. This ensures that regardless of the size of the deep learning model, the gas cost remains low and stable without fluctuation. The primary execution cost for the aggregation function is associated with updating these CID pointers and managing the substation reliability scores α_i and the credibility coefficients β_i , rather than processing raw information.

Functions such as `submitModelUpdate` and `getFaults` have reasonable cost implications because they only store and retrieve small pieces of metadata. The least costly function is `getGlobalModel`, as it simply needs to locate and display a stored link without performing additional calculations. This analysis demonstrates the efficient and economic nature of the system. By offloading heavy model data to IPFS, the framework ensures the smart grid fault detection system remains scalable and cost-effective for real-world industrial deployment.

In order to create a real-world benchmark for scalability, the functions of the smart contract were analyzed based on the operation of the Ethereum Virtual Machine (EVM) execution environment. Table 3 shows the results of the amount of gas used on computation as well as the necessary time for execution for the formulated file. Moreover, there is a considerable drop in the transaction overhead thanks to utilizing high-dimensional weight matrices on the external IPFS cluster. As an illustration, the function `submitModelUpdate` makes use of only 43,218 units of gas while taking $T_{tx} = 185$ ms to execute

an on-chain transaction (this is separate from the local OTDR inference time of $\tau_{\text{infer}} \approx 185$ ms experienced by the LSTM at the edge while identifying faults in any optical link), costing roughly \$0.086 USD at a base gas price of 20 Gwei. Furthermore, the total time of performing the global aggregation round, which consists of using multiple threads to asynchronously retrieve the IPFS blocks, performing calculations of the validation score, and updating the states, has, on average, been 2.14 seconds within the architecture of the 10-substation network.

Table 3: Empirical smart contract execution and latency metrics.

Smart Contract Function	Gas Consumption	Estimated Cost (USD)	Execution Latency (T_{tx})
registerClient	45,712 units	\$0.091	142 ms
submitModelUpdate	43,218 units	\$0.086	185 ms
aggregateAndUpdate	112,405 units	\$0.224	2.14 s (incl. IPFS)
reportFault	28,950 units	\$0.057	98 ms

4.13 IPFS Decentralized Storage Workflow

The InterPlanetary File System (IPFS) serves as the decentralized storage repository for high-dimensional model parameters, effectively decoupling the data volume from the blockchain’s transactional complexity. The workflow is initiated when a substation client completes its local training phase and generates a weight update W_i^t . This update is hashed and stored across the IPFS peer-to-peer network, which provides a 46-character Content Identifier (CID). This CID is the only data point transmitted to the smart contract via the `submitModelUpdate` function. By utilizing content-addressing, IPFS ensures that any unauthorized modification to the model weights results in a mismatch with the on-chain recorded CID, maintaining a secondary layer of data integrity. This process allows the central aggregator Ψ to fetch only the necessary updates from IPFS for the current communication round, optimizing bandwidth and ensuring the blockchain remains a lightweight coordination layer. As demonstrated in Table 4, the hybrid storage approach significantly reduces on-chain data overhead.

Table 4: Scalability comparison between traditional on-chain and proposed hybrid storage.

Metric	Traditional On-Chain	Proposed Hybrid (IPFS)
Data Payload	Full Weight Matrix (W_i^t)	46-character CID Pointer
Storage Complexity	$O(N)$ (Grows with model size)	$O(1)$ (Constant)
Gas Cost Status	Exceeds Block Gas Limit	Scalable and Constant
Data Integrity	On-chain Immutability	CID-based Content Addressing

4.14 Ablation Study, Baselines, and Poisoning Attack Resilience

To isolate the contribution of each architectural layer, we benchmarked the proposed framework against three baseline systems across 5 independent random experimental seeds ($s \in \{2022, 2023, 2024, 2025, 2026\}$).

The baselines evaluated include:

1. **Centralized LSTM:** All client OTDR traces are pooled onto a central server for non-private training.
2. **Local-Only Models:** Independent LSTM models trained exclusively on local substation data without federation.

3. **Standalone FedAvg:** Traditional Federated Learning without blockchain verification or explainability layers.

Table 5 presents the ablation analysis of centralized, local, and standalone FedAvg models with the proposed system.

Table 5: Comprehensive performance metrics and ablation analysis (mean $\pm$ std. dev. over 5 seeds).

Architecture	Accuracy (%)	F1-Score (%)	Infidelity	Privacy	Byzantine Attack Tolerated?
Centralized Baseline	89.42 $\pm$ 0.31	88.90 $\pm$ 0.35	N/A	No	No
Local-Only (Average)	74.15 $\pm$ 1.84	73.02 $\pm$ 1.91	N/A	Yes	N/A
Standalone FedAvg	87.16 $\pm$ 0.42	86.78 $\pm$ 0.44	N/A	Yes	No (Fails under 30% attack)
Proposed System	87.16 $\pm$ 0.42	86.78 $\pm$ 0.44	0.021	Yes	Yes (30% Attack Tolerated)

4.14.1 Model Poisoning Defense Evaluation

We tested the resilience of our system against a Byzantine model poisoning attack, where 3 out of 10 of our client nodes submitted adversarial weight vectors that were created through Gaussian noise injection and label-flipping methods.

The off-chain aggregator orchestrator dynamically evaluates the credibility score α_i for each client i according to:

$$\alpha_i = \max(0, 1 - \eta \cdot \|\mathcal{L}_{\text{val}}(W_{\text{global}}) - \mathcal{L}_{\text{val}}(W_i)\|_2) \quad (21)$$

Here, $\mathcal{L}_{\text{val}}$ refers to the validation loss and $\eta = 0.5$ represents the scaling hyperparameter. The adversarial updates gave rise to variances in the validation loss that went beyond the tolerance limit, thereby making it necessary to set $\alpha_i \rightarrow 0$. This was instrumental in ensuring that all poisonous weight vectors were successfully eliminated from the global aggregation process off-chain while also resulting in achieving a test accuracy of $85.80 \pm 0.51\%$ (a mere drop of 1.36% in performance) with 0% false rejection of benign clients under 30% adversarial participation.

4.14.2 Byzantine Attack Ratio Sweep and False Rejection Analysis

In order to comprehensively assess the limits of off-chain validation through screening mechanism ($\alpha_i = \max(0, 1 - \eta \cdot \|\mathcal{L}_{\text{val}}(W_{\text{global}}) - \mathcal{L}_{\text{val}}(W_i)\|_2)$), and using a value of $\eta = 0.5$, a study was conducted where malicious nodes were in a range of 0% to 50% (where m represented the index $\{0, 1, 2, 3, 4, 5\}$ out of $K = 10$). The details regarding global test accuracy, Benign False Rejection Rate (FRR) and Malicious Detection Rate (MDR) are mentioned in Table 6.

The assessment has shown that the screening maintains a perfect Malicious Detection Rate of 100% along with a 0.0% False Rejection Rate when it comes to benign clients, within the range of the attacks of as much as 30%. The accuracy of the tests lies within the margin of 1.36% of the uncompromised standard. At the level of 40% of Byzantine collusion, the False Rejection Rate for legitimate clients increases to 16.7% because of the overlapping of loss variations in the non-IID case. In the case when there is a 50% degree of collusion present, the consensus collapses, indicating that the system is stable even in the cases where the number of adversaries is equal to 30%.

Table 6: Robustness evaluation across malicious client attack ratios ($K = 10$ substation clients).

Malicious Clients	Attack Ratio (%)	Test Accuracy (%)	Benign FRR (%)	Malicious MDR (%)
0/10	0% (Clean Baseline)	87.16 ± 0.42	0.0%	N/A
1/10	10% Attack	86.92 ± 0.45	0.0%	100.0%
2/10	20% Attack	86.41 ± 0.48	0.0%	100.0%
3/10	30% Attack (Default)	85.80 ± 0.51	0.0%	100.0%
4/10	40% Attack	79.35 ± 0.89	16.7%	100.0%
5/10	50% Attack	63.10 ± 1.45	60.0%	60.0%

5 Conclusion

In conclusion, this research introduced an X-FL framework integrated with a hybrid Blockchain and IPFS architecture for privacy-preserving optical fiber fault identification in smart grid systems. By establishing a clear architectural boundary between optical telemetry signaling and physical metallic power distribution, the framework addresses the operational dependencies of SCADA telemetry, PMU synchronization, and IEC 61850 GOOSE communications under optical link degradation. Across distributed OTDR trace sequences under Dirichlet non-IID data skew ($\alpha = 0.5$), the Adam-optimized LSTM architecture achieved a peak validation accuracy of 87.16% (with 87.95% precision, 87.07% recall, and an 86.78% F_1 -score) alongside a reduced validation loss of 0.2078. Explainable AI evaluations using SHAP and LIME aligned model decision boundaries with physical optical backscatter dynamics (loss, reflectance, and SNR), validated by low infidelity ($INFD = 0.021$) and sensitivity ($SENS = 0.014$) scores. Furthermore, the hybrid Solidity smart contract and IPFS scheme realized lightweight on-chain indexing (43,218 gas for model update registration) while enabling an off-chain validation-based screening mechanism that robustly tolerates up to 30% adversarial Byzantine updates without false rejection of benign clients.

This paper discusses optical telemetry failures and performance measures, but it does not simulate the actual behavior of power systems (e.g., circuit breaker movement, dynamic load flow in electric vehicles, transient voltage stability during communication failures, etc.). Future works will utilize dynamic co-simulation platforms (e.g., OpenDSS in conjunction with NS-3) to evaluate grid stability using closed-loop techniques. Additionally, the implementation of Secure Multi-Party Computation (SMPC) and Differential Privacy (DP) is suggested in order to ensure further data privacy with respect to curious anonymous aggregators.

Acknowledgement: None.

Funding Statement: The study was supported by the Russian Science Foundation (project No. 25-76-31013) at the Southern Federal University.

Author Contributions: Conceptualization: Rajesh Gupta, Sudeep Tanwar, and Wei-Chiang Hong; methodology: Lakshit Pathak, Bansri Choxi, Rajesh Gupta, and Svetlana Sushkova; software: Lakshit Pathak, Vidhi Ruparelia, Bansri Choxi, and Khushi Shah; validation: Rajesh Gupta, Sudeep Tanwar, and Ming-Wei Li; formal analysis: Lakshit Pathak, Vidhi Ruparelia, Rajesh Gupta, and Wei-Chiang Hong; investigation: Vidhi Ruparelia, Bansri Choxi, Khushi Shah, and Sudeep Tanwar; resources: Lakshit Pathak, Bansri Choxi, and Svetlana Sushkova; data curation: Rajesh Gupta, Sudeep Tanwar, Ming-Wei Li, and Wei-Chiang Hong; writing—original draft preparation: Lakshit Pathak, Vidhi Ruparelia, Bansri Choxi, and Khushi Shah; writing—review and editing: Rajesh Gupta, Sudeep Tanwar, Ming-Wei Li, and Svetlana Sushkova; visualization: Lakshit Pathak, Vidhi Ruparelia, Bansri Choxi, and Khushi Shah; supervision: Rajesh Gupta, Sudeep Tanwar, Wei-Chiang Hong, and Svetlana Sushkova; project administration, Rajesh Gupta, Sudeep Tanwar, and Wei-Chiang Hong; funding acquisition: Ming-Wei Li, and Wei-Chiang Hong. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: We used the publicly available data from Kaggle. Link: <https://www.kaggle.com/datasets/yogi2727/optical-fibre-fault-detection>.

Ethics Approval: Not applicable.

Conflicts of Interest: Given his role as Guest Editor of this journal, Wei-Chiang Hong, had no involvement in the peer review of this article and had no access to information regarding its peer review. Full responsibility for the editorial process for this article was delegated to another journal editor. The authors declare no conflicts of interest.

References

1. Xu K, Yuan C. A fault location analysis of optical fiber communication links in high altitude areas. *Electronics*. 2023;12(17):3728. doi:10.3390/electronics12173728.
2. Abdelli K, Grieser H, Ehrle P, Tropschug C, Pachnicke S. Reflective fiber fault detection and characterization using long short-term memory. *J Opt Commun Netw*. 2021;13(10):E32. doi:10.1364/jocn.423625.
3. Usman A, Zulkifli N, Salim MRB, Khairi K. Optical fiber fault detection and localization in an optical access network domain. *Sci Proc Ser*. 2020;2(1):7–11. doi:10.31580/sps.v2i1.1209.
4. Mozafar MR, Moradi MH, Amini MH. A simultaneous approach for optimal allocation of renewable energy sources and electric vehicle charging stations in smart grids based on improved GA-PSO algorithm. *Sustain Cities Soc*. 2017;32(2):627–37. doi:10.1016/j.scs.2017.05.007.
5. Das S, Acharjee P, Bhattacharya A. Charging scheduling of electric vehicle incorporating grid-to-vehicle and vehicle-to-grid technology considering in smart grid. *IEEE Trans Ind Appl*. 2020;57(2):1688–702. doi:10.1109/tia.2020.3041808.
6. Carcangiu S, Fanni A, Montisci A. Optimization of a power line communication system to manage electric vehicle charging stations in a smart grid. *Energies*. 2019;12(9):1767. doi:10.3390/en12091767.
7. Mahfouz MM, El-Sayed MA. Smart grid fault detection and classification with multi-distributed generation based on current signals approach. *IET Gener Transm Distrib*. 2016;10(16):4040–7. doi:10.1049/iet-gtd.2016.0364.
8. Pathak L, Shah K, Gupta R, Tanwar S, Sawhney S, Gupta A. DL-based fault detection in optical cables for smart grid and charging station energy transfer. In: *Proceedings of the 2024 IEEE 21st India Council International Conference (INDICON); 2024 Dec 19–21; Kharagpur, India*. p. 1–6.
9. Pathak L, Patel P, Upadhyay R, Gupta R, Tanwar S, Saleem K, et al. XAI-OptiEV: explainable DL framework for optical fiber fault detection to optimize energy efficiency at EV charging stations. In: *Proceedings of the 2025 10th International Conference on Smart and Sustainable Technologies (SpliTech); 2025 Jun 16–20; Bol and Split, Croatia*. p. 1–6.
10. Njoku JN, Nwakanma CI, Kim DS. Explainable data-driven digital twins for predicting battery states in electric vehicles. *IEEE Access*. 2024;12:83480–501.
11. Khaneghah MZ, Alzayed M, Chaoui H. Fault detection and diagnosis of the electric motor drive and battery system of electric vehicles. *Machines*. 2023;11(7):713. doi:10.3390/machines11070713.
12. Lucas F, Costa P, Batalha R, Leite D, Škrjanc I. Fault detection in smart grids with time-varying distributed generation using wavelet energy and evolving neural networks. *Evol Syst*. 2020;11(2):165–80.
13. Wen M, Xie R, Lu K, Wang L, Zhang K. FedDetect: a novel privacy-preserving federated learning framework for energy theft detection in smart grid. *IEEE Internet Things J*. 2022;9(8):6069–80. doi:10.1109/jiot.2021.3110784.
14. Li Y, Wei X, Li Y, Dong Z, Shahidehpour M. Detection of false data injection attacks in smart grid: a secure federated deep learning approach. *IEEE Trans Smart Grid*. 2022;13(6):4862–72.
15. Huong TT, Bac TP, Ha KN, Hoang NV, Hoang NX, Hung NT, et al. Federated learning-based explainable anomaly detection for industrial control systems. *IEEE Internet Things J*. 2021;8(7):5926–37. doi:10.1109/access.2022.3173288.
16. Zhang W, Lu Q, Yu Q, Li Z, Liu Y, Lo SK, et al. Blockchain-based federated learning for device failure detection in industrial IoT. *IEEE Internet Things J*. 2020;8(7):5926–37. doi:10.1109/jiot.2020.3032544.

17. Han J, Zhang X, Xie Z, Zhou W, Tan Z. Federated learning-based equipment fault-detection algorithm. *Electronics*. 2024;14(1):92. doi:10.3390/electronics14010092.
18. Ning Z, Chen H, Wang X, Wang S, Guo L. Blockchain-enabled electrical fault inspection and secure transmission in 5G smart grids. *IEEE J Sel Top Signal Process*. 2022;16(1):82–96. doi:10.1109/jstsp.2021.3120872.
19. Li M, Zhang K, Liu J, Gong H, Zhang Z. Blockchain-based anomaly detection of electricity consumption in smart grids. *Pattern Recognit Lett*. 2020;138(6):476–82. doi:10.1016/j.patrec.2020.07.020.
20. Abdelli K, Azendorf F, Tropschug C, Griesser H, Pachnicke S, Choo J. Dataset for optical fiber faults. Piscataway, NJ, USA: IEEE Dataport; 2022 [cited 2026 Sep 15]. Available from: <https://www.kaggle.com/datasets/yogi2727/optical-fibre-fault-detection>.
21. McMahan B, Moore E, Ramage D, Hampson S, Arcas BA. Communication-efficient learning of deep networks from decentralized data. In: *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*; 2017 Apr 20–22; Ft. Lauderdale, FL, USA. p. 1273–82.
22. Li Z, Lin T, Shang X, Wu C. Revisiting weighted aggregation in federated learning with neural networks. *arXiv:2302.10911*. 2023.
23. Reyad M, Sarhan AM, Arafa M. A modified adam algorithm for deep neural network optimization. *Neural Comput Appl*. 2023;35(23):17095–112. doi:10.1007/s00521-023-08568-z.
24. Lundberg SM, Lee SI. A unified approach to interpreting model predictions. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*; 2017 Dec 4–9; Long Beach, CA, USA. p. 4768–77.
25. Ribeiro MT, Singh S, Guestrin C. Why should I trust you?: explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*; 2016 Aug 13–17; San Francisco, CA, USA. p. 1135–44.
26. Zhao X, Shen D. FedSW: federated learning with adaptive sample weights. *Inf Sci*. 2024;654:119873. doi:10.1016/j.ins.2023.119873.