Al-Farouk journal of sciences
Volume (2), Issue (3) (JUNE, 2026)
ISSN: 3135-2359

مجلة الفاروق للعلوم
المجلد (2) ملحق العدد 3 يونيو 2026
ردمد: 3135-2359

XFed-PQD: Explainable Federated Learning for Power Quality Disturbance Classification in Edge-Enabled Smart Distribution Networks

Abdulbari Ali Mohamed Frei [1], Saad Mohamed Eshtewi [2]

[1] Higher Institute of Sciences and Technology Msallatah, Libya
[2] Higher Institute of Sciences and Technology Msallatah, Libya
freiabd@gmail.com[1] , Saadeshtewi@yahoo.co.uk[2]

Received: 29/6/2026 Revised: 20/7/2026 Accepted: 28/7/2026 Publication: 8/8/2026

Abstract

Power-quality disturbance (PQD) recognition is increasingly being pushed toward the sensing point rather than handled only at a central server. That shift creates a useful opportunity for distributed learning, but it also exposes two practical questions: how much performance is lost when raw waveforms remain local, and whether the resulting decisions can still be interpreted. This paper examines both questions through XFed-PQD, an explainable federated-learning framework for 17 single and compound PQD categories in edge-enabled smart distribution networks. The framework combines a compact multi-branch one-dimensional convolutional network with FedAvg and FedProx training, temporal attribution, robustness testing, and deployment profiling. The controlled experiment used three seeds, 17,000 reference waveforms, ten fully participating clients, 15 centralized epochs, and 25 federated rounds. Centralized training produced 91.35% ± 0.70% accuracy and 91.33% ± 0.70% macro-F1. With FedAvg and an IID partition, the same model reached 83.89% ± 0.78% accuracy and 83.85% ± 0.89% macro-F1 while exchanging 47.57 MB of model traffic, 83.4% less than the CNN-LSTM benchmark. Client heterogeneity had a much larger effect: Dirichlet label skew reduced macro-F1 to 69.41% ± 1.70%, and FedProx reached 69.67% ± 2.05%. The resulting 0.26-percentage-point average gain was small and did not point in the same direction for every seed. In the extended Seed-42 analysis, local-to-global Integrated-Gradients cosine similarity was 0.924 ± 0.053. The same checkpoint was highly sensitive to added noise, with macro-F1 falling from 83.82% on clean data to 54.97%, 17.01%, and 3.71% at 40, 30, and 20 dB. Dynamic INT8 quantization changed macro-F1 by -0.13 percentage points, reduced serialized size by 3.44%, and increased mean CPU latency. The dataset audit also uncovered a repeated healthy waveform that crossed data splits and appeared once under the Swell label; for that reason, three-seed sensitivity results excluding the true healthy class are reported. Taken together, the results show that explainable federated PQD monitoring is feasible under the evaluated conditions, while also making clear that client heterogeneity, duplicated waveforms, noise sensitivity, and hardware-dependent runtime behavior remain central concerns.

Keywords— power quality disturbances; federated learning; FedAvg; FedProx; explainable artificial intelligence; Integrated Gradients; Grad-CAM; edge intelligence; dynamic quantization; non-IID data.

1. Introduction

Distribution networks now combine inverter-interfaced generation, nonlinear loads, electric-vehicle chargers, and a growing amount of electronically controlled equipment. Their switching behavior affects both waveform shape and frequency content, so power quality can no longer be treated only as a local compliance issue. The practical consequences are different from one event to another: a short sag may interrupt a process line, persistent harmonics may accelerate thermal aging, and a compound disturbance may be harder to separate than either constituent event. For automated monitoring, the classifier therefore has to distinguish closely related signatures quickly and consistently, including cases in which disturbances overlap.

Deep learning has improved PQD recognition because discriminative features can be learned directly from time-domain samples or time-frequency representations [26], [28]-[43]. The usual centralized workflow still assumes, however, that measurements collected at meters, substations, or industrial sites can be moved to one repository. That assumption is often inconvenient in practice. Waveforms may expose operating schedules, equipment behavior, or customer activity; available bandwidth differs across sites; and concentrating the data in one location creates an additional security and governance burden. Federated learning (FL) takes a different route by keeping training close to the data and aggregating model updates rather than routinely transferring the source waveforms [1]-[25]. FL is not, by itself, a complete privacy solution, but it provides a useful distributed-learning architecture and reduces the normal movement of raw measurements.

Classification performance is only part of the operational question. An operator also needs some indication that the model is reacting to the disturbed portion of a waveform rather than to an incidental background feature. Integrated Gradients (IG) and Grad-CAM can reveal which temporal samples contributed to a decision [46], [47], and toolkits such as Quantus encourage quantitative checks instead of relying on a heatmap alone [48]. Deployment introduces a second practical constraint. At an intelligent electronic device or gateway, model size, communication traffic, inference latency, and memory behavior all matter. Quantization may reduce some of these costs, although the actual benefit depends on the operators present in the model and on the target hardware [44], [45], [49]. For that reason, a high accuracy value by itself is not enough to establish readiness for use at the edge.

The literature contains strong individual pieces of this problem, but less evidence that evaluates them together. FL has already been studied in smart-grid forecasting, anomaly detection, false-data-injection problems, and distributed-energy applications [4]-[25]. More directly, a 2025 study applied FL to PQD classification and examined robustness to poisoning, confirming that federated training is relevant to this task [27]. Other recent PQD studies have concentrated on interpretability, explainable classifiers, compact networks, or embedded execution [29]-[33], [44]. Within the literature reviewed for this study, we did not identify one work that jointly examined IID and label-skewed federated optimization, temporal explanation consistency, noise robustness, and FP32/INT8 deployment on a 17-class PQD problem. This is a statement about the coverage of the reviewed corpus, not a claim of absolute priority.

XFed-PQD was developed to investigate that integration gap. Four contributions are evaluated. The first is a compact multi-branch one-dimensional architecture that retains short- and medium-range temporal structure with 22,689 parameters. The second is a matched comparison of centralized training and federated training under IID and Dirichlet label-skew partitions, including FedAvg and FedProx under controlled conditions. The third contribution adds IG and one-dimensional Grad-CAM and then measures how closely client explanations agree with the global model. Finally, communication volume, serialized size, CPU latency, throughput, and dynamic INT8 behavior are profiled together with predictive accuracy, rather than being reported as disconnected engineering measurements.

The data audit forms another important part of the study. In the reference collection, the 1,000 records labeled as healthy are exact copies of a single waveform, and that same waveform appears once under a disturbance label. With a random stratified split, identical signal content therefore enters the training, validation, and test subsets. This limitation is quantified rather than hidden. In addition to the primary results, the paper reports a sensitivity calculation in which the true healthy class is excluded from the macro metrics. The audit is important because it changes the interpretation of the otherwise near-perfect healthy-class performance.

The rest of the paper proceeds as follows. Section 2 summarizes the relevant work on PQD classification, federated intelligence, explainability, and edge deployment. Section 3 states the distributed classification problem, and Section 4 describes XFed-PQD. The dataset, integrity checks, execution protocol, and evaluation metrics are given in Section 5. Section 6 presents the measured results, followed by the research-question discussion and practical implications in Section 7. Limitations and future directions are collected in Section 8, and Section 9 closes the paper.

2. Background and Related Work

2.1 Power-quality disturbance classification

Power-quality disturbances cover magnitude changes, interruptions, waveform distortion, flicker, notching, and transient events. The compound cases are especially difficult because the relevant signatures do not all live on the same temporal or spectral scale. A representation that reacts strongly to an abrupt transient may give less emphasis to periodic flicker; conversely, a long receptive field can smooth a short high-frequency event. As a result, the way signals are generated and labeled directly affects the validity of any classifier comparison. Machlev et al. addressed reproducibility by introducing an open PQD dataset generator with controlled waveform synthesis and disturbance combinations [26]. The present study works with a 17-class reference collection indexed alphabetically and, in addition to classification, audits exact duplicates and cross-label collisions in those files.

Recent PQD classifiers span convolutional, recurrent, attention-based, kernel, and decomposition-based designs. Vision Transformers and image-like signal encodings have been tested for multi-class recognition [28]. Interpretable DWT-1DCNN-LSTM models combine an engineered decomposition stage with learned temporal features [29], while CNN-LSTM variants remain a useful sequential benchmark [31]. Other examples include multi-scale CNNs [34], CNN-GRU hybrids [36], ensemble CNNs [38], and multimodal models [39]. Classical approaches based on support-vector machines, wavelet or variational-mode decomposition, and optimized feature selection are still relevant when data or computation is limited [40]-[43]. The variety is useful, but it also makes headline accuracies difficult to compare across papers because class definitions, disturbance generators, split rules, noise assumptions, and reported metrics are rarely identical.

A smaller set of studies has focused explicitly on deployment. Bai et al. proposed a lightweight PQD model aimed at edge implementation [30], while Khaldi et al. demonstrated a one-dimensional CNN in a low-cost smart-meter setting [44]. These studies make compact local inference credible, but they do not remove the need for device-specific timing. Parameter count is only one factor in runtime: parallel branches, depthwise kernels, memory access, and poorly optimized operators can make a small model slower than expected on a particular backend.

2.2 Federated learning in power and energy systems

FedAvg provides the standard synchronous FL pattern: the server distributes a global model, the selected clients optimize that model on their local records, and the returned parameters are combined with a sample-weighted average [1]. FedProx keeps this structure but adds a proximal term to discourage a client model from moving too far from the received global state, which can be useful when local objectives differ [2]. Power and energy applications have since adapted this collaboration pattern to load forecasting, distributed monitoring, and infrastructure analytics [3]-[8]. The key difference from pooled learning is that federated clients may hold unequal sample counts, different label mixtures, and site-specific measurement conditions, so their local objectives need not resemble one another.

Smart-grid FL research has moved beyond early frameworks and surveys [4]-[9] toward more specialized work in forecasting, anomaly detection, security, and resource-aware coordination. Studies of non-IID solar generation and demand forecasting show, in particular, that statistical heterogeneity can dominate the behavior of aggregation [10]-[14]. Other approaches address confidentiality, personalization, and distribution mismatch [12], [15], [16], while anomaly detection and false-data-injection defense have also been studied in federated settings [17], [18]. Edge-oriented work then connects these ideas to smart meters and gateways [19]-[25]. Taken together, the literature supports FL as an appropriate architecture for distributed grid analytics, but it also makes clear why one IID experiment is not enough to characterize a distributed monitoring system.

For the PQD task itself, Alsabaan et al. compared centralized and federated learning and also studied adversarial poisoning [27]. Among the sources reviewed here, it is the closest domain-specific precedent. The emphasis of XFed-PQD is different: the present evaluation combines a compact classifier, IID and Dirichlet label-skew partitions, a controlled FedAvg-FedProx comparison, explanation consistency, additive-noise sensitivity, and dynamic quantization in one pipeline. Claims are limited to the methods that were actually executed in all three training runs.

2.3 Explainability and explanation evaluation

Integrated Gradients attributes a prediction at the input level by accumulating gradients along a straight path from a baseline to the observed sample [46]. One useful property is completeness, which connects the summed attributions to the change in the model output between the two endpoints. Grad-CAM works at a different level. It weights convolutional activation maps by class-score gradients and produces a coarser class-discriminative localization [47]; in one dimension, that localization becomes a temporal relevance curve. The two views are therefore complementary. IG preserves input-level detail, whereas Grad-CAM reflects the support carried by an internal convolutional representation.

A visually convincing explanation is not automatically a trustworthy one. Quantus groups evaluation criteria such as faithfulness, robustness, localization, complexity, and sensitivity to randomization [48], and broader work on smart-energy governance treats explainability as only one part of accountability and risk management [50]. Within PQD research, explainable classifiers and uncertainty-aware analyses have begun to connect model evidence with waveform morphology [29], [32], [33]. Here, explanation quality is approached through temporal sparsity and agreement, using both cosine similarity and rank correlation between local and global IG vectors. The reference data do not include sample-level event masks, so localization IoU and area under the precision-recall curve are deliberately not reported. Without such masks, a plausible heatmap should not be presented as a validated physical event detector.

2.4 Edge intelligence and quantization

Edge intelligence places analytics across meters, gateways, and control infrastructure rather than at one central location [19]-[25]. In a federated system, that choice introduces two recurring costs: local computation and repeated model exchange. A compact serialized checkpoint can reduce the second cost, but production traffic would also contain protocol overhead, encryption, retransmissions, orchestration messages, and other metadata. The communication figures reported here therefore count only bidirectional model-state transfer for participating clients. They are intended as a matched accounting measure, not as a reconstruction of a complete production network stack.

Dynamic INT8 quantization can be applied after training without a separate calibration set, but it typically converts supported linear or recurrent weights while many convolutional operators remain in

floating point [49]. That detail is especially important for a CNN-dominant architecture. In such a model, the technically correct outcome may be a small file-size reduction and no speedup, even though the label 'INT8' can suggest a fourfold reduction in storage or arithmetic. The broader measurement context discussed by NREL [45] supports the same practical point: deployment metrics have meaning only when they are tied to a specific monitoring environment.

Table 1. Selected literature and the coverage gap addressed by XFed-PQD.

Study	Domain	Training	Heterogeneity	XAI	Edge	Primary focus
McMahan et al. [1]	General	FedAvg	Partly	No	No	Foundational averaging
Li et al. [2]	General	FedProx	Yes	No	No	Client-drift control
Bousbiat et al. [7]	Smart grid	Survey	Yes	Limited	Yes	FL–grid research map
Zheng et al. [8]	Power systems	Survey	Yes	Limited	Yes	Power-service synthesis
Li et al. [19]	Smart meters	Split/FL	Yes	No	Yes	Edge intelligence
Alsabaan et al. [27]	PQD	FedAvg	IID focus	No	No	Poisoning robustness
Yang et al. [29]	PQD	Centralized	N/A	Yes	No	DWT–CNN–LSTM
Altun et al. [32]	PQD	Centralized	N/A	Yes	No	Explainable classification
Bai et al. [30]	PQD	Centralized	N/A	No	Yes	Lightweight edge model
Khaldi et al. [44]	PQD	Centralized	N/A	No	Yes	Embedded 1D CNN
XFed-PQD	PQD	FedAvg/FedProx	IID + skew	IG + Grad-CAM	FP32/INT8	Integrated evaluation

N/A: not applicable to centralized training. Source: Authors' synthesis of [1], [2], [7], [8], [19], [27], [29], [30], [32], and [44].

3. System Model and Problem Formulation

3.1 Edge-enabled monitoring architecture

Assume a distribution network with K edge clients. Depending on the deployment, one client can correspond to a smart meter, an intelligent electronic device, a feeder gateway, or an organizational site. Client k stores $D_k = \{(x_i, y_i)\}$, where each x_i is a normalized voltage waveform of length L=100 and y_i is one of C=17 PQD labels. During federated optimization, the raw records remain with the client. The coordinator sends model parameters, receives the locally updated states, aggregates them, and distributes the new global model. This arrangement reduces routine raw-waveform movement. It does not, however, provide formal differential privacy, secure aggregation, or protection against information leakage from shared gradients.

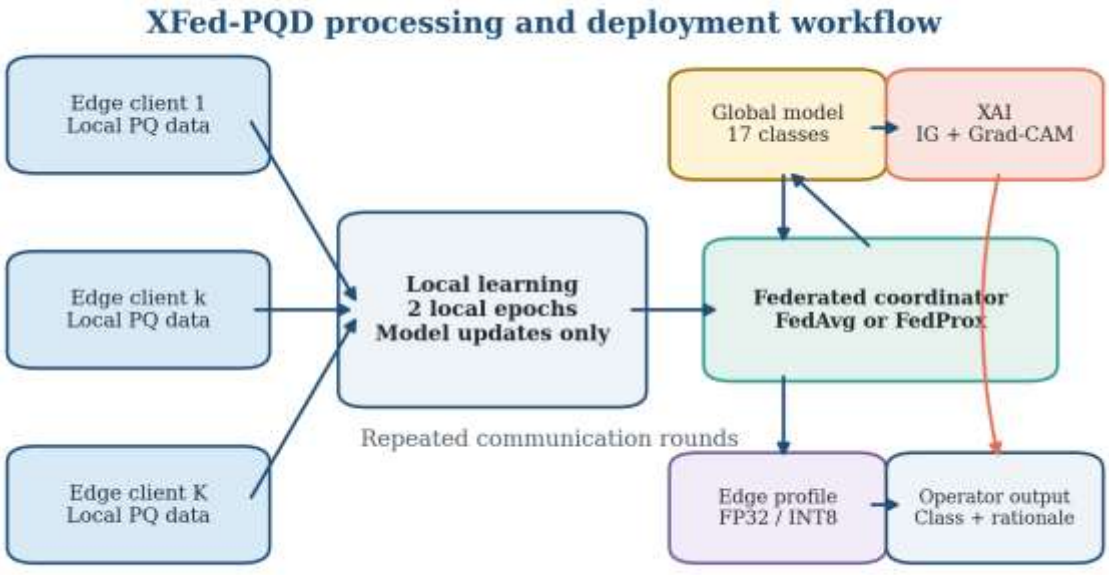

Figure 1. XFed-PQD workflow from local waveform learning to global prediction, explanation, and edge profiling.
Source: Authors' construction.

The deployment path returns more than a class label. One output is the prediction used by an operator or a downstream control application; the other is a temporal rationale produced with IG or Grad-CAM. In parallel, the deployment profiler records serialized model size and inference behavior for the FP32 and dynamically quantized variants. Keeping these outputs separate is useful in practice because 'What was predicted?' and 'Which part of the waveform influenced that prediction?' are different questions.

3.2 Learning objective

Let w represent the global model parameters and let $n_k=|D_k|$. Client k minimizes its empirical cross-entropy objective $F_k(w)$, giving the population-weighted federated objective

$$\min_{\mathbf{w}} F(\mathbf{w}) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(\mathbf{w}), \qquad F_k(\mathbf{w}) = -\frac{1}{n_k} \sum_{i=1}^{n_k} \log p_{\mathbf{w}}(y_i \mid \mathbf{x}_i) \tag{1}$$

Here, $n = \sum_k n_k$. The centralized reference exposes all training records to one optimizer and minimizes the pooled loss with the same architecture. Its purpose is to isolate the performance cost associated with distributed optimization; it is not a privacy-preserving baseline.

Predictive performance is summarized with accuracy, macro precision, macro recall, macro-F1, class-wise recall and F1, cross-entropy loss, and a macro one-versus-rest false-alarm rate. These measures are supplemented by communication and edge metrics. For the explanation stage, the reported quantities describe attribution sparsity and agreement between attribution vectors.

3.3 Research questions

Five research questions guide the experiments. RQ1 measures how much of the centralized performance is retained when the proposed architecture is trained with FedAvg on IID client partitions. RQ2 focuses on Dirichlet label skew and asks whether FedProx changes the observed heterogeneous result. RQ3 examines whether local client explanations resemble the explanation produced by the global model. RQ4 tests the selected model under additive white Gaussian noise. Finally, RQ5 considers the trade-off among model traffic, serialized size, latency, and predictive quality for FP32 and dynamically quantized INT8 execution.

4. Proposed XFed-PQD Framework

4.1 Signal preprocessing and class indexing

The loader reads 17 CSV files in alphabetical filename order, with one 100-sample waveform in each row. Each signal is normalized independently so that absolute scale differences are removed while its relative waveform shape is retained. Class-stratified splitting then assigns 70% of the data to training, 10% to validation, and 20% to testing. The validation subset is used for checkpoint selection; the independent test subset is held back for final evaluation. Class indices are fixed from the alphabetical file mapping before any client partitioning, so every client, checkpoint, and report uses the same label order.

Table 2. Alphabetical class-index mapping used by the loader and all reported models.

Index	Class	Physical description
0	Flicker	Periodic amplitude modulation
1	Flicker_with_Sag	Flicker combined with voltage reduction
2	Flicker_with_Swell	Flicker combined with voltage increase
3	Harmonics	Non-sinusoidal periodic distortion
4	Harmonics_with_Sag	Harmonic distortion plus sag
5	Harmonics_with_Swell	Harmonic distortion plus swell
6	Interruption	Near-total short-duration voltage loss
7	Notch	Short periodic voltage indentation
8	Oscillatory_Transient	Decaying oscillatory transient
9	Pure_Sinusoidal	Nominal sinusoidal waveform
10	Sag	Short-duration RMS voltage reduction
11	Sag_with_Harmonics	Sag combined with harmonics
12	Sag_with_Oscillatory_Transient	Sag plus oscillatory transient
13	Swell	Short-duration RMS voltage increase
14	Swell_with_Harmonics	Swell combined with harmonics
15	Swell_with_Oscillatory_Transient	Swell plus oscillatory transient
16	Transient	Short non-oscillatory impulsive event

Compound class names preserve the stored dataset labels. Source: Authors' extraction from the executed reference CSV files and benchmark metadata [27]; physical descriptions follow standard PQD terminology summarized in [26].

4.2 Lightweight multi-branch classifier

The proposed classifier starts with a 1×5 convolution that expands the single input channel to 24 feature maps. The stem output is then sent through three depthwise-separable branches with kernel sizes 3, 5, and 7. Their outputs are concatenated to form 72 channels, fused to 64 channels with a 1×1 convolution, and reweighted by channel attention. Two further depthwise-separable blocks raise the channel count to 72 and then 96 while reducing temporal resolution by a factor of two at each stage. Adaptive average pooling, dropout of 0.15, and a 96-to-17 linear layer produce the final logits.

Using three kernel widths lets the network respond to brief edges, intermediate oscillations, and slower local morphology without introducing a large recurrent block. The full model contains 22,689 trainable parameters and requires approximately 1.109 million multiply-accumulate operations for one 100-sample waveform. The CNN-LSTM benchmark is larger: two convolutional blocks, a 144-unit LSTM, and a two-layer classifier give 143,121 parameters and roughly 3.334 million multiply-accumulate operations.

4.3 Centralized reference training

Centralized training uses Adam with a learning rate of 10^{-3}, weight decay of 10^{-4}, a batch size of 128, and cross-entropy loss. The maximum training budget is 15 epochs. The retained checkpoint is the one with the best validation macro-F1, and early stopping uses a patience of four epochs. Proposed and benchmark models follow the same preprocessing and data-split logic.

4.4 Federated training process

Federated training runs synchronously for 25 rounds, with all ten clients participating in every round. The coordinator begins by broadcasting the current global parameters. Each client then performs two local epochs with Adam and returns its updated model state. After aggregation, the candidate global model is evaluated on the validation subset. Whenever validation macro-F1 improves, that state becomes the retained checkpoint. The final test metrics therefore come from the best validation checkpoint observed within the 25-round budget, not automatically from the last round.

FedAvg combines the returned client states with a sample-weighted parameter mean:

$$\mathbf{w}^{(t+1)} = \sum_{k \in S_t} \frac{n_k}{n_{S_t}} \mathbf{w}_k^{(t+1)} \tag{2}$$

In (2), S_t denotes the clients participating in round t and n_s is their total number of samples. Because participation is complete in this experiment, S_t contains all ten clients in every round.

For FedProx, each local objective also includes a quadratic penalty around the downloaded global parameters $w^{(t)}$:

$$F_k^{\text{prox}}(\mathbf{w}) = F_k(\mathbf{w}) + \frac{\mu}{2} \| \mathbf{w} - \mathbf{w}^{(t)} \|_2^2 \quad (3)$$

The coefficient is μ=0.01. At the start of local training the penalty is zero; it grows as the client parameters move away from the received global state. FedProx is therefore being used here to limit client drift, not as a direct class-imbalance correction. Within a seed, FedAvg and FedProx use the same local-epoch budget, participation pattern, partition, initialization policy, and number of communication rounds.

4.5 IID and label-skew partitioning

For the IID condition, stratified training records are distributed approximately evenly across the ten clients. Heterogeneity is introduced by drawing class proportions from a Dirichlet distribution with concentration alpha=0.5. At this concentration, individual clients can become dominated by only part of the label set while other labels are sparsely represented. Figure 2 summarizes those allocations. The important control is that, within each seed, FedAvg and FedProx receive the same realized label-skew partition and the same deterministic initialization policy. A new partition is generated for each seed, so the three-seed summary reflects three independently realized client allocations.

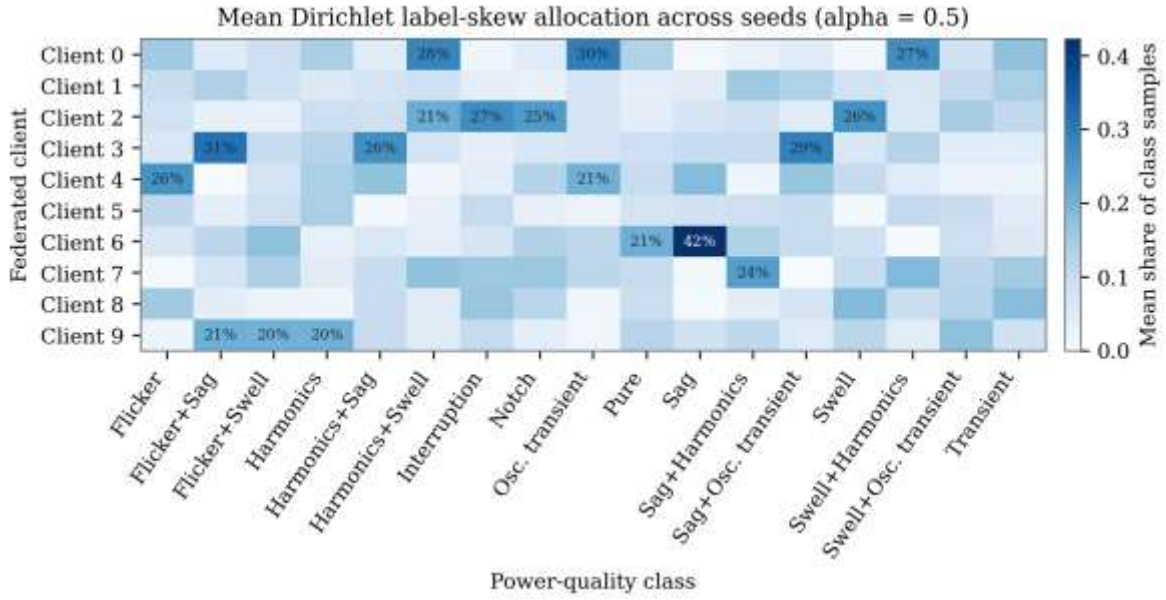

Figure 2. Mean ten-client class allocation across the three Dirichlet label-skew partitions (alpha=0.5).

Source: Authors' calculations from the Seed-42, Seed-52, and Seed-62 client partitions.

4.6 Explainability module

For class score f_c and input x, IG uses baseline x′ with M=12 integration steps:

$$\text{IG}_i(\mathbf{x}) = (x_i - x_i') \frac{1}{M} \sum_{m=1}^{M} \frac{\partial f_c\left(\mathbf{x}' + \frac{m}{M}(\mathbf{x} - \mathbf{x}')\right)}{\partial x_i} \quad (4)$$

For each explanation method, three test examples are taken from every class, giving 51 IG records and 51 one-dimensional Grad-CAM records. Sparsity is computed after attribution normalization to describe how concentrated the relevance is. Consistency is evaluated separately: one example per class is passed through the global model and through ten locally trained client models. IG vectors are then compared with cosine similarity and Spearman rank correlation. The former captures agreement in overall direction; the latter asks whether the sample-wise relevance ordering is similar.

Grad-CAM forms its temporal map by computing a class-specific weight for every channel in the selected convolutional activation and combining those channels after rectification:

$$L^c = \mathrm{ReLU}\left(\sum_k \alpha_k^c A^k\right), \qquad \alpha_k^c = \frac{1}{Z}\sum_t \frac{\partial y^c}{\partial A_t^k} \tag{5}$$

In (5), A^k is channel k of the selected activation tensor, Z is its temporal length, and y^c is the class score. Because Grad-CAM acts on an internal feature map, its output is coarser than the input-level IG vector and must be aligned back to the 100-sample waveform. The two explanation methods are used as complementary views rather than as interchangeable estimates.

4.7 Edge profiling and dynamic quantization

The selected FedAvg-IID checkpoint is saved once in FP32 and again after dynamic INT8 quantization. CPU timing uses batch size one, three warm-up executions, and 30 measured repetitions. Mean latency and its standard deviation are reported, while throughput is obtained from the reciprocal rate in samples per second. The benchmark is profiled in exactly the same way. Communication traffic is computed from the serialized model state transmitted to and returned from ten clients over 25 rounds. Optimizer state, protocol headers, encryption, retransmissions, and secure-aggregation traffic are outside this accounting.

4.8 Execution states and checkpoint semantics

A run is recorded as a sequence of auditable states rather than reduced to one final score. Under the same run identifier, the pipeline stores the configuration, model initialization, client allocation, validation history, selected checkpoint, test predictions, confusion matrices, attribution arrays, and deployment measurements. This separation matters because the stages serve different purposes. Validation selects the model state; the test set estimates final performance; and the explanation and edge analyses must load that same selected state if they are to describe the classifier reported in the results.

The federated checkpoint is not assumed to be the model from round 25. After every aggregation, the candidate global state is evaluated on validation macro-F1. An improvement replaces the retained checkpoint; otherwise, the previous best state remains in place. This avoids a situation in which a late unstable round silently displaces a stronger earlier model. Accordingly, '25-round model' refers to the optimization budget. The checkpoint evaluated on the test set is the best state observed within those 25 rounds.

The client models used in the Seed-42 consistency analysis come from that same global execution, so local-global similarity compares related models rather than separately initialized networks. The label-skew comparison is paired in the same way: FedAvg and FedProx reuse the identical realized client allocation within a seed. These controls remove avoidable sources of comparison noise. Repeating the experiment over three seeds then exposes the remaining sensitivity to the train/validation/test split, the client partition, and stochastic initialization.

For each of Seeds 42, 52, and 62, the completed matrix contains six reportable training scenarios. The additional noise, explainability, and edge analyses were run once, using the selected Seed-42 FedAvg-IID checkpoint. They were not repeated and then presented as independent training evidence. Tables therefore identify explicitly whether a result is a three-seed aggregate or part of the extended Seed-42 evaluation.

5. Experimental Methodology

5.1 Dataset and disturbance categories

The reference collection contains 17,000 waveforms, with 1,000 labeled examples in each of 17 classes. Every waveform has 100 samples; the associated sampling rate is 5 kHz and the fundamental frequency is 50 Hz. The class set contains isolated events as well as combinations such as flicker with sag, harmonics with swell, and sag with an oscillatory transient. Although the collection is globally balanced, these compound labels make macro metrics useful because overall accuracy can hide a class-specific weakness.

The global split is created before client assignment. Test examples stay outside training, while the development portion supplies the client records and a held-out validation subset. The centralized baseline uses the same training and validation records after pooling them. This gives an internally matched comparison between centralized and federated training on the supplied data. It does not make the resulting numbers directly comparable with studies that use different generators, signal durations, class sets, noise assumptions, or split procedures.

5.2 Integrity audit and leakage control

Exact waveform hashes were compared within each class, across classes, and across the three data splits. Sixteen classes contain 1,000 distinct rows. Pure_Sinusoidal is different: its file contains the same waveform 1,000 times. One exact copy of that waveform is also found in Swell. The 17,000 labeled records therefore correspond to 16,000 unique waveform vectors, with one duplicate group of multiplicity 1,001 that spans two labels.

Because the healthy vector is repeated, random stratification cannot keep identical signal content out of training, validation, and testing at the same time. The hash intersections confirm one shared signal value for every pair of splits. A group-aware split alone would still leave a problem, since one exact input has two labels and would require a curation decision. The primary experiments retain the original protocol so that the execution remains reproducible. As a diagnostic check, aggregate metrics are also recalculated after excluding samples whose true class is Pure_Sinusoidal. The possibly mislabeled Swell instance is not removed from training, and that residual limitation is stated explicitly.

Table 3. Exact-waveform audit and sensitivity metrics excluding the true healthy class.

Category	Audit item	Measured value	Implication
Collection	Labeled records	17,000	1,000 per class
Exact hashes	Unique waveforms	16,000	One duplicate group has multiplicity 1,001
Healthy class	Unique Pure_Sinusoidal rows	1 / 1,000	999 within-class duplicates
Cross-label	Exact match	Pure ↔ Swell	The same vector has conflicting labels
Split overlap	Shared hashes	1 per split pair	Identical content crosses train/validation/test
Sensitivity	Central proposed macro-F1	90.93 ± 0.73%	True Pure rows excluded; full metric 91.33 ± 0.70%
Sensitivity	FedAvg-IID proposed macro-F1	83.25 ± 0.85%	True Pure rows excluded; full metric 83.85 ± 0.89%
Sensitivity	FedAvg-skew proposed macro-F1	68.26 ± 1.87%	True Pure rows excluded; full metric 69.41 ± 1.70%
Sensitivity	FedProx-skew proposed macro-F1	68.52 ± 2.25%	True Pure rows excluded; full metric 69.67 ± 2.05%

The original splits are retained for reproducibility; sensitivity metrics do not remove the conflicting Swell training record. Source: Authors' hash audit and recomputation from the three saved confusion matrices.

5.3 Executed scenarios

Each of the three seeds executes six main training scenarios: centralized CNN-LSTM, centralized proposed model, FedAvg-IID CNN-LSTM, FedAvg-IID proposed model, proposed FedAvg with Dirichlet label skew, and proposed FedProx on the same within-seed label-skew partition. Seeds 42, 52, and 62 each regenerate the stratified data split, the client allocation, and the stochastic training trajectory. In total, the aggregate predictive analysis is therefore based on 18 trained scenario instances.
The extended analyses are attached to one selected model rather than averaged over all three seeds. Specifically, the Seed-42 proposed FedAvg-IID checkpoint is used for additive-noise testing, explainability, client-consistency analysis, and edge profiling. RQ1 and RQ2 are therefore supported by repeated training runs, whereas RQ3-RQ5 describe one reproducible selected checkpoint in greater detail.

5.4 Training configuration

Table 4. Executed training, explanation, robustness, and profiling configuration.

Component	Setting
Dataset	17 CSV files; 17,000 waveforms; 100 samples each
Signal context	5 kHz sampling; 50 Hz fundamental
Clients	10; full participation; no simulated dropout
Split	70% training; 10% validation; 20% testing (stratified)
Partition	IID or Dirichlet label skew, α=0.5
Independent seeds	42, 52, and 62
Federated rounds	25
Local optimization	2 epochs/round; Adam; learning rate 10^{-3}
Batch / regularization	128; weight decay 10^{-4}
FedProx	μ=0.01
Central training	15 epochs maximum; patience 4
Checkpoint	Highest validation macro-F1
Extended analysis	Seed 42: XAI, AWGN robustness, and FP32/INT8 profiling
XAI	IG: 12 integration steps; 3 samples/class; Grad-CAM 1D
Consistency	1 sample/class across the global and 10 final client models
Noise	AWGN at 40, 30, and 20 dB
Edge timing	Batch 1; 3 warm-ups; 30 repetitions; CPU
Execution	Python 3.12.13; PyTorch 2.5.0+cpu; Linux x86-64

The three-seed training matrix is complete; extended XAI, noise, and edge stages use the selected Seed-42 FedAvg-IID checkpoint. Source: Executed configuration files and experiment manifests.

5.5 Evaluation metrics

Macro precision, recall, and F1 give every class equal weight, which is useful when a client partition is locally imbalanced. For class c, FAR is the fraction of negative examples that are incorrectly assigned to that class:

$$\mathrm{FAR}_c = \frac{\mathrm{FP}_c}{\mathrm{FP}_c + \mathrm{TN}_c}, \qquad \text{Macro-FAR} = \frac{1}{C}\sum_{c=1}^{C} \mathrm{FAR}_c \tag{6}$$

Macro-FAR is the unweighted mean of FAR_c over the 17 one-versus-rest confusion matrices; it is not 1-accuracy. Class-wise recall and F1 are retained so that specific failure modes remain visible. Cross-entropy loss is also reported because models with similar hard predictions can still differ substantially in confidence.
Communication is measured in megabytes of model traffic using the bidirectional accounting rule stated above. The edge profile includes parameter count, multiply-accumulate operations, serialized size, latency, throughput, and the macro-F1 change after quantization. For explainability, the study reports

mean sparsity, local-global cosine similarity, cross-client cosine similarity, and local-global Spearman correlation. No event-mask localization metric can be computed from the available labels.

5.6 Convergence and statistical interpretation

Validation macro-F1 determines which checkpoint is retained. The curves shown in this study cover a fixed budget of 25 federated rounds, so they describe the training behavior that was actually observed; they are not evidence of asymptotic convergence. Results across the three seeds are summarized as arithmetic mean ± sample standard deviation. We also report 95% confidence intervals using the t distribution with two degrees of freedom. With n=3, these intervals are unavoidably wide and are best read as a rough indication of uncertainty rather than as precise estimates of a population value.

The hypothesis tests are included for context, but they are not asked to carry more weight than the experiment can support. Pairwise contrasts use the matched values from the three seeds in an exact Wilcoxon signed-rank test, and a Friedman test is used to screen the four proposed-model settings together. At this sample size, exact p-values can take only a small number of values. For that reason, the interpretation focuses first on the seed-wise differences: how large they are and whether their direction is consistent. If a small mean advantage changes sign in one seed, we do not treat it as convincing evidence that one optimizer is superior.

5.7 Reproducibility and comparison controls

Several comparison controls were kept fixed so that a change in method was not confounded with an avoidable change in data handling or initialization. The class order is created once from the alphabetically sorted CSV filenames and reused throughout the pipeline. Centralized and federated runs use the same test records, the same normalization procedure, the same loss definition, and the same evaluation path. Deterministic initialization settings are also held consistent within a given architecture and partition. For the label-skew experiments, FedAvg and FedProx are evaluated on the same realized client allocation; drawing a second Dirichlet partition would introduce another source of difficulty that has nothing to do with the optimizer itself.

Checkpoint selection never uses the test set. The model is selected with validation macro-F1, and only then are the held-out test predictions passed through the common evaluation routine that produces overall metrics, class-wise reports, and confusion matrices. FAR is computed from the same predictions used for accuracy and macro-F1, which avoids the incorrect shortcut of treating FAR as 1-accuracy. The edge experiment follows the same paired logic: it reloads the selected checkpoint, evaluates the same test labels before and after quantization, and reports the resulting F1 change from those matched evaluations.

Reproducibility also requires stating what the experiment does not control. Timing was recorded on a Linux x86-64 CPU environment rather than on a physical edge gateway, so elapsed time can change with backend kernels and system load. Three completed seeds provide a clearer picture than a single run, but they still leave only a small basis for statistical inference. Each seed therefore keeps its configuration, split indices, client partitions, training histories, checkpoints, confusion matrices, and final metrics. This makes it possible to trace an aggregate number back to the individual execution that produced it.

Published PQD studies are used here mainly to compare scope and methodology, not to construct a leaderboard of accuracy values. Different papers may rely on wavelet images, different synthetic equations, longer waveform windows, alternative class definitions, or noise introduced at another point in the data pipeline. A higher accuracy reported under those conditions does not establish that the same classifier would perform better on the present test set. Table 1 therefore compares research coverage, Tables 5-8 summarize the matched three-seed experiments, and Tables 9-10 report the extended analyses performed on Seed 42.

6. Results and Analysis

6.1 Centralized classification performance

Over Seeds 42, 52, and 62, the centralized proposed model produced 91.35% ± 0.70% accuracy and 91.33% ± 0.70% macro-F1; its macro FAR averaged 0.540% ± 0.043%. The centralized CNN-LSTM benchmark was lower and much more variable, with 84.64% ± 4.73% accuracy, 84.36% ± 5.01% macro-F1, and 0.960% ± 0.295% FAR. Averaged over the three seeds, the proposed architecture gained 6.97 ± 4.99 macro-F1 percentage points. The direction of the gain was the same in every run, but not its size: +4.36 points for Seed 42, +3.81 for Seed 52, and +12.72 for Seed 62. That spread, together with the wide benchmark confidence interval, is why the aggregate difference is not presented without the underlying seed values.

Table 5. Three-seed predictive, communication, and training results for all executed scenarios.

Scenario	Accuracy	Macro-F1	F1 95% CI	FAR	Parameters	Traffic (MB)	Time (s)
Centralized CNN-LSTM	84.64 ± 4.73%	84.36 ± 5.01%	71.92–96.80%	0.960 ± 0.295%	143,121	—	34.1 ± 7.0
Centralized proposed	91.35 ± 0.70%	91.33 ± 0.70%	89.59–93.07%	0.540 ± 0.043%	22,689	—	52.1 ± 1.5
FedAvg-IID CNN-LSTM	70.50 ± 0.38%	69.14 ± 0.44%	68.05–70.22%	1.844 ± 0.023%	143,121	286.63	126.5 ± 3.1
FedAvg-IID proposed	83.89 ± 0.78%	83.85 ± 0.89%	81.64–86.05%	1.007 ± 0.049%	22,689	47.57	201.7 ± 8.4
FedAvg-skew proposed	71.33 ± 1.48%	69.41 ± 1.70%	65.18–73.64%	1.792 ± 0.092%	22,689	47.57	194.9 ± 2.3
FedProx-skew proposed	71.57 ± 1.61%	69.67 ± 2.05%	64.58–74.76%	1.777 ± 0.100%	22,689	47.57	201.1 ± 4.1

Values are mean ± sample SD across seeds 42, 52, and 62. The 95% intervals use the t distribution with two degrees of freedom and are therefore wide. Traffic counts bidirectional model-state transfer over 25 rounds and ten participating clients. Source: Authors' calculations from the completed experiment archive.

The compact network contains 22,689 parameters versus 143,121 in the benchmark, a 6.31-fold reduction. On the evaluated CPU, however, it was not the faster model to train. Centralized training took 52.1 ± 1.5 s for the proposed network and 34.1 ± 7.0 s for the benchmark. The result is not contradictory: reducing stored state and multiply-accumulate count does not remove the effects of branch scheduling, memory traffic, kernel implementation, and other runtime overheads.

Table 6. Seed-wise macro-F1 values underlying the aggregate comparisons.

Scenario	Seed 42	Seed 52	Seed 62	Mean ± SD
Centralized CNN-LSTM	86.23%	88.17%	78.69%	84.36 ± 5.01%
Centralized proposed	90.59%	91.98%	91.41%	91.33 ± 0.70%
FedAvg-IID CNN-LSTM	68.75%	69.61%	69.06%	69.14 ± 0.44%
FedAvg-IID proposed	83.82%	84.74%	82.97%	83.85 ± 0.89%
FedAvg-skew proposed	70.45%	70.33%	67.44%	69.41 ± 1.70%
FedProx-skew proposed	71.19%	70.49%	67.34%	69.67 ± 2.05%

Source: Authors' calculations from the saved test-set prediction metrics for all three independent executions.

Figure 3 combines the three centralized confusion matrices and normalizes each row. Its strong diagonal matches the 91.33% mean macro-F1. At class level, the highest mean F1 values include Harmonics (99.08%), Notch (96.96%), Pure_Sinusoidal (97.64%), Swell (95.28%), and Interruption (95.03%). The Pure_Sinusoidal value should not be read as independent evidence of healthy-class generalization because that waveform is duplicated across splits. The more difficult centralized classes were Harmonics_with_Sag (84.82%), Flicker_with_Sag (86.57%), Harmonics_with_Swell (87.70%), and Flicker_with_Swell (87.99%), indicating that some compound morphologies remain difficult even before federated training is introduced.

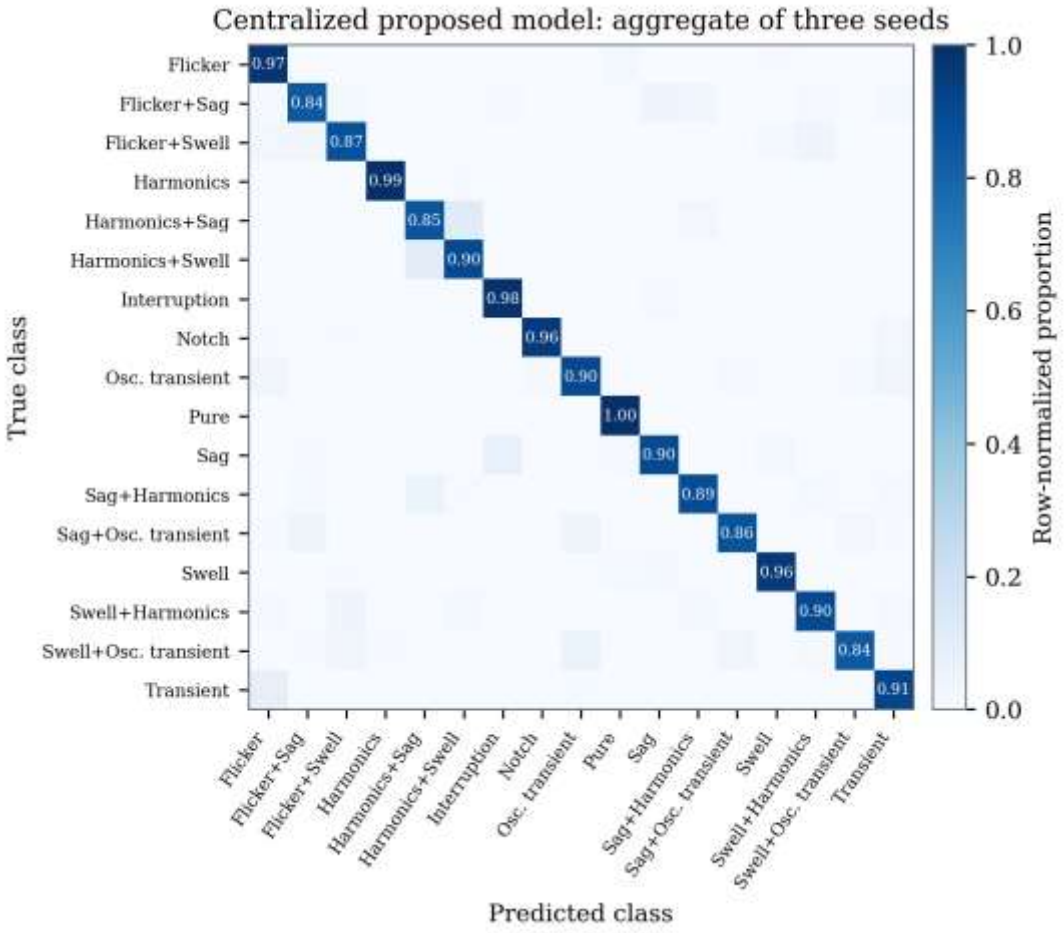

Figure 3. Row-normalized centralized proposed-model confusion matrix aggregated across three seeds.

Source: Authors' aggregation of three saved test-set confusion matrices. Repeated resamples are not treated as independent field events.

Pooling three resampled test confusion matrices makes the recurring error structure easier to see, but the result is not treated as 10,200 independent field observations. Each seed contributes its own 3,400-record test set, and the same underlying reference collection can be sampled differently from one seed to another. For that reason, uncertainty is calculated at seed level rather than from the pooled individual predictions.

6.2 Federated learning under IID distribution

With FedAvg on the IID client split, the proposed global model achieved 83.89% ± 0.78% accuracy and 83.85% ± 0.89% macro-F1. The same architecture was 7.48 ± 0.86 macro-F1 points higher when trained centrally. The paired centralized-to-federated gaps were 6.77, 7.24, and 8.44 points for Seeds 42, 52, and 62, so the direction of the loss was stable. Macro FAR also increased, from 0.540% in centralized training to 1.007% under FedAvg-IID.

The federated CNN-LSTM benchmark reached 70.50% ± 0.38% accuracy and 69.14% ± 0.44% macro-F1. Under the same federated protocol, the proposed model was higher by 14.71 ± 0.69 macro-F1 points. Its seed-wise advantages were 15.08, 15.14, and 13.91 points. Unlike the more variable centralized benchmark comparison, this federated gap was very similar across the three runs. In the evaluated configuration, the compact multi-scale representation therefore preserved considerably more discrimination after repeated local optimization and averaging than the recurrent benchmark.

The communication difference was also large. Across ten clients, the proposed state produced 1.903 MB of bidirectional model traffic per round and 47.57 MB over all 25 rounds. The CNN-LSTM state

required 11.465 MB per round and 286.634 MB in total. Under this shared accounting rule, the proposed model reduces transferred model state by 83.4%. The absolute values omit orchestration messages, transport headers, retransmissions, authentication, and encryption, but those omitted components do not alter the relative checkpoint-size difference measured here.

Figure 4 shows that the IID federated penalty is not distributed evenly across classes. Harmonics kept a mean F1 of 96.08% and Interruption 93.55%, while Transient fell to 72.72%. Swell_with_Harmonics, Flicker_with_Swell, and Sag_with_Harmonics reached 74.78%, 76.16%, and 77.42%, respectively. Macro-F1 remains useful as a summary, but these class-level differences matter in practice. A deployment threshold should therefore be tied to the disturbances that are operationally important, not only to one global average.

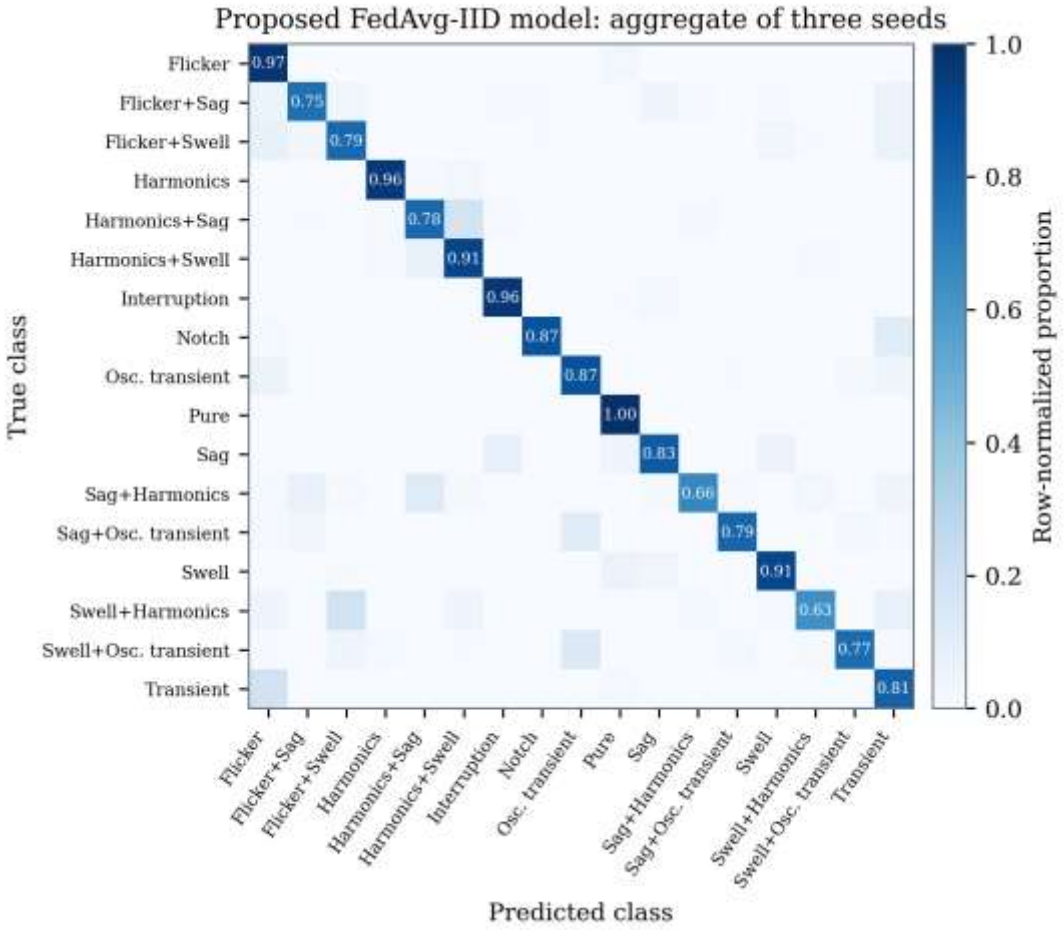

Figure 4. Row-normalized proposed FedAvg-IID confusion matrix aggregated across three seeds.

Source: Authors' aggregation of three saved test-set confusion matrices. Repeated resamples are not treated as independent field events.

Training time tells a different story from communication volume. The proposed federated run took 201.7 ± 8.4 s, compared with 126.5 ± 3.1 s for the benchmark. This is consistent with the later latency profile: fewer parameters and less transmitted state do not guarantee faster execution on the tested backend. In other words, the proposed architecture is communication-efficient here without also being compute-optimal.

6.3 Federated learning under label skew

The largest loss in the training matrix appeared when the client labels were drawn from the Dirichlet partition. FedAvg accuracy moved from 83.89% ± 0.78% under IID assignment to 71.33% ± 1.48%, and macro-F1 moved from 83.85% ± 0.89% to 69.41% ± 1.70%. The paired macro-F1 losses were 13.38,

14.41, and 15.53 points, averaging 14.44 ± 1.08 points. Macro FAR rose at the same time, from 1.007% ± 0.049% to 1.792% ± 0.092%.

The allocation map in Figure 2 provides a direct explanation for this behavior. Global balance does not translate into local balance: some clients are dominated by a subset of labels, while others see those labels rarely. Their local gradients therefore emphasize different objectives, and averaging them does not reconstruct the same update as training on a balanced pooled set. Full participation ensures that every client contributes, but it does not remove that conflict. More local epochs could increase the drift; fewer local epochs would trade part of the problem for more communication at a fixed amount of local work.

On the same within-seed label-skew partitions, FedProx achieved 71.57% ± 1.61% accuracy, 69.67% ± 2.05% macro-F1, and 1.777% ± 0.100% FAR. Its average macro-F1 advantage over FedAvg was only +0.26 ± 0.43 points. At seed level the differences were +0.74, +0.15, and -0.10 points. Accuracy increased by 0.24 ± 0.26 points, while FAR declined by just 0.0147 percentage points on average. These are small shifts, so the result is described as a modest average change rather than a decisive FedProx improvement.

This paired result also shows why the three runs matter. Seed 42 on its own would suggest a +0.74-point benefit for FedProx; Seed 62 slightly favors FedAvg. Taken together, the evidence supports a narrower conclusion: with μ=0.01 and the present training budget, the proximal term can limit drift for some realized partitions, but it does not reliably close the heterogeneity gap.

6.4 Convergence behavior and training budget

Figure 5 places the training histories on a common view, using the seed mean with sample-SD bands. Centralized loss falls quickly over the first few epochs and then becomes flatter. The Seed-42 centralized run stopped after 14 recorded epochs under the patience rule, while the other two reached 15. In every case, the reported test result comes from the best validation checkpoint rather than simply from the final epoch.

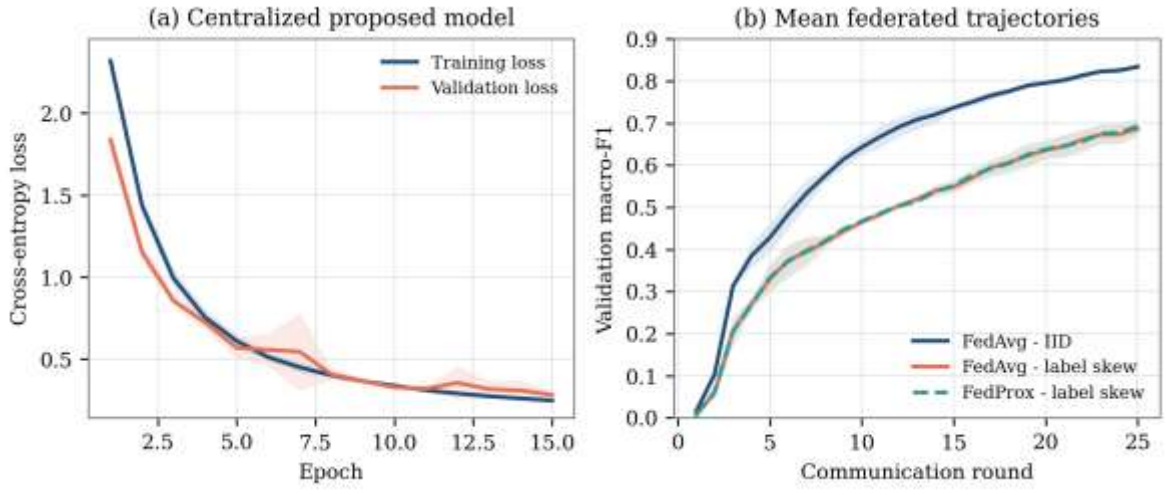

Figure 5. Mean centralized and federated training histories; shaded bands denote sample SD across three seeds.

Source: Authors' calculations from the saved epoch and round histories.

All federated scenarios used the full 25-round budget. The mean best validation macro-F1 was about 0.834 for proposed FedAvg-IID, 0.686 for proposed FedAvg with label skew, and 0.693 for proposed FedProx with label skew. The IID trajectory separates early from the two heterogeneous curves and

remains higher. The two label-skew trajectories stay close to one another, which matches the small difference observed later in their test metrics.

The curves show improvement within the available budget; they do not establish asymptotic convergence. The IID trajectory is still moving slightly at round 25, and neither heterogeneous trajectory has reached a demonstrated plateau. Extra rounds could therefore alter the absolute values or even the FedAvg-FedProx ordering. For that reason, the paper does not claim that FedProx 'converges better.' What the data support is more limited: both heterogeneous methods improved from their early rounds and finished with similar test performance after the same 25-round budget.

Recorded training time was 194.9 ± 2.3 s for proposed FedAvg-skew, 201.1 ± 4.1 s for proposed FedProx-skew, and 201.7 ± 8.4 s for FedAvg-IID. These timings describe the actual CPU execution and the project data path. They are reproducibility measurements, not predictions of how long a substation device or smart meter would take.

6.5 Per-class behavior and leakage sensitivity

Table 7. Class-wise macro-F1 component values for the proposed model.

Class	Centralized	FedAvg-IID	FedAvg-skew	FedProx-skew
Flicker	90.8 ± 0.9%	80.3 ± 0.8%	44.4 ± 32.9%	46.2 ± 32.5%
Flicker_with_Sag	86.6 ± 2.1%	79.2 ± 2.3%	55.5 ± 10.9%	57.9 ± 9.4%
Flicker_with_Swell	88.0 ± 2.2%	76.2 ± 3.5%	42.3 ± 34.4%	44.0 ± 34.2%
Harmonics	99.1 ± 0.8%	96.1 ± 0.5%	91.9 ± 2.7%	92.5 ± 2.2%
Harmonics_with_Sag	84.8 ± 4.3%	78.8 ± 1.4%	48.4 ± 22.9%	49.3 ± 24.0%
Harmonics_with_Swell	87.7 ± 2.7%	83.7 ± 1.3%	72.3 ± 5.2%	73.4 ± 6.4%
Interruption	95.0 ± 1.6%	93.5 ± 1.3%	86.1 ± 1.6%	86.2 ± 2.9%
Notch	97.0 ± 0.9%	91.9 ± 0.5%	89.6 ± 2.6%	88.7 ± 2.9%
Oscillatory_Transient	89.1 ± 0.9%	82.3 ± 0.5%	79.0 ± 4.2%	78.9 ± 3.7%
Pure_Sinusoidal	97.6 ± 0.6%	93.4 ± 1.8%	87.9 ± 1.3%	88.1 ± 1.6%
Sag	90.8 ± 1.5%	84.5 ± 1.6%	76.0 ± 5.4%	74.8 ± 4.8%
Sag_with_Harmonics	90.0 ± 2.1%	77.4 ± 1.9%	62.1 ± 2.0%	63.4 ± 3.6%
Sag_with_Oscillatory_Transient	90.4 ± 1.3%	86.7 ± 1.8%	78.1 ± 3.8%	76.7 ± 4.6%
Swell	95.3 ± 1.7%	89.8 ± 1.9%	86.1 ± 2.7%	85.0 ± 3.3%
Swell_with_Harmonics	90.6 ± 2.0%	74.8 ± 7.0%	59.8 ± 13.2%	58.3 ± 16.8%
Swell_with_Oscillatory_Transient	89.5 ± 1.6%	84.2 ± 3.2%	79.4 ± 4.7%	79.3 ± 4.1%
Transient	90.4 ± 1.9%	72.7 ± 2.6%	41.1 ± 22.9%	41.9 ± 21.4%

Values are mean ± sample SD across three seeds. Pure_Sinusoidal is affected by exact duplicate leakage; see Table 3. Source: Authors' calculations from per-class test metrics.

The per-class results make the cost of heterogeneity clearer than the overall average alone. With label-skew FedAvg, mean F1 fell to 44.39% for Flicker, 42.34% for Flicker_with_Swell, 48.38% for Harmonics_with_Sag, and 41.15% for Transient. FedProx moved the same four means to 46.18%, 43.95%, 49.29%, and 41.85%. Those differences are small beside the change from IID federation, where the corresponding values were 80.25%, 76.16%, 78.83%, and 72.72%. In this experiment, the client distribution therefore changes the class error profile much more than the choice between FedAvg and FedProx.

Other classes were relatively stable. Harmonics remained at 91.91% F1 with FedAvg-skew and 92.46% with FedProx-skew; Notch stayed around 89%, Interruption around 86%, and Swell around 85-86%. One plausible reason is that these disturbances retain more distinctive waveform morphology even when local prevalence changes. Flicker combinations and Transient, by contrast, appear more dependent on local diversity and are more readily absorbed into neighboring patterns.

A second comparison groups the nine isolated or nominal categories—Flicker, Harmonics, Interruption, Notch, Oscillatory_Transient, Pure_Sinusoidal, Sag, Swell, and Transient—against the eight compound categories. The mean class-level F1 for the first group was 93.90% centrally and 87.16% under FedAvg-IID; for compound events, it was 88.44% and 80.12%. Under label skew, isolated/compound means became 75.78%/62.24% with FedAvg and 75.80%/62.78% with FedProx. Thus, a 5.46-point centralized gap widened to roughly 13 points under heterogeneous federation.

The grouped calculation includes Pure_Sinusoidal, so it has to be read together with the integrity audit. When all records whose true label is Pure_Sinusoidal are removed from the metric calculation, macro-F1 over the remaining 16 classes becomes 90.93% ± 0.73% centrally, 83.25% ± 0.85% for FedAvg-IID, 68.26% ± 1.87% for FedAvg-skew, and 68.52% ± 2.25% for FedProx-skew. The method ordering does not change, although every sensitivity value is lower than the 17-class value. Because the conflicting Swell example remains in training, this is a diagnostic sensitivity bound rather than a leakage-free retraining experiment.

6.6 Statistical interpretation

Table 8. Exploratory paired statistical comparisons for macro-F1.

Contrast	Seed deltas (pp)	Mean ± SD / statistic	p	Interpretation
Centralized vs FedAvg-IID (proposed)	+6.77, +7.24, +8.44	+7.48 ± 0.86	0.250	Direction consistent; n is too small for decisive inference
FedAvg-IID vs FedAvg-skew (proposed)	+13.38, +14.41, +15.53	+14.44 ± 1.08	0.250	Large descriptive heterogeneity penalty
FedProx-skew vs FedAvg-skew	+0.74, +0.15, -0.10	+0.26 ± 0.43	0.500	Small and not directionally stable
Friedman: four proposed settings	—	χ^2=8.200	0.042	Exploratory omnibus result with only three blocks

Pairwise p-values are exact two-sided Wilcoxon signed-rank results with n=3 paired seeds. They have very low resolution and must not be used alone to claim superiority. Source: Authors' calculations from the completed three-seed matrix.

For the proposed centralized-versus-FedAvg-IID comparison, the exact two-sided Wilcoxon p-value is 0.250. The FedAvg-IID-versus-FedAvg-skew comparison gives the same value. In both cases, all three paired differences point in one direction, but three pairs are too few to reject at conventional significance levels. The absence of significance is therefore not evidence that the settings are equivalent; it mostly reflects the limited resolution of a nonparametric test with n=3. The descriptive pattern is clearer than the p-value: the mean macro-F1 gaps are 7.48 points and 14.44 points, respectively, and their direction is consistent across the three seeds.

The FedProx-skew-versus-FedAvg-skew result looks different. Its paired p-value is 0.500, the average difference is small, and one seed moves in the opposite direction. That combination does not support a strong claim in favor of either optimizer. The Friedman test across the four proposed settings yields χ^2=8.200 and p=0.042. Because only three seed blocks contribute to that omnibus statistic, we treat it as exploratory evidence that the settings are ordered differently overall, not as a basis for post-hoc superiority statements that are not supported by the pairwise results.

The confidence intervals in Table 5 need the same restraint. For centralized proposed macro-F1, the reported 95% interval is about 89.59-93.07%; for the benchmark it expands to 71.92-96.80%. With only three runs, one seed can move either interval substantially. Reporting these ranges is still useful because it makes the uncertainty visible, but a larger number of seeds would be needed before treating the intervals as stable estimates.

6.7 Noise robustness

Noise produces the sharpest single-checkpoint degradation in the extended analysis. The clean Seed-42 proposed FedAvg-IID model starts at 83.82% accuracy and 83.82% macro-F1. At 40 dB AWGN, those values fall to 57.62% and 54.97%. At 30 dB they are 20.65% and 17.01%, and at 20 dB they reach only 7.88% accuracy and 3.71% macro-F1. At that point, the multi-class decision rule is effectively collapsed rather than operating as a useful detector.

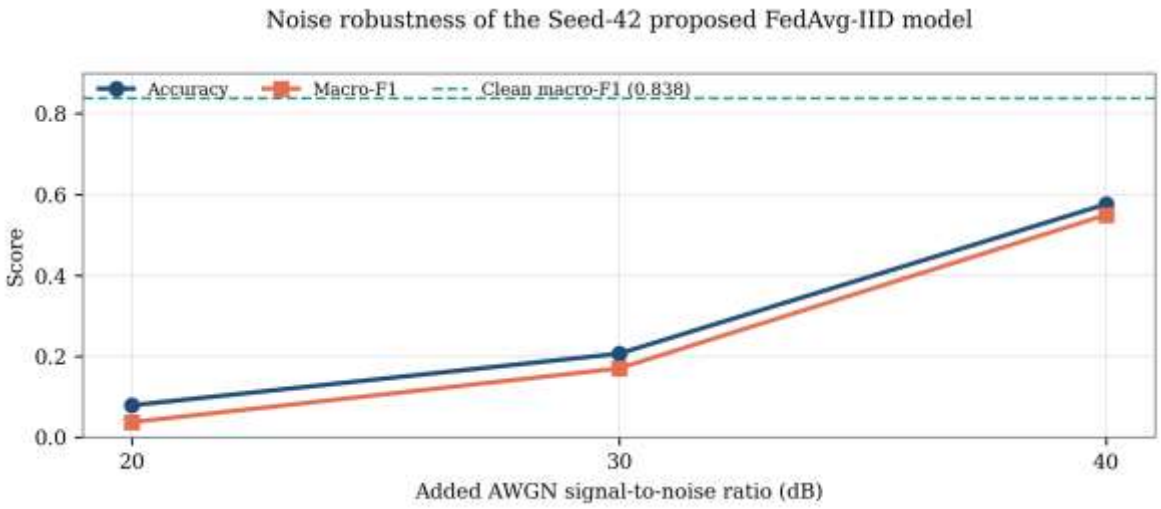

Figure 6. Seed-42 accuracy and macro-F1 under added AWGN; the dashed line denotes clean macro-F1.

Source: Authors' direct evaluation of the saved Seed-42 FedAvg-IID checkpoint.

Table 9. Seed-42 robustness of the proposed FedAvg-IID global model to added AWGN.

Input	SNR (dB)	Accuracy	Macro-F1	FAR	F1 change (pp)
Clean	—	83.82%	83.82%	1.011%	0.00
AWGN	40	57.62%	54.97%	2.649%	-28.86
AWGN	30	20.65%	17.01%	4.960%	-66.82
AWGN	20	7.88%	3.71%	5.757%	-80.12

Percentage-point changes are relative to the clean Seed-42 macro-F1 of 83.82%. Source: Authors' direct evaluation of the saved Seed-42 checkpoint.

Even 40 dB produces a macro-F1 loss of 28.86 percentage points relative to the clean Seed-42 result. FAR rises from 1.011% on clean data to 2.649%, 4.960%, and 5.757% at progressively lower SNR. The robustness confusion matrices show the predictions concentrating into a small number of classes as noise increases, which helps explain why macro-F1 falls faster than accuracy.

These numbers should not be interpreted as proof that the architecture is inherently incapable of noise tolerance. Training followed the clean reference protocol and included no noise augmentation, denoising stage, or sensor-domain adaptation. What the experiment does show is that per-signal normalization alone is insufficient. A field-oriented training distribution would need realistic measurement noise, transformer distortion, missing samples, calibration error, and device-specific spectral effects. Robustness was also tested on only one selected checkpoint, so its variability across seeds is not known.

6.8 Explainability and consistency

For Seed 42, both explanation methods were applied three times per class, producing 51 class-conditioned IG explanations and 51 one-dimensional Grad-CAM explanations. Mean sparsity was 0.588 ± 0.093 for IG and 0.340 ± 0.228 for Grad-CAM. The difference is expected from the methods themselves: IG assigns relevance at input resolution, whereas Grad-CAM combines internal activation channels and then aligns a coarser relevance map with the original 100-sample waveform.

Across ten local models and 17 evaluated samples per model, client-to-global IG cosine similarity averaged 0.924 ± 0.053. Cross-client cosine similarity was lower at 0.887 ± 0.065, and local-to-global Spearman correlation averaged 0.817 ± 0.106. The local explanations therefore tend to point in the same broad direction as the global explanation, but they are not identical. The gap between cosine and Spearman is also informative: coarse relevance profiles are more stable than the exact ranking of individual samples.

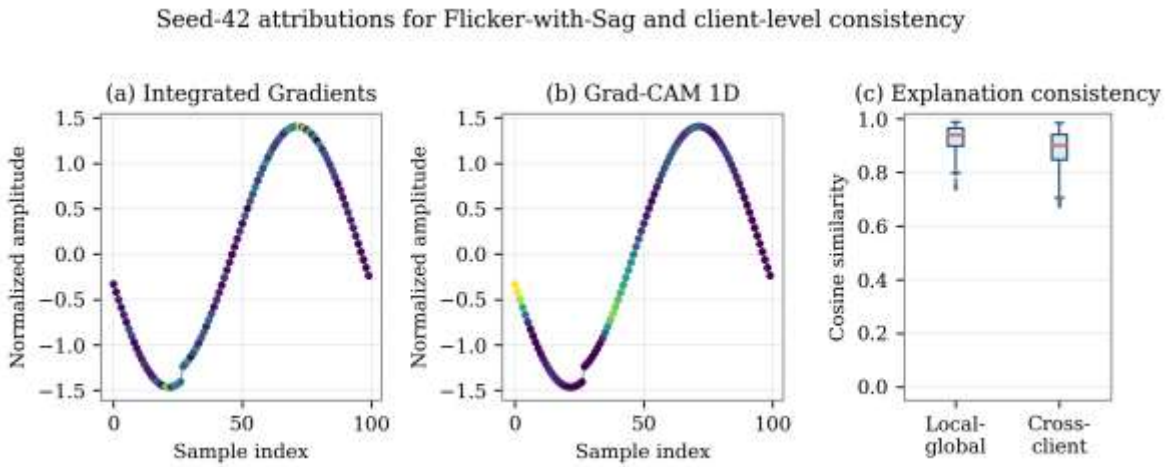

Figure 7. Seed-42 IG and Grad-CAM temporal attributions and client-level explanation consistency.

Source: Authors' XAI outputs; the waveform example is Flicker_with_Sag.

Figure 7 makes that distinction visible. IG and Grad-CAM emphasize similar regions of the Flicker_with_Sag example, but they do not assign the same intensity or the same local ordering. The client boxplots show a comparable spread in agreement. A high similarity score is useful for spotting gross disagreement; it is not evidence that the explanation is physically correct. If every model reacts to the same preprocessing artifact, for example, their explanations could agree while still being non-causal.

The reference data provide class labels but not annotated disturbance intervals. Localization IoU and attribution AUPRC are therefore intentionally omitted. A stronger physical validation would first obtain masks from the disturbance-generation equations or from field annotations and would then combine localization with perturbation-based faithfulness, model-parameter randomization, and expert review. The present XAI result demonstrates consistency between models, not causal validity.

6.9 Edge deployment and quantization

The Seed-42 proposed FP32 checkpoint has 22,689 parameters and a serialized size of 0.112 MB. Batch-one CPU latency averaged 1.107 ± 0.279 ms, corresponding to an estimated throughput of 903.5 samples/s. Dynamic INT8 quantization changed this profile only modestly in size: the file fell by 3.44% to 0.108 MB. Macro-F1 moved from 83.82% to 83.70%, a decrease of about 0.13 percentage points. Mean latency, however, increased by 29.5% to 1.433 ms, and throughput dropped to 697.9 samples/s.

The CNN-LSTM benchmark reacts differently to the same quantization procedure. Its FP32 checkpoint is 0.552 MB, with 0.539 ± 0.062 ms latency, 1856.7 samples/s throughput, and 68.75% macro-F1 for Seed 42. Dynamic quantization reduces its size by 69.73% to 0.167 MB and changes macro-F1 to 68.36%, but mean latency rises to 1.517 ms. The much larger compression is consistent with dynamic quantization acting on the benchmark's recurrent and linear layers. In the proposed model, most weights remain inside convolutional operations that this dynamic procedure does not convert.

Table 10. Seed-42 explanation-consistency and edge-profile summary.

Group	Metric	Value	Evaluation basis	Interpretation
XAI	IG sparsity	0.588 ± 0.093	51 explanations	Input-resolution attribution
XAI	Grad-CAM sparsity	0.340 ± 0.228	51 explanations	Coarse temporal attribution
XAI	Local-global IG cosine	0.924 ± 0.053	10 clients × 17	Strong directional agreement
XAI	Cross-client IG cosine	0.887 ± 0.065	Client pairs	Lower inter-client agreement
XAI	Local-global Spearman	0.817 ± 0.106	10 clients × 17	Rank agreement
Proposed FP32	Size / latency	0.112 MB / 1.107 ms	F1=0.8382	903.5 samples/s
Proposed INT8	Size / latency	0.108 MB / 1.433 ms	F1=0.8370	3.44% smaller; +29.5% latency
Benchmark FP32	Size / latency	0.552 MB / 0.539 ms	F1=0.6875	1856.7 samples/s
Benchmark INT8	Size / latency	0.167 MB / 1.517 ms	F1=0.6836	69.73% smaller; +181.6% latency

Latency is mean batch-one CPU time over 30 post-warm-up repetitions. Dynamic quantization does not convert every convolutional operator. Source: Authors' XAI and CPU profiling outputs for Seed 42.

Two practical conclusions follow from the edge profile. First, 'INT8' describes the requested transformation; it does not prove that every weight and operator is executed with integer arithmetic. Second, a lower parameter count and fewer multiply-accumulate operations do not guarantee lower latency on a given backend. The proposed FP32 checkpoint is small and communication-efficient, yet the benchmark is faster before quantization on this CPU. Any deployment decision should therefore be preceded by target-device profiling and, where appropriate, tests of operator fusion, static quantization, quantization-aware training, and energy use.

The process-level memory profiler did not resolve a change in resident memory during the individual inference windows. That observation reflects measurement resolution, not zero memory consumption. Parameter count and serialized size are still available, but peak activation memory should be measured later with instrumentation designed for the actual target device.

6.10 Overall trade-off and evidence synthesis

Figure 8 brings the four proposed-model training settings together. Centralized training remains the strongest predictive reference. FedAvg-IID provides the best distributed compromise observed here, with 83.85% ± 0.89% macro-F1 and 47.57 MB of measured model traffic. The dominant penalty appears when the client distribution becomes label-skewed. Under the chosen μ and 25-round budget, FedProx changes that heterogeneous result only slightly.

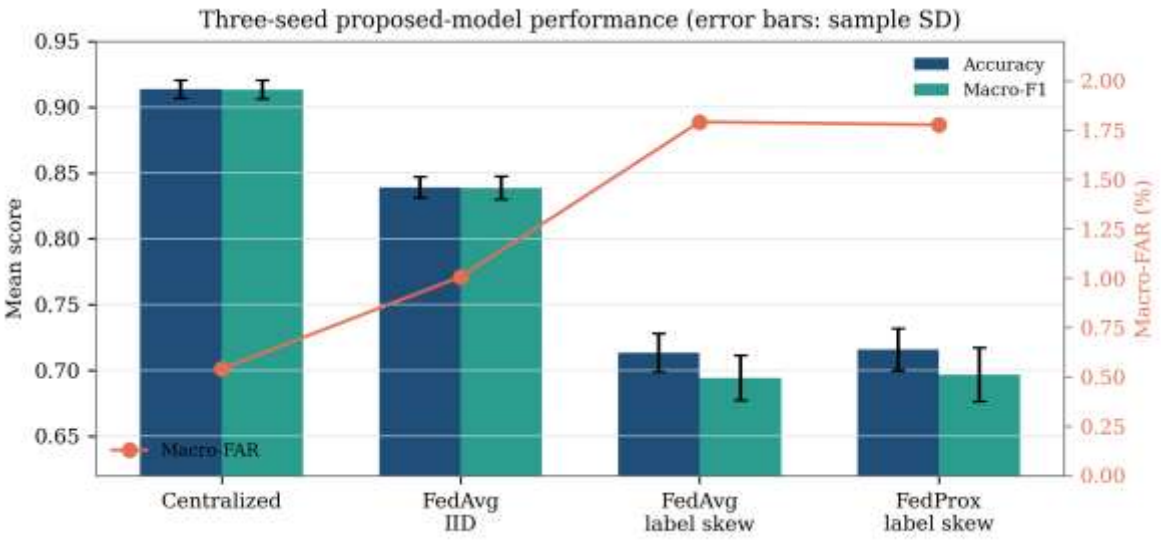

Figure 8. Three-seed proposed-model performance across centralized, IID, and label-skew settings.

Source: Authors' calculations from the completed three-seed matrix.

There is no single configuration that wins on every metric. If data pooling is allowed, the centralized proposed model gives the strongest predictive result. Among the federated alternatives, proposed FedAvg-IID provides the best measured balance between predictive quality and communication. On the tested CPU, the benchmark is faster in FP32. FedProx cannot be preferred generally to FedAvg on the

basis of these three heterogeneous runs because its mean advantage is small and the ordering reverses in one seed.

The different parts of the evaluation tell a consistent story. Fewer parameters explain the reduction in model traffic. The confusion matrices and class-wise results show where macro-F1 is lost. The label-skew allocation and paired drop identify heterogeneity as a first-order challenge. XAI measurements show strong but incomplete agreement between client and global explanations. Noise and quantization experiments keep clean-data accuracy from being mistaken for deployment readiness, while the duplicate audit shows that data provenance can limit the meaning of an otherwise strong metric.

Within the boundaries of these experiments, XFed-PQD is best interpreted as a reproducible integrated research framework rather than as a finished alarm or protection product. Its contribution is the combination of model design, federated evaluation, explanation, robustness, data-integrity analysis, and edge profiling in one auditable workflow, with the limits of each result stated explicitly.

7. Discussion

7.1 Answers to the research questions

For RQ1, the central comparison is the drop from pooled to IID federated training. The proposed model records 91.33% ± 0.70% macro-F1 centrally and 83.85% ± 0.89% with FedAvg-IID, giving a paired gap of 7.48 ± 0.86 points. Even with that loss, the federated proposed model remains 14.71 ± 0.69 points above the federated CNN-LSTM and requires 83.4% less measured model traffic. Under the protocol used here, the proposed architecture therefore offers a better predictive-quality-versus-communication balance. The remaining centralized-federated gap could still change with a longer or differently tuned training budget.

RQ2 is dominated by the effect of the client distribution. Changing FedAvg from IID to label-skew partitions costs 14.44 ± 1.08 macro-F1 points on average. Replacing FedAvg with FedProx on those same heterogeneous partitions changes the mean from 69.41% to 69.67%, or +0.26 ± 0.43 points. Seed 62 moves slightly in the other direction and favors FedAvg. In other words, heterogeneity is the main result here; with the selected μ, the proximal term does not remove that problem and does not behave as a consistently better alternative.

RQ3 asks whether the global explanation resembles the explanations produced by local client models. In the Seed-42 analysis, the mean local-global cosine similarity is 0.924, which points to strong agreement in overall attribution direction. The corresponding Spearman correlation is lower, at 0.817, indicating that the detailed ordering of individual samples varies more than the broad relevance profile. The global explanation is therefore broadly representative of client evidence in this selected run, although local explanations are not interchangeable. Since the dataset has no event masks, this finding concerns agreement between models and should not be interpreted as proof that the highlighted samples are physically correct.

RQ4 gives a much less favorable result. The Seed-42 checkpoint was trained under the clean reference protocol, and its macro-F1 falls to 54.97% at 40 dB AWGN before declining further as SNR is reduced. Under these conditions, the model cannot be described as noise robust. Any operational claim would need retraining with noise-aware augmentation and another evaluation using sensor conditions that are closer to the intended deployment environment.

RQ5 shows that a smaller federated state is not automatically a faster one. The proposed FP32 checkpoint is compact and cuts communication substantially, yet its measured latency on the tested CPU is higher than that of the benchmark. Dynamic INT8 quantization changes F1 only slightly, but it produces little size reduction for the CNN-dominant proposed model and increases latency because much of the convolutional path remains in floating point. The practical choice of deployment format therefore has to follow measurements on the target backend rather than the datatype label by itself.

7.2 Practical implications

An edge-enabled FL topology reduces the need to move raw waveforms routinely between sites, which is useful when organizations cannot pool measurements or when links are constrained. Under the implemented accounting rule, the proposed state requires about 1.903 MB of bidirectional transfer per round across ten clients, compared with 11.465 MB for the benchmark. Over 25 rounds, the difference is greater than 239 MB. These are model-state figures; a production deployment would add encryption, authentication, orchestration, and retry traffic to both.

Explanations can also support operator triage. Pairing a predicted class with a concentrated temporal attribution gives the operator a specific portion of the waveform to inspect, while consistency metrics indicate whether local and global models are relying on broadly similar evidence. Such outputs are aids, not replacements for protection logic or electrical rules. Their reliability still has to be checked against physical event masks, counterfactual perturbations, and domain-expert judgment.

The integrity audit has an equally practical implication. A classifier can appear exceptionally reliable on the healthy class when the identical healthy waveform is present in every split. If such a result were used to set alarm expectations, false-alarm performance could be interpreted too optimistically. Unique-event identifiers, group-aware splitting, source tracking, and review of label conflicts are therefore not secondary data-management details; they are part of the model-validation problem.

7.3 Relationship to previous studies

The numerical results in this study should not be ranked directly against accuracies reported from other PQD collections. Published work differs in class count, signal-generation equations, sampling rates, transforms, window lengths, and noise conditions [26], [28]-[44]. Methodological comparison is more defensible. Alsabaan et al. demonstrated FL-based PQD classification together with poisoning analysis [27]; XFed-PQD adds repeated label-skew evaluation, client-global explanation consistency, edge profiling, and an explicit leakage audit. Interpretable PQD studies, including Altun et al., show the value of explaining waveform decisions [29], [32], [33]; here, that explanation is evaluated inside a federated client-global setting. Lightweight and embedded studies show that inference can be moved near the meter [30], [44]. The present edge measurements add a complementary caution: neither parameter reduction nor dynamic INT8 automatically produces lower latency.

The wider smart-grid FL literature emphasizes non-IID data, personalization, privacy, security, and client variability [4]-[25]. The label-skew results reported here fit that picture: a dataset can be balanced globally while the client objectives remain highly unbalanced. XFed-PQD does not implement secure aggregation, differential privacy, adversarial-client defenses, or personalized heads. Those additions would alter both predictive performance and resource cost, so they should be measured jointly rather than appended as untested claims.

7.4 Reliability envelope and deployment gates

The experiments define a limited reliability envelope rather than a simple deployment verdict. What has actually been tested is a system built around clean reference waveforms, a fixed 100-sample window, known labels, the recorded class map, full client participation, and a workstation CPU profile. Conditions such as field sensor noise, variable waveform lengths, missing values, changes in feeder topology, previously unseen compound disturbances, malicious clients, and hardware-specific runtime constraints lie outside that envelope. Claims about the system should stay inside the measured conditions until a dedicated experiment extends them.

The results point to four practical checks before deployment. First, the data need to pass an integrity check in which exact duplicates and cross-label conflicts are removed or reconciled before a group-aware resplit. Second, robustness has to be judged by acceptable per-class recall under realistic sensor and network conditions, not only by a clean-data average. Third, the federated setting needs more repeated trials across client distributions, participation rates, and seeds, together with predefined minimum performance for vulnerable classes. The fourth check is device specific: latency, memory, energy use, and thermal behavior should be measured using the exact model format and runtime intended for operation.

Explanation quality becomes another acceptance check if the attributions are expected to support operator trust. A useful test would combine event-window localization, perturbation-based faithfulness, randomization checks, and stability across seeds. The measured local-global cosine similarity is encouraging, but agreement alone is not enough; several models can agree and still focus on the same non-causal artifact. Physical masks derived from disturbance-generation equations or synchronized event annotations would provide a much stronger basis for that validation.

Framing the limitations as acceptance checks makes the next engineering step easier to define. The 40 dB result, for example, does not simply suggest collecting 'more data'; it identifies a need for noise-aware retraining followed by another test at or below that condition. The very small size reduction obtained from dynamic quantization leads to a different next step: static quantization or operator fusion

should be tested before committing to an INT8-specific runtime. In each case, a failed check should trigger a specific design change and then a repeat evaluation under the same criterion.

7.5 Monitoring lifecycle and model governance

A fielded PQD classifier would need to be managed as a measurement component whose operating conditions can change over time. Before a site is activated, sensor scaling, waveform-window alignment, the supported class vocabulary, and the expected noise range should be checked. A small local qualification set can then show whether that site's data are materially different from the reference collection. If the difference is large, adding the client immediately to synchronous aggregation risks contaminating the global update; the better response is to inspect data quality, adjust preprocessing, or use a personalized calibration layer first.

Once the system is running, the useful audit trail is broader than the predicted class alone. Prediction confidence, alarm frequency, client participation, validation surrogates, and model-version identifiers can be logged without routinely retaining raw waveforms. A change in class prevalence or feature statistics may indicate distribution shift, but it does not prove that a new electrical phenomenon has appeared. Selected records therefore need a controlled review in which domain experts confirm the label and decide whether the event belongs to an existing class, a known compound class, or an out-of-scope condition. Only reviewed records should enter a later federated update, and they should still pass duplication and data-quality checks first.

Promotion of a new global checkpoint should be staged rather than automatic. The candidate first needs to outperform the active model on a fixed regression suite that includes the difficult compound categories and the leakage-free sensitivity subset. It then needs to pass the noise, explanation, and device-profile checks on the target runtime. A small gain in an aggregate score is not enough if recall drops for an operationally important class or if the explanations become unstable. For later audit, the approved checkpoint should be stored together with its class map, preprocessing version, quantization mode, and acceptance-test results.

Rollback has to be planned at the same time. If alarm behavior or latency deteriorates after a model promotion, the previous approved checkpoint should remain available and the system should record which clients received each version. Federated coordination makes this more complicated because a partial update can leave clients training from different global states. Version checks at the beginning and end of each round can stop incompatible updates from entering the same aggregation. None of these controls increases macro-F1 directly, but they determine whether the measured model can be operated safely and whether a later audit can reconstruct what happened.

8. Limitations and Future Work

The clearest limitation comes from the data themselves. The reference collection is not a multi-site field dataset, and the Pure_Sinusoidal class is made up of repeated copies of one waveform. In addition, one exact waveform appears under both Pure_Sinusoidal and Swell, which gives the same input two labels. Reporting the primary results alongside the sensitivity metrics makes this problem explicit, but no analysis can recover signal diversity that is absent from the source files. A stronger dataset would include event identifiers, source-aware grouping, and explicit masks marking disturbance intervals.

The experiment is also narrow in repetition and federated conditions. Three completed seeds are enough to show means, dispersion, and the direction of paired changes, but they are not enough for fine distinctions between optimizers. The current matrix does not cover additional Dirichlet concentrations, partial participation, client dropout, personalized heads, or alternative numbers of local epochs. A more definitive aggregation study would use more seeds, state the primary contrasts in advance, and run every method on the same realized partition draws. For this reason, the inferential results in the present paper are intentionally cautious and mostly descriptive.

A related limitation is the fixed training budget. Twenty-five federated rounds show the trajectory reached within that budget; they do not establish asymptotic convergence. More rounds, server learning-rate schedules, client sampling, balanced aggregation, or personalized FL may change the heterogeneous results. Robustness also needs to move beyond the current AWGN test. Training-time noise augmentation, colored noise, measurement-transformer harmonics, missing samples, calibration error, concept drift, and transfer across devices would better reflect the range of conditions expected outside the reference dataset.

Formal privacy and security guarantees were not implemented. Model parameters are shared, the coordinator is trusted, and parameter exchange can still leak information. Differential privacy, secure aggregation, encrypted transport, poisoning defenses, and malicious-client detection are therefore important extensions. Their cost, however, should be measured directly: each can affect both communication and predictive performance, and that overhead should not be assumed to be negligible.

The deployment measurements have a similar boundary. They come from a workstation CPU using dynamic quantization and cannot stand in for every embedded processor or accelerator. Static quantization, quantization-aware training, operator fusion, ONNX or vendor-specific runtimes, and structured pruning may lead to different size and latency behavior. Future device tests should therefore include energy per inference, peak memory, concurrent-stream behavior, thermal throttling, and on-device latency. The explainability study also needs a broader validation layer, including localization masks, perturbation faithfulness, randomization tests, expert assessment, and stability checks across seeds, partition types, and noise levels.

9. Conclusion

XFed-PQD combines compact temporal classification, federated optimization, explanation, robustness analysis, and edge profiling in a single 17-class power-quality monitoring framework. Across three seeds, the proposed architecture records 91.33% ± 0.70% macro-F1 with centralized training and 83.85% ± 0.89% with FedAvg-IID. Its measured model traffic over 25 federated rounds is 83.4% lower than that of the CNN-LSTM benchmark. The main performance loss appears when the client data become heterogeneous: Dirichlet label skew reduces macro-F1 to 69.41% ± 1.70%. FedProx raises the mean to 69.67% ± 2.05%, but the +0.26-point average difference is small and does not keep the same direction across all three seeds. The extended Seed-42 analysis also shows strong, although not perfect, agreement between local and global IG explanations.

Those results are encouraging, but they do not remove the limits revealed by the rest of the experiment. Added noise causes severe degradation, and dynamic INT8 quantization preserves nearly all of the predictive quality without making the proposed CNN-dominant model faster on the evaluated CPU. More importantly, the waveform audit identifies duplicated healthy data and one exact cross-label collision, which changes how the healthy-class result has to be interpreted and motivates the accompanying sensitivity analysis. The evidence therefore supports explainable federated PQD monitoring as a feasible direction under the conditions that were tested, not as a deployment guarantee. Extending that evidence will require more diverse field data, broader repeated heterogeneous trials, formal privacy and security mechanisms, physically grounded explanation tests, and optimization on the hardware that will actually run the model.

References

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. 20th Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2017, pp. 1273–1282.

[2] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Machine Learning and Systems (MLSys)*, vol. 2, 2020, pp. 429–450.

[3] M. N. Fekri, K. Grolinger, and S. Mir, "Distributed load forecasting using smart meter data: Federated learning with recurrent neural networks," *Int. J. Electr. Power Energy Syst.*, vol. 137, Art. no. 107669, 2022, doi: 10.1016/j.ijepes.2021.107669.

[4] X. Cheng, C. Li, and X. Liu, "A review of federated learning in energy systems," in *Proc. IEEE/IAS Industrial and Commercial Power System Asia (I&CPS Asia)*, 2022, pp. 2089–2095, doi: 10.1109/ICPSAsia55496.2022.9949863.

[5] M. N. Fekri, K. Grolinger, and S. Mir, "Asynchronous adaptive federated learning for distributed load forecasting with smart meter data," *Int. J. Electr. Power Energy Syst.*, vol. 153, Art. no. 109285, 2023, doi: 10.1016/j.ijepes.2023.109285.

[6] N. Gholizadeh and P. Musilek, "Distributed learning applications in power systems: A review of methods, gaps, and challenges," *Energies*, vol. 14, no. 12, Art. no. 3654, 2021, doi: 10.3390/en14123654.

[7] H. Bousbiat, R. Bousselidj, Y. Himeur, A. Amira, F. Bensaali, F. Fadli, W. Mansoor, and W. Elmenreich, "Crossing roads of federated learning and smart grids: Overview, challenges, and perspectives," arXiv:2304.08602, 2023, doi: 10.48550/arXiv.2304.08602.

[8] R. Zheng, A. Sumper, M. Aragüés-Peñalba, and S. Galceran-Arellano, "Advancing power system services with privacy-preserving federated learning techniques: A review," *IEEE Access*, vol. 12, pp. 76753–76780, 2024, doi: 10.1109/ACCESS.2024.3407121.

[9] Z. Zhang, S. Rath, J. Xu, and T. Xiao, "Federated learning for smart grid: A survey on applications and potential vulnerabilities," *ACM Comput. Surv.*, 2026, doi: 10.1145/3760788.

[10] R. Zhao, J. Lu, Z. Liu, T. Wang, W. Guo, T. Lan, and C. Hu, "A federated-learning algorithm based on client sampling and gradient projection for the smart grid," *Electronics*, vol. 13, no. 11, Art. no. 2023, 2024, doi: 10.3390/electronics13112023.

[11] H. Lee, "Towards convergence in federated learning via non-IID analysis in a distributed solar energy grid," *Electronics*, vol. 12, no. 7, Art. no. 1580, 2023, doi: 10.3390/electronics12071580.

[12] R. Rahman, N. Kumar, and D. C. Nguyen, "Electrical load forecasting in smart grid: A personalized federated learning approach," arXiv:2411.10619, 2024, doi: 10.48550/arXiv.2411.10619.

[13] S. Ghosh and G. Mittal, "Federated learning for critical electrical infrastructure—handling data heterogeneity for predictive maintenance of substation equipment," *Front. Artif. Intell.*, vol. 8, Art. no. 1697175, 2026, doi: 10.3389/frai.2025.1697175.

[14] A. Tibermacine, I. Naidji, I. E. Tibermacine, S. Sahraoui, M. Zouai, A. Rabehi, and M. Habib, "Privacy-preserving load forecasting in smart grids using federated learning: A comparative analysis of aggregation strategies," *Front. Energy Res.*, vol. 14, Art. no. 1854654, 2026, doi: 10.3389/fenrg.2026.1854654.

[15] M. A. Husnoo, A. Anwar, N. Hosseinzadeh, S. N. Islam, A. N. Mahmood, and R. Doss, "A secure federated learning framework for residential short-term load forecasting," *IEEE Trans. Smart Grid*, vol. 15, no. 2, pp. 2044–2055, 2024, doi: 10.1109/TSG.2023.3292382.

[16] M. A. Husnoo, A. Anwar, H. T. Reda, N. Hosseinzadeh, S. N. Islam, A. N. Mahmood, and R. Doss, "FedDiSC: A computation-efficient federated learning framework for power systems disturbance and cyber attack discrimination," *Energy and AI*, vol. 14, Art. no. 100271, 2023, doi: 10.1016/j.egyai.2023.100271.

[17] J. Jithish, B. Alangot, N. Mahalingam, and K. S. Yeo, "Distributed anomaly detection in smart grids: A federated learning-based approach," *IEEE Access*, vol. 11, pp. 7157–7179, 2023, doi: 10.1109/ACCESS.2023.3237554.

[18] M. Kesici, B. Pal, and G. Yang, "Detection of false data injection attacks in distribution networks: A vertical federated learning approach," *IEEE Trans. Smart Grid*, vol. 15, no. 6, pp. 5952–5964, 2024, doi: 10.1109/TSG.2024.3399396.

[19] Y. Li, D. Qin, H. V. Poor, and Y. Wang, "Introducing edge intelligence to smart meters via federated split learning," *Nat. Commun.*, vol. 15, Art. no. 9044, 2024, doi: 10.1038/s41467-024-53352-9.

[20] N. Hudson, M. J. Hossain, M. Hosseinzadeh, H. Khamfroush, M. Rahnamay-Naeini, and N. Ghani, "A framework for edge intelligent smart distribution grids via federated learning," in *Proc. 30th Int. Conf. Computer Communications and Networks (ICCCN)*, 2021, doi: 10.1109/ICCCN52240.2021.9522360.

[21] D. N. Molokomme, A. J. Onumanyi, and A. M. Abu-Mahfouz, "Edge intelligence in smart grids: A survey on architectures, offloading models, cyber security measures, and challenges," *J. Sens. Actuator Netw.*, vol. 11, no. 3, Art. no. 47, 2022, doi: 10.3390/jsan11030047.

[22] D. Liu, H. Liang, X. Zeng, Q. Zhang, Z. Zhang, and M. Li, "Edge computing application, architecture, and challenges in ubiquitous power Internet of Things," *Front. Energy Res.*, vol. 10, Art. no. 850252, 2022, doi: 10.3389/fenrg.2022.850252.

[23] W. Li, N. Zhang, Z. Liu, S. Ma, H. Ke, J. Wang, and T. Chen, "A trusted decision fusion approach for the power Internet of Things with federated learning," *Front. Energy Res.*, vol. 11, Art. no. 1061779, 2023, doi: 10.3389/fenrg.2023.1061779.

[24] H. Gupta, P. Agarwal, K. Gupta, S. Baliarsingh, O. P. Vyas, and A. Puliafito, "FedGrid: A secure framework with federated learning for energy optimization in the smart grid," *Energies*, vol. 16, no. 24, Art. no. 8097, 2023, doi: 10.3390/en16248097.

[25] B. Saylam and Ö. D. İncel, "Federated learning on edge sensing devices: A review," arXiv:2311.01201, 2023, doi: 10.48550/arXiv.2311.01201.

[26] R. Machlev, A. Chachkes, J. Belikov, Y. Beck, and Y. Levron, "Open source dataset generator for power quality disturbances with deep-learning reference classifiers," *Electr. Power Syst. Res.*, vol. 195, Art. no. 107152, 2021, doi: 10.1016/j.epsr.2021.107152.

[27] M. Alsabaan, A. Elsayed, A. Bondok, M. M. Badr, M. Mahmoud, T. Alshawi, and M. I. Ibrahem, "Robust federated-learning-based classifier for smart grid power quality disturbances," *Sensors*, vol. 25, no. 22, Art. no. 6880, 2025, doi: 10.3390/s25226880.

[28] A. M. Saber, A. Selim, M. M. Hammad, A. Youssef, D. Kundur, and E. El-Saadany, "A novel approach to classify power quality signals using vision transformers," in *Proc. 50th Annual Conf. IEEE Industrial Electronics Society (IECON)*, 2024, doi: 10.1109/IECON55916.2024.10905293.

[29] S. Yang, T. Shan, and X. Yan, "Interpretable DWT-1DCNN-LSTM network for power quality disturbance classification," *Energies*, vol. 18, no. 2, Art. no. 231, 2025, doi: 10.3390/en18020231.
[30] H. Bai, R. Yao, T. Liu, Z. Ma, S. Liu, Y. Lei, and Y. Zheng, "A lightweight model for power quality disturbance recognition targeting edge deployment," *Energies*, vol. 19, no. 2, Art. no. 368, 2026, doi: 10.3390/en19020368.
[31] H. Bai, R. Yao, W. Zhang, Z. Zhong, and H. Zou, "Power quality disturbance classification strategy based on fast S-transform and an improved CNN-LSTM hybrid model," *Processes*, vol. 13, no. 3, Art. no. 743, 2025, doi: 10.3390/pr13030743.
[32] B. E. Altun, F. Alpsalaz, H. Uzel, and Y. Türkay, "Explainable DL based classification for power quality disturbances in renewable-energy-integrated distribution networks," *IET Renew. Power Gener.*, vol. 20, no. 1, Art. no. e70269, 2026, doi: 10.1049/rpg2.70269.
[33] Y. Chen, S. S. Yu, and K. M. Muttaqi, "A Bayesian framework for uncertainty-aware explanations in power quality disturbance classification," arXiv:2604.13658, 2026, doi: 10.48550/arXiv.2604.13658.
[34] F. A. Albalooshi and M. R. Qader, "Deep learning algorithm for automatic classification of power quality disturbances," *Appl. Sci.*, vol. 15, no. 3, Art. no. 1442, 2025, doi: 10.3390/app15031442.
[35] C. A. Iturrino Garcia, M. Bindi, F. Corti, A. Luchetta, F. Grasso, L. Paolucci, M. C. Piccirilli, and I. Aizenberg, "Power quality analysis based on machine learning methods for low-voltage electrical distribution lines," *Energies*, vol. 16, no. 9, Art. no. 3627, 2023, doi: 10.3390/en16093627.
[36] J. Cai, K. Zhang, and H. Jiang, "Power quality disturbance classification based on parallel fusion of CNN and GRU," *Energies*, vol. 16, no. 10, Art. no. 4029, 2023, doi: 10.3390/en16104029.
[37] S. Samanta et al., "A comprehensive review of deep-learning applications to power quality analysis," *Energies*, vol. 16, no. 11, Art. no. 4406, 2023, doi: 10.3390/en16114406.
[38] M. A. A. Baig, N. I. Ratyal, A. Amin, U. Jamil, S. Liaquat, H. M. Khalid, and M. F. Zia, "An ensemble deep CNN approach for power quality disturbance classification: A technological route towards smart cities using image-based transfer," *Future Internet*, vol. 16, no. 12, Art. no. 436, 2024, doi: 10.3390/fi16120436.
[39] M. N. Islam, "A multimodal deep learning model with differential evolution-based optimized features for classification of power quality disturbances," *J. Electr. Syst. Inf. Technol.*, vol. 12, Art. no. 11, 2025, doi: 10.1186/s43067-025-00194-0.
[40] W.-M. Lin and C.-H. Wu, "Fast support vector machine for power quality disturbance classification," *Appl. Sci.*, vol. 12, no. 22, Art. no. 11649, 2022, doi: 10.3390/app122211649.
[41] S. Chamchuen, A. Siritaratiwat, P. Fuangfoo, P. Suthisopapan, and P. Khunkitti, "High-accuracy power quality disturbance classification using the adaptive ABC-PSO as optimal feature selection algorithm," *Energies*, vol. 14, no. 5, Art. no. 1238, 2021, doi: 10.3390/en14051238.
[42] L. Gao, J. Wang, M. Zhang, S. Zhang, H. Wang, and Y. Wang, "Classification strategy for power quality disturbances based on variational mode decomposition algorithm and improved support vector machine," *Processes*, vol. 12, no. 6, Art. no. 1084, 2024, doi: 10.3390/pr12061084.
[43] Q. Xu, F. Zhu, W. Jiang, X. Pan, P. Li, X. Zhou, and Y. Wang, "Efficient identification method for power quality disturbance: A hybrid data-driven strategy," *Processes*, vol. 12, no. 7, Art. no. 1395, 2024, doi: 10.3390/pr12071395.
[44] B. F. Khaldi, F. Z. Dekhandji, and A. Recioui, "Low-cost IoT-based smart meter for real-time power quality monitoring and disturbance detection using embedded 1D CNN," *J. Energy Syst.*, vol. 9, no. 4, pp. 365–379, 2025, doi: 10.30521/jes.1718242.
[45] M. Ingram, R. Mahmud, and D. Narang, "Background information on the power quality requirements in IEEE Std 1547-2018," National Renewable Energy Laboratory, Golden, CO, USA, Tech. Rep. NREL/TP-5D00-78751, 2021.
[46] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. 34th Int. Conf. Machine Learning (ICML)*, 2017, pp. 3319–3328.
[47] R. R. Selvaraju et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, 2017, pp. 618–626, doi: 10.1109/ICCV.2017.74.
[48] A. Hedström et al., "Quantus: An explainable AI toolkit for responsible evaluation of neural network explanations and beyond," *J. Mach. Learn. Res.*, vol. 24, no. 34, pp. 1–11, 2023.
[49] Alriheebi, R. A. (2026). Integrated Modeling and Mult objective Control of Multilevel DC/AC Inverters for Conducted Electromagnetic Interference Mitigation. *Al-Farooq Journal of Sciences*, *2*(3), 897-925.
[50] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. van Baalen, and T. Blankevoort, "A white paper on neural network quantization," arXiv:2106.08295, 2021, doi: 10.48550/arXiv.2106.08295.
[51] R. Alsaigh, R. Mehmood, and I. Katib, "AI explainability and governance in smart energy systems: A review," *Front. Energy Res.*, vol. 11, Art. no. 1071291, 2023, doi: 10.3389/fenrg.2023.1071291.