WWW.IJRPR.COM

**International Journal of Research Publication and Reviews**

Journal Homepage: www.ijrpr.com ISSN 2582-7421

# Federated Learning-Based Intelligent Self-Healing Mechanism in Smart Power Grids

***Prof. J. M. Nimgade, Mohit Nitnaware, Dhanshri Undirwade, Vidhi Kshirsagar and Amit Hedau***

Department of Electronics and Communication Engineering, Priyadarshini College of Engineering, Nagpur, Maharashtra, India
Email: jyotinimgade1981@gmail.com

**ABSTRACT—**

Modern power distribution networks are becoming more complex as electricity demand grows and renewable sources are added to the grid. Conventional monitoring and protection approaches may depend on centralized supervision or manual intervention, which can delay fault identification and power restoration. This paper presents a proposed federated learningbased self-healing framework for smart power grids. The framework combines voltage and current monitoring, machinelearning-based fault classification, relay-based fault isolation, and power restoration when a suitable backup source is available. In federated learning, monitoring nodes train local models and share model updates with an aggregation server rather than sending raw electrical measurements. This can support collaborative model improvement while limiting the transfer of sensitive operational data. When the model identifies a possible fault, the controller can activate a relay to isolate the affected section. After the fault is cleared and operating conditions are verified, the load may be returned to the primary supply. The framework can be assessed using fault-detection accuracy, false-alarm rate, detection latency, switching time, and restoration time. Physical integration and experimental evaluation are required to establish the system's performance under practical operating conditions.

*Index Terms*—Smart grid, federated learning, self-healing grid, fault detection, machine learning, automatic switching, distributed learning, renewable energy.

## I. INTRODUCTION

The growing demand for electricity and the increasing use of renewable energy sources have raised the need for reliable and responsive power distribution systems. Traditional grids commonly use established monitoring and protection methods. However, depending on the network design, fault detection and restoration may take time, particularly when manual intervention or centralized analysis is required.

Smart-grid technology combines sensors, communication systems, automation, and control to improve the monitoring and operation of electrical networks. A self-healing grid aims to detect abnormal conditions, isolate the affected section, and restore service to unaffected loads with limited human intervention. Voltage and current sensors, a microcontroller, and relay-based switching can form part of such a system. Where a compatible backup source is available, source switching may also help maintain supply to eligible loads.

Machine learning can be used to distinguish normal electrical conditions from measurements associated with potential faults. However, collecting all measurements at a central location may create data-management and privacy concerns. Federated learning offers another approach: participating nodes train local models and share model updates for aggregation, without routinely sending their raw measurements to the central server. The quality of the resulting model depends on factors such as local training data, communication reliability, and the computing resources available at each node.

This paper describes a proposed framework that combines these functions. It also outlines a web-based monitoring dashboard referred to in the project material as *POWER MONITOR – AI Telemetry & Self-Healing*. The framework is intended to support fault monitoring and response; its complete performance must be confirmed through hardware integration and systematic testing.

## II. AIM AND OBJECTIVES

### *A. Aim*

The aim is to develop a federated learning-based intelligent self-healing mechanism for smart grids that supports fault detection, automatic fault isolation, and safe power restoration while enabling distributed model training.

### *B. Objectives*

The objectives of the proposed system are:

1. To monitor electrical parameters, including voltage and current, in real time.
2. To identify and classify possible electrical faults using machine-learning techniques.
3. To support collaborative model training through federated learning without routinely sharing raw electrical measurements.
4. To isolate a faulty section using relay-based switching.
5. To support transfer to an alternative supply when one is available and operating conditions permit.
6. To improve the responsiveness and reliability of smartgrid monitoring and fault management.
7. To assess performance using measures such as detection accuracy, response time, and restoration time.

## III. PROPOSED SYSTEM

The proposed framework contains sensing, data-processing, fault-classification, federated-learning, and control stages. Voltage and current sensors measure the electrical condition of the monitored circuit. A microcontroller receives the measurements and prepares them for analysis. A machine-learning model then classifies the operating condition as normal or potentially faulty.

Multiple monitoring nodes may train local models using their available data. The nodes send model updates to an aggregation server, which combines the updates and distributes a revised global model. Raw electrical measurements are not intended to be transferred as part of the normal federatedtraining process. The effectiveness of this approach depends on the quality and diversity of the local datasets and on reliable communication between nodes and the server.

When a fault is identified, the controller can operate the relevant relay to isolate the affected section. If a compatible backup supply is available, the system may transfer an eligible load to that source after checking the required protection conditions. The primary supply should only be reconnected after the fault has been cleared and normal conditions have been verified. These switching actions must follow suitable electrical protection and safety requirements.

## IV. SYSTEM METHODOLOGY

*A. Electrical Parameter Monitoring*

Voltage and current sensors provide measurements of the monitored electrical circuit. Changes in these measurements can indicate an abnormal or unstable operating condition. The readings are passed to the microcontroller for subsequent processing. The reliability of this stage depends on correct sensor selection, calibration, installation, and electrical isolation.

*B. Data Processing and Preprocessing*

Before fault analysis, the controller or connected processing unit prepares the collected measurements. Depending on the sensor and data format, preprocessing may include filtering measurement noise, checking whether readings are valid, and arranging the values in a format accepted by the machinelearning model. Consistent preprocessing is important because poor-quality or incorrectly formatted data can lead to incorrect classifications.

*C. Federated Learning*

Each participating monitoring node trains a local model using its available electrical data. Instead of sending the raw measurements to a central location, the node sends model updates to an aggregation server. The server combines the updates to produce a global model and sends the updated model back to the participating nodes. The process can be repeated as new local data become available.

Federated learning can reduce the need to centralize raw electrical data, but it does not by itself guarantee complete privacy or security. The implementation must consider communication protection, access control, the reliability of participating nodes, and the computing capacity of the devices. The aggregation method and model architecture must also be selected and evaluated during implementation.

*D. Intelligent Fault Detection and Isolation*

The trained model analyzes the processed electrical measurements and classifies the operating condition. If the output indicates a potential fault, the controller initiates the appropriate relay operation to disconnect the affected section. The isolation logic must be coordinated with electrical protection requirements so that switching does not create an unsafe condition or unnecessarily interrupt unaffected loads.

*E. Power Restoration and Performance Evaluation*

After a faulty section has been isolated, an eligible load may be transferred to a compatible backup source if the source is available and the protection conditions are satisfied. Once the fault has been cleared and normal operating conditions are confirmed, the load may be reconnected to the primary supply.

The proposed system can be evaluated using the following measures:

- Fault-detection accuracy: the proportion of evaluated cases classified correctly.
- False-alarm rate: the proportion of normal cases incorrectly identified as faults.
- Detection latency: the time between the occurrence or presentation of a fault condition and its detection.
- Switching time: the time required for the commanded switching operation to complete.
- Restoration time: the time needed to restore eligible loads after the fault has been cleared.

No numerical performance results are reported here because measured experimental data were not provided in the source draft. These metrics should be calculated from documented test runs before making quantitative performance claims.

*F. Process Flow*

Fig. 1 summarizes the intended sequence of monitoring, classification, isolation, and restoration.

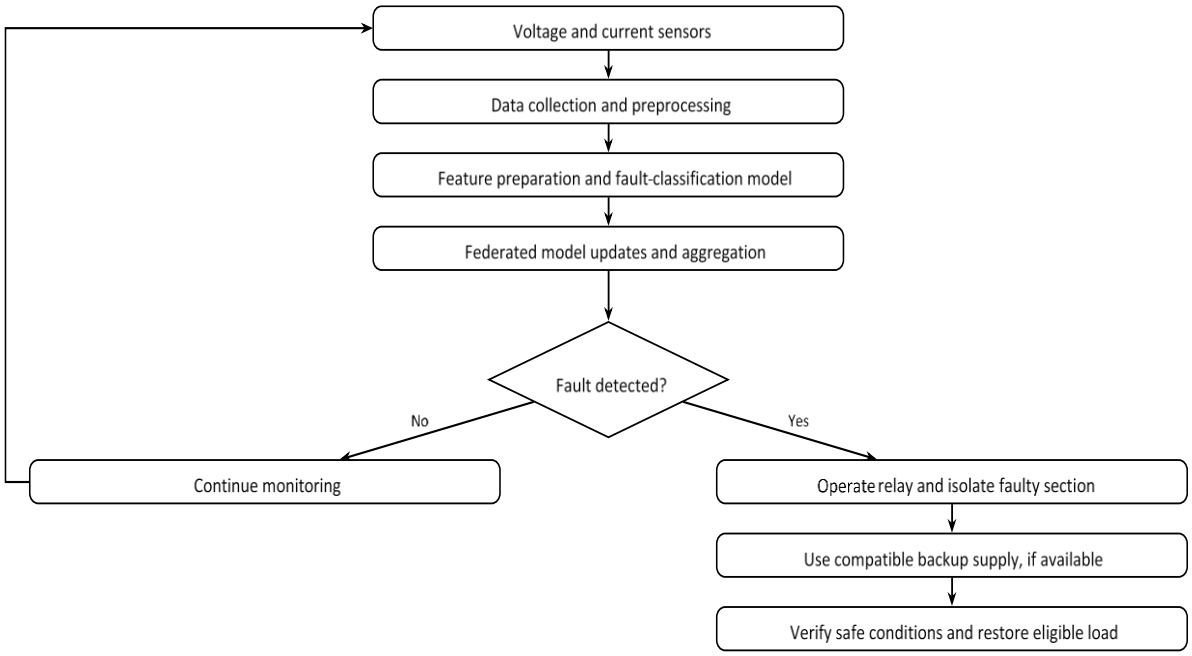



Fig. 1. Proposed monitoring and self-healing process. Federated model training supports the classification stage; the exact deployment arrangement depends

## V. AI AND MACHINE-LEARNING MODULE

The AI and machine-learning module is intended to analyze electrical measurements and distinguish normal operating conditions from possible faults. The data are collected through voltage and current sensors, checked and prepared for analysis, and supplied to the trained classification model. The model's output is passed to the controller to support a switching decision.

***A. Data Collection and Preprocessing***

Sensor readings are collected and prepared for model input. Filtering and validity checks may be applied where required by the sensor characteristics and the selected model. The training and testing datasets should represent the operating conditions and fault types that the system is expected to identify.

***B. Fault Classification***

The model classifies the processed measurements into the operating categories defined during system development. The fault categories, training procedure, and decision thresholds must be specified in the implementation and validated using labelled data. The current draft does not provide a complete dataset description or numerical classification results, so these details should be added after testing.

***C. Model Aggregation and Updating***

Local models are trained at participating nodes. The aggregation server combines their updates and distributes the resulting global model. The aggregation algorithm, communication protocol, update frequency, and model architecture should be documented in the final implementation so that the procedure can be reproduced.

### *D. Decision-Making and Control*

The controller uses the classification result to determine whether a fault-isolation action is required. Relay operation and power restoration must be subject to independent protection checks. A machine-learning prediction should not replace required electrical protection devices or safety interlocks.

## VI. HARDWARE AND SOFTWARE COMPONENTS

The components listed in this section are those proposed in the source draft. Their actual use should be confirmed against the final hardware and software implementation.

TABLE I

PROPOSED HARDWARE COMPONENTS AND ROLES

| Component | Proposed role |
|---|---|
| Voltage sensor | Measures voltage variations in the monitored circuit. |
| Current sensor | Monitors current flow and helps identify abnormal current conditions. |
| Microcontroller (Arduino/ESP32) | Receives sensor readings, processes measurements, and coordinates control actions. |
| Relay module | Switches the relevant circuit to isolate a faulty section, subject to protection requirements. |
| Backup power supply | Provides an alternative source when available and compatible with the load. |
| Power supply unit | Supplies the required operating voltage to the electronic components. |
| Communication module (Wi-Fi) | Supports data exchange between monitoring nodes and the server or dashboard. |
| Connecting wires and protection circuit | Connects the components and supports safe circuit operation. |

TABLE II

| Software/technology | Proposed role |
|---|---|
| Visual Studio Code | Writing, editing, and managing project code. |
| HTML5 | Structuring the web dashboard and its interface elements. |
| CSS3 | Styling the dashboard layout and responsive appearance. |
| JavaScript | Implementing dashboard interactions and display-side logic. |
| ESP8266 firmware | Supporting hardware communication and electrical telemetry when the device is integrated. |
| Web browser | Displaying the web-based monitoring dashboard. |

SOFTWARE COMPONENTS AND PROPOSED ROLES

*A. Hardware Components B. Software Components*

The source draft mentions Arduino/ESP32 as possible controller hardware and ESP8266 firmware in the software list. The final paper should identify the exact controller used in the working prototype and avoid presenting alternative boards as simultaneously implemented unless they were all tested.

## VII. ILLUSTRATIVE EVALUATION FRAMEWORK AND EXAMPLE DATA

Important data-integrity note: The original project document does not provide measured experimental observations, a labelled dataset, or completed model-evaluation results. Therefore, the values in this section are synthetic examples created only to demonstrate how results may be reported. They must not be described as real-life measurements or as results obtained from the project. Replace them with recorded data from the actual prototype before submitting the paper for publication. The figures are explicitly illustrative and do not establish the performance of the proposed system.

### *A. Suggested Test Procedure*

A reproducible evaluation should define the test circuit, sensor calibration, sampling interval, fault categories, protection settings, model architecture, training/test split, and the number of trials. Normal operation and each fault category should be tested separately. For each trial, record the sensor readings,

reference label, model prediction, time of fault injection, detection time, relay command time, confirmed isolation time, and restoration time. Tests involving mains voltage should only be carried out using appropriately rated equipment, isolation, protection, and qualified supervision.

For a binary normal/fault classifier, accuracy, precision, recall, and F1-score may be calculated as

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

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (3)$$

$$F_1 = 2 \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \quad (4)$$

where *TP*, *TN*, *FP*, and *FN* denote true positives, true negatives, false positives, and false negatives. Report the number of trials alongside each metric; accuracy alone can be misleading when fault cases are rare.

***B. Synthetic Example Observations***

Table III shows a small, hypothetical set of operating observations. The values are illustrative only and are not claims about a physical prototype. They demonstrate how voltage, current, model status, and a controller response can be organized in a results table.

TABLE III

SYNTHETIC EXAMPLE OBSERVATIONS (NOT MEASURED PROJECT DATA)

| Sample | Voltage (V) | Current (A) | Example label | Illustrative action |
|---|---|---|---|---|
| 1 | 230.1 | 1.8 | Normal | Continue monitoring |
| 2 | 229.4 | 2.1 | Normal | Continue monitoring |
| 3 | 228.7 | 3.9 | Abnormal current | Check / isolate |
| 4 | 231.0 | 2.0 | Normal | Continue monitoring |
| 5 | 184.2 | 2.7 | Undervoltage | Check / isolate |
| 6 | 230.5 | 5.2 | Overcurrent | Check / isolate |
| 7 | 226.8 | 2.2 | Normal | Continue monitoring |
| 8 | 0.0 | 0.0 | Supply loss | Backup only if safe |

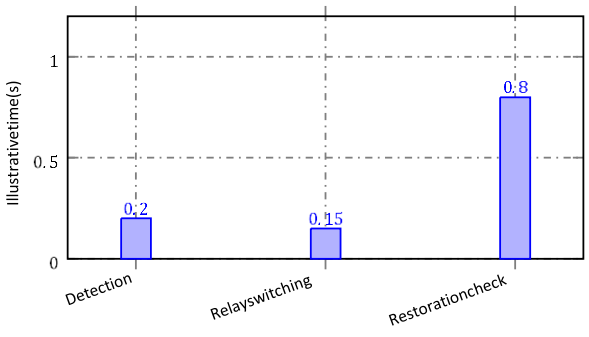



Fig. 2. Hypothetical response-time values for illustrating result presentation; not experimental measurements.

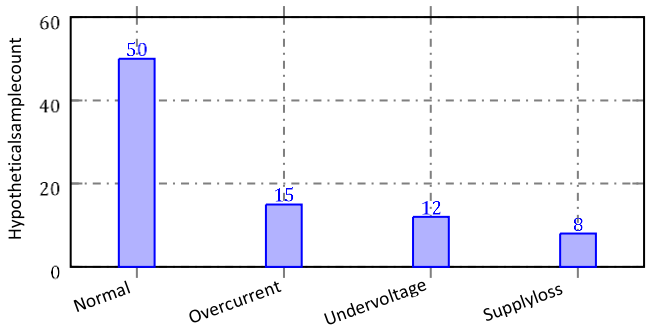



Fig. 3. Hypothetical class counts used only to demonstrate dataset reporting.

***C. Illustrative Response-Time Comparison***

For demonstration, assume hypothetical response-time observations for three stages of the control path. These example values can be replaced with measured timestamps from repeated trials. Figure 2 visualizes the illustrative values and should not be interpreted as a benchmark or a measured improvement over an existing system.

***D. Illustrative Normal and Fault-Class Counts***

The second chart gives a hypothetical class distribution for a mock dataset. It illustrates why a future experimental report should state the sample count for each class and should avoid relying on overall accuracy alone. These counts are not a description of a collected dataset.

***E. How Actual Results Should Be Reported***

Once tests have been performed, replace the synthetic table and chart values with the original recorded observations. Report the hardware configuration and calibration procedure, dataset size and source, fault-injection procedure, model and training parameters, number of participating federated nodes, aggregation method, and train/validation/test split. For response-time metrics, report the mean and standard deviation across repeated trials and, where possible, the minimum and maximum. Do not claim that federated learning improves privacy, accuracy, or speed quantitatively unless the claim is supported by a suitable comparison and measured evidence.

## VIII. LIMITATIONS AND FUTURE WORK

***A. Limitations***

The proposed POWER MONITOR dashboard is intended to support electrical monitoring and self-healing control. Its practical performance depends on reliable sensors, suitable hardware, and communication between the devices and the dashboard or server. A dashboard alone cannot guarantee realtime hardware operation without a working device interface and control path.

The accuracy of AI-based fault detection depends on the quality and coverage of the training data and on the model's implementation. Federated learning additionally requires participating nodes, local model training, and an aggregation mechanism. The proposed approach must be tested under different normal and fault conditions to establish its reliability, safety, and response time. Safe switching and restoration also require suitable electrical protection and interlocks.

***B. Future Work***

Future work can include integration of physical sensors and ESP8266 hardware for real-time monitoring, implementation and validation of a machine-learning model, and federated training across multiple monitoring nodes. Additional features may include cloud-based data storage, remote monitoring, fault alerts, and historical data analysis. Experimental evaluation should report the test setup, fault scenarios, dataset, model configuration, and measured detection and restoration metrics. Further safety testing can help assess reliability and practical applicability.

## IX. CONCLUSION

This paper has presented a proposed federated learningbased framework for smart-grid monitoring and self-healing. The framework combines voltage and current sensing, data preprocessing, machine-learning-based fault classification, relay-based isolation, and conditional power restoration. Federated learning allows participating monitoring nodes to contribute model updates without routinely transferring raw electrical measurements to a central server. The project material also describes a web-based dashboard named POWER MONITOR for monitoring and control-related functions.

The practical benefits of the framework depend on completing and validating the hardware, communication, machinelearning, and federated-learning components. Future testing should measure detection accuracy, false alarms, detection latency, switching time, and restoration time across documented operating conditions. These results are necessary before making quantitative claims about improved reliability or response speed.

**ACKNOWLEDGMENT**

The authors thank their project guide and faculty members for their guidance and support during the development of the project, "POWER MONITOR – AI Telemetry & SelfHealing." The authors also acknowledge Priyadarshini College of Engineering for providing learning opportunities and project resources, and thank their friends and families for their encouragement.

**REFERENCES**

1. D. Pansari and A. Yadav, "Adaptive Fault Diagnosis in Renewable Integrated Microgrids Using Hybrid Machine Learning Approach," *Electric Power Systems Research*, 2026. [Bibliographic details to be verified.]
2. P. Krishna, "Emerging AI for Resilient Smart-Grid Fault Detection: An Evidence-Based Capability Synthesis, Maturity Assessment, and Research Roadmap," *Journal of Electrical Systems and Information Technology*, 2026. [Bibliographic details to be verified.]
3. Z. Zhang et al., "Federated Learning for Smart Grid: Applications and Challenges," *ACM Transactions on Cyber-Physical Systems*, 2025. [Bibliographic details to be verified.]
4. M. Hassan et al., "Intelligent Self-Healing Smart Grid Framework," in *IEEE SmartGridComm*, 2025. [Bibliographic details to be verified.]
5. R. Chen et al., "Explainable AI for Smart Grid Fault Diagnosis," *IEEE Access*, 2025. [Bibliographic details to be verified.]
6. A. H. Ahmed et al., "Artificial Intelligence for Smart Grid Automation," *Energy Reports*, 2025. [Bibliographic details to be verified.]
7. R. Akter et al., "Privacy-Preserving Federated Learning in Smart Grid Networks," *IEEE Transactions on Industrial Informatics*, 2025. [Bibliographic details to be verified.]
8. C. Keçeci et al., "Distributed Detection and Mitigation of FDIAs in Smart Grids via Federated Learning," *International Journal of Electrical Power & Energy Systems*, 2025. [Bibliographic details to be verified.]
9. P. Zhang et al., "Privacy-Preserving Federated Learning Scheme for Distributed Smart Grid Based on Multi-Key Homomorphic Encryption," *Peer-to-Peer Networking and Applications*, 2025. [Bibliographic details to be verified.]
10. M. A. Elsadd et al., "A Novel Traveling-Wave-Based Detection and SelfHealing Approach for Series Faults in Smart Grids under Renewable Energy Uncertainty," *Scientific Reports*, 2025. [Bibliographic details to be verified.]
11. J. Li et al., "Secure Federated Learning for Smart Energy Systems," *IEEE Access*, 2024. [Bibliographic details to be verified.]
12. S. Deng et al., "Deep Learning-Based Fault Prediction in Smart Grid," *IEEE Transactions on Smart Grid*, 2024. [Bibliographic details to be verified.]
13. A. Kumar et al., "AI-Based Fault Detection in Smart Power Grid," *Energy Reports*, 2024. [Bibliographic details to be verified.]
14. Y. Zhou et al., "Distributed Federated Learning for Smart Grid Security," *IEEE Internet of Things Journal*, 2024. [Bibliographic details to be verified.]
15. H. Wang et al., "Machine Learning Techniques in Smart Grid Protection," *Applied Energy*, 2024. [Bibliographic details to be verified.]