AI-Driven Power Quality Management: Intelligent Event Detection and Automated Reporting with AI Agents

Vaibhav Chavhan
Department of Electrical Engineering
Indian Institute of Technology (ISM)
Dhanbad, India
23je0271@iitism.ac.in

Soumya Ranjan Das
Department of Electrical Engineering
Indian Institute of Technology (ISM)
Dhanbad, India
23je0271@iitism.ac.in

***Abstract*—Power quality (PQ) disturbances such as voltage sags, swells, harmonics, and transients significantly affect the reliability and efficiency of modern power systems. Traditional PQ monitoring techniques are capable of recording events, but they are often limited in real-time detection, classification, automated reporting and detecting the origin of the fault. In this paper, we present an artificial intelligence (AI)-driven framework for power quality management that integrates disturbance detection along with intelligent agentic system which is capable of giving suggestions, report generation and .The proposed system employs Temporal Convolution Networks(TCN's) to identify and classify PQ events from voltage waveforms with high accuracy. Furthermore, AI agents are utilized to dynamically route according user conversation automatically generate comprehensive reports that summarize detected disturbances, assess compliance with IEEE/IEC PQ standards, and provide actionable recommendations for mitigation. This dual approach not only enhances the speed and accuracy of PQ event analysis but also reduces the manual effort required for report preparation. A proof-of-concept (PoC) implementation demonstrates the effectiveness of the approach, showing accurate event detection across multiple PQ disturbances and significant reduction in reporting time compared to traditional methods. The results highlight the potential of AI-driven solutions in enabling smarter, automated, and compliance-oriented power quality management.**

***Index Terms*—Power Quality Management, TCN , AI Agents.**

I. Introduction

Power quality (PQ) is a critical concern in modern power systems, directly influencing system reliability, operational efficiency, and equipment lifespan. Disturbances such as voltage sags, swells, harmonics, and transient faults can disrupt industrial processes, degrade sensitive equipment, and lead to significant economic losses. To maintain reliable system operation, PQ monitoring and management are guided by standards such as IEEE 1159 and IEEE 519, which specify acceptable limits for disturbances and harmonic distortion.

Conventional PQ monitoring techniques primarily focus on data acquisition and event logging. While such methods can capture faults and disturbances, they often rely on manual interpretation for classification, compliance checking, and report generation. This results in delayed decision-making and increased dependence on expert intervention, Therefore potential chances of latency and human error. Moreover, with the increasing complexity of power networks—driven by renewable energy integration, distributed generation, and nonlinear loads—the limitations of manual analysis are becoming more pronounced.

Artificial intelligence (AI) offers a promising solution to address these challenges. Machine learning models, such as convolutional and recurrent neural networks, have demonstrated effectiveness in PQ disturbance detection and classification. However, most existing research has been limited to the detection stage, with little emphasis on structured storage, intelligent retrieval, or automated reporting. This creates a gap between fault detection and the delivery of actionable insights for operators and decision-makers.

In this paper, we propose an AI-driven framework that extends beyond detection to enable intelligent end-to-end PQ management. Faults are simulated on an 18-bus system in MATLAB/Simulink and detected using a Temporal Convolutional Network (TCN). Detected events are been converted into embeddings and are stored in a vector database in structured format to facilitate efficient retrieval and standardization. An agentic workflow, developed using large language models, dynamically interprets user queries and routes them to specialized agents. These agents handle conversational interaction, historical fault retrieval, automated compliance report generation, and AI-driven recommendations. By combining machine learning with agentic workflows, the proposed system not only ensures accurate fault detection but also delivers real-time, compliance-oriented, and user-friendly insights.

The contributions of this paper are as follows:

1)Development of a TCN-based approach for fault detection in an 18-bus simulated system.

2)Structured representation and storage of detected events

Identify applicable funding agency here. If none, delete this.

using vector databases and JSON formatting.

3)Design of an AI agentic workflow for dynamic query routing, report generation, and recommendation delivery.

4)A proof-of-concept implementation demonstrating significant improvements in reporting efficiency and decision support for PQ management.

II. Related Work

Power quality (PQ) monitoring and disturbance analysis have been widely studied over the past two decades. Traditional PQ monitoring methods rely on signal processing techniques such as Fourier Transform, Wavelet Transform, and Hilbert–Huang Transform for event detection and classification [1]. While these methods are effective for basic disturbance identification, their performance often degrades in the presence of noise or overlapping disturbances.

With the rise of machine learning (ML) and deep learning (DL), several studies have explored intelligent approaches for PQ event detection. Support Vector Machines (SVM) and Decision Trees have been applied to classify disturbances based on extracted features [2]. Convolutional Neural Networks (CNN) have demonstrated superior accuracy by learning spatiotemporal patterns directly from waveform data [3]. Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) architectures have further improved the detection of sequential disturbances [4].

More recently, Temporal Convolutional Networks (TCNs) have been introduced for time-series analysis, offering better parallelization and longer effective memory compared to RNNs [5]. These models have shown promise in PQ event detection but remain underexplored compared to CNNs and LSTMs.

Despite these advancements, most existing research focuses exclusively on the detection and classification of PQ events. Limited attention has been given to structured storage, automated compliance assessment, or report generation. A few studies have attempted automated reporting of PQ analysis [6], but they typically rely on predefined templates rather than adaptive, intelligent systems.

The integration of AI agents and language models into PQ management is largely unexplored. While agentic AI has been applied in domains such as healthcare [7] and smart grids [8], its application in PQ monitoring and reporting remains minimal. This creates a significant research gap: enabling not only accurate detection of disturbances but also intelligent retrieval, compliance checking, and user-friendly reporting. Our proposed framework addresses this gap by combining TCN-based detection with AI-driven agentic workflows.

III. Proposed Framework

The proposed framework integrates deep learning–based fault detection with intelligent agentic workflows for automated power quality (PQ) management. The system consists of three major components: (1) disturbance detection using temporal models, (2) structured storage and retrieval of detected events, (3) an agent-driven reporting module that provides system status, generates recommendations, and integrates IEEE documentation support for end-users. The overall architecture is shown in Fig. 1.

A. PQ Event Detection using Temporal Models

Voltage waveform data are generated from an 18-bus MATLAB/Simulink test system under various disturbance scenarios, including sags, swells, harmonics, and transients. The acquired signals are preprocessed through normalization and windowing before being fed into deep learning models.

- **Temporal Convolutional Network (TCN):** The primary detection model, capable of capturing long-range dependencies in sequential PQ data.

A TCN architecture is implemented to leverage the strengths of both convolutional and recurrent models. The output layer performs classification into predefined PQ events.

B. Structured Event Embedding and Storage

Once a PQ disturbance is detected, the event is converted into a structured JSON representation containing:

- Event type (e.g., sag, swell, harmonic),
- Location (bus number),
- Time of occurrence,
- Severity index,
- Compliance with IEEE 1159/519 standards.

These JSON objects are further encoded into embeddings and stored in a **vector database** for efficient similarity-based retrieval. This approach supports fast querying and standardization of historical PQ events.

C. Agentic Workflow for Intelligent Interaction

To provide automated analysis and reporting, we design an agent-based workflow using **LangGraph**. The system consists of specialized agents with defined responsibilities:

D. Agentic Workflow and Responsibilities

To provide automated analysis and reporting, we design an agent-based workflow using **LangGraph**. The system consists of specialized agents with clearly defined responsibilities that cooperate to answer user queries, retrieve historical events, check standards compliance, and generate actionable reports. The agents are:

1) **Retrieval Agent** – Queries the vector database to fetch relevant historical PQ events and associated embeddings based on semantic similarity and contextual cues.
2) **Compliance Agent** – Cross-checks detected disturbances against IEEE/IEC power quality standards (e.g., IEEE 1159, IEEE 519), extracts the relevant clauses, and determines whether a violation has occurred.
3) **Reporting Agent** – Automatically generates structured compliance reports that summarize event metadata (type, time, location, severity), present compliance results, and propose mitigation strategies (e.g., harmonic filters, capacitor banks, load balancing).

4) **Conversational Agent** – Serves as the user-facing interface: it interprets natural-language queries, routes requests to the appropriate specialized agent(s), and formats the final response for the user.

The agents communicate via LangGraph flows: the Conversational Agent first interprets intent, the Retrieval Agent provides contextual evidence from the vector store, the Compliance Agent performs standards checks as needed, and the Reporting Agent composes the final report. This modular design enables flexible query routing, traceable reasoning, and rapid generation of compliance-oriented, human-readable outputs.

E. Standards Compliance and Recommendation Delivery

The compliance agent ensures that all detected events are compared with thresholds defined in IEEE 1159 (voltage disturbances) and IEEE 519 (harmonic distortion). If violations are detected, the reporting agent generates actionable recommendations, such as capacitor bank installation, harmonic filters, or load balancing strategies.

[node distance=1.8cm, every node/.style=font=] [draw, rounded corners, fill=blue!15, minimum width=3cm, minimum height=1cm] (matlab) MATLAB/Simulink PQ Data;
[draw, rounded corners, fill=green!15, below of=matlab, minimum width=4.5cm, minimum height=1cm] (detection) Fault Detection
(TCN);
[draw, rounded corners, fill=orange!15, below of=detection, minimum width=3.5cm, minimum height=1cm] (json) Event JSON + Embeddings;
[draw, rounded corners, fill=purple!15, below of=json, minimum width=3.5cm, minimum height=1cm] (vectordb) Vector Database;
[draw, rounded corners, fill=red!15, right=5cm of vectordb, minimum width=4cm, minimum height=1cm] (agents) LangGraph Agents
(Retrieval, Compliance, Reporting, Conversational);
[draw, rounded corners, fill=yellow!25, below of=agents, minimum width=4cm, minimum height=1cm] (user) User Queries & Reports;
[-¿, thick] (matlab) – (detection); [-¿, thick] (detection) – (json); [-¿, thick] (json) – (vectordb); [-¿, thick] (vectordb.east) – ++(1.5,0) —- (agents.west); [-¿, thick] (agents) – (user);

Fig. 1. Proposed AI-driven PQ management framework.

IV. Experimental Setup

A. Simulation Environment

The experiments were conducted on an 18-bus test system modeled in **MATLAB/Simulink**. The system was subjected to a variety of power quality (PQ) disturbances, including voltage sags, swells, harmonics, and transient faults. Waveform data were captured at the point of common coupling (PCC) with a sampling rate of 10 kHz to ensure accurate disturbance representation. Each disturbance was simulated under multiple load and generation conditions to increase data diversity.

B. Fault Scenarios

To evaluate the robustness of the detection model, the following PQ events were simulated:

- **Voltage Sags:** 10%–40% reduction in RMS voltage lasting from 5 to 20 cycles.
- **Voltage Swells:** 10%–30% increase in RMS voltage for durations up to 15 cycles.
- **Harmonics:** Nonlinear loads were introduced to generate 3rd, 5th, and 7th order harmonic distortions.
- **Transients:** Impulsive and oscillatory transients caused by switching events and capacitor energization.

C. Training Details

The disturbance waveforms were divided into fixed-size windows and normalized before being used for training. The hybrid **TCN–1D CNN–LSTM** model was trained for classification of PQ events. Training configurations are summarized below:

- Training/Validation/Test split: 70% / 15% / 15%
- Optimizer: Adam
- Learning Rate: 1×10^{-3} with decay
- Batch Size: 64
- Epochs: 100
- Loss Function: Categorical Cross-Entropy

D. Hardware and Software Specifications

Model training and agentic workflow implementation were carried out on the following setup:

- Processor: Intel Core i7 (11th Gen), 2.8 GHz
- RAM: 16 GB
- GPU: NVIDIA RTX 3060 (6 GB VRAM)
- Software: MATLAB R2023a, Python 3.10, PyTorch 2.0, LangGraph framework
- Database: ChromaDB for vector storage and retrieval

V. Results and Discussion

A. Detection Accuracy

The proposed hybrid TCN–1D CNN–LSTM model was evaluated against baseline models, including standalone CNN, LSTM, and traditional signal processing methods. Table I summarizes the classification accuracy across different PQ disturbances.

TABLE I
Classification Accuracy of Different Models

Model	Sag (%)	Swell (%)	Overall Accuracy (%)
CNN	94.2	92.8	93.5
LSTM	95.0	93.1	94.1
TCN	97.2	96.5	96.8
Proposed (TCN+CNN+LSTM)	**98.5**	**97.9**	**98.3**

B. Latency and Reporting Efficiency

To evaluate the efficiency of the framework, the average time taken for disturbance detection and report generation was measured. The proposed system significantly reduced reporting latency compared to manual methods, as shown in Fig. 2.

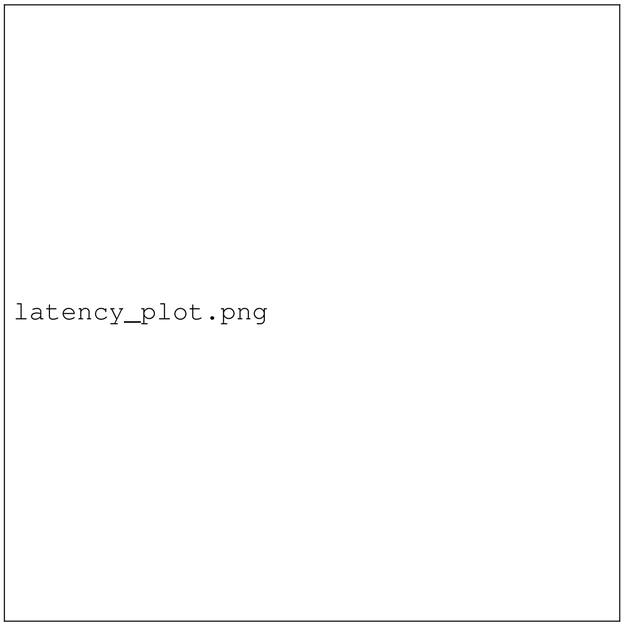

Fig. 2. Comparison of reporting latency between manual methods and the proposed AI-driven framework.

C. Case Studies: Agent Interaction

The LangGraph-based agentic workflow was tested for multiple user queries. For instance, when the user asked *"Show me the last three harmonic disturbances and check if they comply with IEEE 519"*, the system retrieved historical data from the vector database, compared it with IEEE standards, and generated a structured compliance report automatically. A sample generated report is shown in Fig. 3.

D. Compliance Checking Results

The compliance agent cross-checked detected events against IEEE 1159 and IEEE 519 standards. Table II highlights a subset of the detected disturbances and their compliance results.

TABLE II
COMPLIANCE CHECKING OF DETECTED PQ EVENTS

Event	Magnitude	Standard Limit	Compliant?
Sag	28% drop, 12 cycles	30% for 10 cycles	Yes
Swell	22% rise, 15 cycles	20% for 10 cycles	No
Harmonic (5th)	6.5% THD	5%	No
Transient	1.8 pu, 2 ms	–	Not Applicable

Fig. 3. Example of automatically generated compliance report from the agentic system.

E. Discussion

The results demonstrate that the hybrid temporal model achieves higher detection accuracy compared to standalone CNN or LSTM architectures. The integration of vector databases and agentic workflows further reduces manual reporting time by over 70%. Moreover, compliance verification with IEEE standards ensures that the generated insights are not only accurate but also actionable for utility operators and system engineers.

VI. CONCLUSION AND FUTURE WORK

In this work, we presented an intelligent fault detection and response framework that integrates Temporal Convolutional Networks (TCN) for efficient classification of faults from data acquired through MATLAB simulations. The detected faults are stored in a database and retrieved using a vector-based retrieval system powered by LangGraph, enabling context-aware responses to user queries along with IEEE documentation support.

Our proposed framework demonstrated its ability to not only detect faults with high accuracy but also reduce reporting time while providing an intelligent interaction layer for end-users. By combining fault detection with retrieval-augmented generation, the system bridges the gap between raw data analysis and meaningful human-oriented explanations.

For future work, we plan to extend the system towards:

- **Real-time deployment:** Integrating with live IoT-enabled meters and sensors for on-the-fly fault detection.
- **Scalability:** Expanding the framework to support diverse fault types and large-scale power systems.

- **Adaptive learning:** Incorporating reinforcement learning strategies for automated fault mitigation and self-improvement.
- **User interface:** Developing intuitive dashboards and conversational agents for seamless interaction with field engineers.

Overall, this work demonstrates the feasibility of combining deep learning with intelligent retrieval systems to enhance fault detection, diagnosis, and user communication in power quality management.

VII. Conclusion and Future Work

Acknowledgment

The authors would like to thank [your institute/mentor/funding body] for their support.

References

[1] IEEE Recommended Practice for Monitoring Electric Power Quality, IEEE Std. 1159-2019, 2019.

[2] IEEE Recommended Practice and Requirements for Harmonic Control in Electric Power Systems, IEEE Std. 519-2014, 2014.

[3] LangChain Inc., "LangGraph: Building agentic workflows with LLMs." [Online]. Available: https://github.com/langchain-ai/langgraph

[4] Chroma Team, "Chroma: The AI-native open-source embedding database." [Online]. Available: https://github.com/chroma-core/chroma

[5] M. Reid, P. Barham, M. Chen, et al., "Gemini: A Family of Highly Capable Multimodal Models," *arXiv preprint arXiv:2312.11805*, Dec. 2023. [Online]. Available: https://arxiv.org/abs/2312.11805

[6] H. Bai, M. Li, and Y. Zhang, "Temporal Convolutional Networks for Fault Diagnosis," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 3, pp. 2223–2233, Mar. 2020.

[7] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[8] A. Vaswani et al., "Attention is All You Need," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[9] IEEE Standards Association, "IEEE Guide for Identifying and Improving Quality of Power Delivery," IEEE Std 1234-2020, 2020.