IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 30 June 2025, accepted 19 July 2025, date of publication 22 July 2025, date of current version 28 July 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3591604

APPLIED RESEARCH

Explainable Graph Neural Networks for Power Grid Fault Detection

RICHARD BOSSO, COREY CHANG, MAHDI ZARIF, AND YUFEI TANG, (Senior Member, IEEE)
Department of Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431, USA

Corresponding author: Yufei Tang (tangy@fau.edu)

This work was supported by U.S. National Science Foundation under Grant CMMI-2145571, Grant OAC-2320972, and Grant IIS-2302786.

ABSTRACT This paper proposes the application of explanation methods to enhance the interpretability of graph neural network (GNN) models in fault location for power grids. GNN models have exhibited remarkable precision in utilizing phasor data from various locations around the grid and integrating the system's topology, an advantage rarely harnessed by alternative machine learning techniques. This capability makes GNNs highly effective in identifying fault occurrences in power grids. Despite their greater performance, these models can encounter criticism for their "black box" nature, which conceals the reasoning behind their predictions. Lack of transparency significantly hinders power utility operations, as interpretability is crucial to building trust, accountability, and actionable insights. This research presents a comprehensive framework that systematically evaluates state-of-the-art explanation strategies, representing the first use of such a framework for Graph Neural Network models for defect location detection. By assessing the strengths and weaknesses of different explanatory methods, it identifies and recommends the most effective strategies for clarifying the decision-making processes of GNN models. These recommendations aim to improve the transparency of fault predictions, allowing utility providers to better understand and trust the models' output. The proposed framework not only enhances the practical usability of GNN-based systems but also contributes to advancing their adoption in critical power grid applications.

INDEX TERMS Fault location detection, graph neural network (GNN), Fidelity, characterization score, model explainability and transparency, power grids.

I. INTRODUCTION

A. POWER GRID FAULT EVENT DETECTION

In a 3-phase power grid, short-circuit fault events are generally among the most common types of fault events that can occur [1]. Fault events can be caused by a variety of different phenomena, such as natural disasters, human error, equipment problems, or even man-made attempts to sabotage infrastructure. In many instances, short-circuit fault events can cause significant damage to equipment and costly interruptions for power delivery services if they are not discovered in a minimal amount of time [2]. For example, Figure 1 shows how damage caused by a tornado can impact power lines, and such damage can cause short-circuit fault events. Utility providers can incur significant costs whenever fault events lead to massive regional outages. However, even smaller and more isolated fault events can also cause damage and disruption that may hurt individual customers in a variety of different ways, and in some cases may have the potential to become a public health and safety issue [3]. Given the apparent benefits of detecting short-circuit fault events as quickly as possible, formal methodologies for determining where a fault event occurs in a power grid may help mitigate the damage caused by the fault event [4].

The research literature in the domain of fault event detection is populated with many different types of methods that have been proposed for detecting a variety of different types of fault events in power grids [4]. Some fault detection methods employ impedance-based analysis to estimate fault locations by examining how grid impedances are altered during fault events [5]. Other approaches utilize waveform analysis, relying on voltage or current readings to identify

The associate editor coordinating the review of this manuscript and approving it for publication was Sarasij Das.

FIGURE 1. Damage from natural disasters, such as damage caused by a tornado as shown here [8], can cause short-circuit fault events.

potential fault locations [6]. Additionally, research has demonstrated the use of time-frequency analysis, which leverages waveform readings and time-frequency data to estimate the location of higher-resistance fault events [7].

To develop more accurate fault detection methods using data from power grids, recent research has increasingly focused on implementing machine learning models to predict fault locations with high precision [9]. While a detailed exploration of other fault detection methods is beyond the scope of this work, our study centers on the application of a Graph Neural Network (GNN)—a machine learning model designed to capture and learn from the graph structure of data [10], [11]. We show that GNNs are highly effective for fault location detection and, when combined with a proposed explainability framework, offer improved transparency compared to current GNN-based fault detection approaches.

B. CONTRIBUTION

Building on our preliminary work [12], this paper shows that explanation methods can be used to make short-circuit fault location detection by GNN models more understandable and more transparent for power systems. With the development of our Explainable Graph Neural Network (EGNN), we also apply an EGNN Evaluation Framework that can systematically inform us how well any applied GNN explanation methods may actually perform. Given the scope of this work, we propose several unique contributions:

- A high-performance Explainable Graph Neural Network (EGNN) is proposed for short-circuit fault location detection in power systems.
- GNN models often operate as "black boxes," leading to challenges in model transparency and trust. To address this, we integrate various explanation methods to make the EGNN's decision-making process more interpretable.
- We propose to use a robust evaluation framework for power grid fault detection, enabling a comprehensive comparative analysis of different explanation methods compatible with our EGNN.

The remainder of this paper is organized as follows: Section II reviews the literature on the application of machine learning methods to fault detection in power grids. Section III provides an overview of GNN models and introduces the proposed explanation evaluation framework. Section IV details the case studies, discusses the results of these case studies, and demonstrates the application of the proposed EGNN explanation methods. Finally, Section V concludes the paper.

II. RELATED WORKS

A. MACHINE LEARNING FOR GRID FAULT DETECTION

Machine learning has been widely applied in power systems for stability and reliability, with notable potential in Smart Grids managing diverse alternative energy resources [13], [14]. For fault detection, the literature is populated with many recent examples that use measured voltage and current values to train a variety of different machine learning models [15]. A k-Nearest Neighbor (kNN) model was tested with different feature selection methods for fault detection and classification in a smart grid system [16], a Long Short-Term Memory (LSTM) model [17] was used to detect high-impedance fault events in a microgrid system in the context of variable weather conditions [18], and different deep learning models which included variations of Restricted Boltzmann Machine (RBM) and Convolutional Neural Networks (CNN) were used to detect and classify faults in transmission lines [19].

B. GRAPH LEARNING MODELS FOR FAULT DETECTION

For much of the previous literature discussed, many of the applied models didn't incorporate the actual topology of power grid structures, while research interest in applying graph learning models to power systems has grown significantly in recent years [20]. When compared to more traditional machine learning models, or models that can't take power grid topology into account, graph learning models have generally shown better performance for anomaly detection [21], [22], [23], [24]. These graph learning models are a type of deep learning that are generally categorized as Graph Neural Network (GNN) models [25]. In [21], a GNN model based on Chebyshev spectral graph learning [26] was found to perform better at fault location detection when compared to other non-graph learning methods. Using a model based on the GNN designed by Kipf and Welling [27], a comparison with several non-graph learning models found that the GNN was more effective at detecting and classifying different types of faults in transmission lines [22]. In [28], a GCN-based fault detection method was proposed for low-voltage DC microgrids, outperforming CNN, SVM, and FCN in accuracy and robustness. From research published in 2023, a heterogeneous multi-task learning GNN (MTL-GNN) was found to be effective at various prediction tasks related to fault classification [23]. In addition, a location method using a deep Graph Convolutional Network (GCN) that integrates system topology and multiple bus measurements was proposed in [29]. Tested on the IEEE 123-bus system, it outperforms traditional and machine learning methods in accuracy and robustness to noise and data loss. In 2024, using the novel PowerGraph benchmark dataset that they developed, Velickovic et al,. found that several variations of GNN model architectures, including Graph Attention Networks (GAT) [30], modified Graph Isomorphism Networks (GINe) [31], and Transformers [32], all performed effectively for the detection of cascading failures in power systems [24].

C. MAKING GRAPH LEARNING MODELS EXPLAINABLE

Despite the competitive performance that has been attributed to GNN models across many different domains, deep learning models are often criticized for their lack of transparency since the logic of these models is often impossible to interpret on their own [33]. When any model lacks transparency, even when such a model exhibits experimentally strong performance, the lack of informativeness that comes from these "black box" predictions can make it very difficult to trust such a model with real-world applications [34].

Fortunately, a wide range of supplementary explainability methods has been developed to address the limitations of "black box" models [34], including several designed specifically for GNN models [33]. While most explanation methods in the research literature for power systems and smart grids do not focus on GNNs [35], [36], some notable examples demonstrate the application of such methods to GNN models within the power grid domain [24], [37], [38]. Using a GNN model for the detection of cascading power failures, the explanation visualization method Layer-wise Relevance Propagation (LRP) [39] was used to help explain the logic behind the model by determining what factors in the power system were considered most relevant by the model [37]. Showing how a Spatio-Temporal Graph Neural Network (STGNN) could be used to predict energy production for photovoltaic (PV) units distributed in a power system, a GNNExplainer [40] was used to generate explanations to show what aspects of power system topology and node features represented the most important patterns in the data [38]. With the models trained by their PowerGraph benchmark data, several different explanation methods were tested with GNN models to determine the relative effectiveness of these different approaches [24].

III. PROPOSED APPROACH

A. PRELIMINARIES: CHEBYSHEV SPECTRAL GNN

For this work, the GNN is based off of the spectral graph network from Defferrard et al. [26], which is also described by Kipf and Welling [27]. We define a graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$, where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ is the set of edges, and A is the adjacency matrix defined as $A \in \{0, 1\}^{|\mathcal{V}|\times|\mathcal{V}|}$ with $|\mathcal{V}|$ the number of nodes. Given that the normalized adjacency matrix $\mathcal{A}$ is defined as $\mathcal{A} = D^{-1/2}AD^{-1/2}$, D is the diagonal degree matrix such that $D \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ and $D_{ii} = \sum_j A_{ij}$, and I is the identity matrix. Since the standard Laplacian matrix L_G is defined as $L_G = D - A$, the normalized Laplacian matrix L can be derived such that:

$$\begin{aligned} L &= D^{-1/2}L_G D^{-1/2} = D^{-1/2}(D - A)D^{-1/2} \\ &= D^{-1/2}DD^{-1/2} - D^{-1/2}AD^{-1/2} \\ &= I - D^{-1/2}AD^{-1/2} \end{aligned} \tag{1}$$

Since the Laplacian matrix is defined to be a positive semidefinite matrix, partly due to A being a symmetric matrix and D being a diagonal matrix, $L = U\Lambda U^T$ can be derived through eigendecomposition, such that $U \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ contains column-wise eigenvectors corresponding to L and $\Lambda \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ is a diagonal matrix of eigenvalues. Given a signal vector $x \in \mathbb{R}^{|\mathcal{V}|}$ representing a vector of scalar values from each node in $\mathcal{G}$, designating $U^T x$ as the graph Fourier transform of x, and f_θ being a filter operation that is a function of the eigenvalues of L as $f_\theta(\Lambda)$, this can be derived as:

$$f_\theta * x = f_\theta(L)x = f_\theta(U\Lambda U^T)x = Uf_\theta(\Lambda)U^T x \tag{2}$$

where polynomial parameters using the coefficients $\theta \in \mathbb{R}^K$ can be used to calculate f_θ as:

$$f_\theta = \sum_{k=0}^{K-1} \theta_k \Lambda^k \tag{3}$$

with K defined as the degrees in the polynomial. However, considering how this series of matrix multiplications with these polynomial coefficients are noted for being computationally complex [26], such an approach would not be a scalable solution with larger graphs, such as large power systems with many nodes. Alternatively, this polynomial expansion can be more efficiently approximated using a recursive algorithm, which in our case is the Chebyshev polynomial expansion $T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x)$ that is recursively defined as $T_0 = 1$ and $T_1 = x$, such that $T_k(\hat{\Lambda}) \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ represents each k-th order of Chebyshev polynomial, defined as:

$$f_\theta \simeq \sum_{g=0}^{K} \theta_k T_k(\hat{\Lambda}) \tag{4}$$

where $\hat{\Lambda}$ is defined to be a normalized version of Λ, such that:

$$\hat{\Lambda} = 2/\lambda_{max} \times \Lambda - I \tag{5}$$

where λ_{max} is defined as the largest eigenvalue found in the matrix of eigenvalues Λ derived from the Laplacian matrix L. Using the definitions that we have already established, we can derive $U(\Lambda)^k U^T = (U\Lambda U^T)^k = L^k$ as a general property to further extend this recursive Chebyshev approximation to the Laplacian matrix L itself, such that:

$$f_\theta * x \simeq U\sum_{k=0}^{K} \theta_k T_k(\hat{\Lambda})U^T x = \sum_{k=0}^{K} \theta_k T_k(\hat{L})x \tag{6}$$

where $\hat{L}$, similar to the calculation performed to generate the normalized eigenvalue matrix $\hat{\Lambda}$, is defined as:

$$\hat{L} = 2/\lambda_{max} \times L - I \tag{7}$$

The Chebyshev spectral graph algorithm defined in this section serves as the design for the graph convolutional filter used in our EGNN model. By leveraging the Chebyshev filter's ability to operate on node signal vectors x, it can be extended to node feature matrices in conjunction with adjacency matrices, which are the standard inputs for GNN models. This spectral graph convolutional operator enables our GNN model to learn from both nodal features and graph topology, as captured in our power grid data. Incorporating this topology-aware approach allows the graph convolutional filters to scale effectively with larger and more complex power system data, enhancing the model's ability to capture deeper insights from the data's structural context.

B. USING GNN FOR FAULT LOCATION DETECTION

Since the GNN model in this work is designed for graph classification, the training and testing data consist of input graphs containing voltage data recorded by Phasor Measurement Units (PMUs) located at various bus locations. These graphs also include the power grid topology corresponding to different combinations of fault scenarios.

For each input graph, we define **n** as the number of nodes and **d** as the number of features for each node, such that the node feature matrix for the input graph is defined as $X \in \mathbb{R}^{\mathbf{n}\times\mathbf{d}}$, with an accompanying adjacency matrix defined as $A \in \{0, 1\}^{\mathbf{n}\times\mathbf{n}}$ with 1 indicating that the nodes referred to by the embedding do connect and 0 otherwise. Given that this data is recorded using PMU installed at each bus location under different fault conditions, where each input graph represents PMU readings from any specific fault scenario, the nodes represent the total of **n** buses in a power system where **m** voltage features are recorded from each bus, and the connections shown by the adjacency matrix A represent the power line connections between each bus.

The phasor data recorded by each PMU consists of voltage magnitudes V_i and voltage angles $\angle_i$ for each i-th phase of a three-phase power line, such that $(V_1, \angle_1, V_2, \angle_2, V_3, \angle_3) \in \mathbb{R}^6$ is measured from each bus during a fault event. In cases where a bus is connected through only one phase or two-phase power lines, values of V_i and $\angle_i$ that would correspond to any phases i that are absent from that bus measurement are recorded as $V_i = 0$ and $\angle_i = 0$. With input graphs that are structured in this way, our GNN model uses this input data to generate predictions that attempt to indicate where a fault event has occurred.

For this work, the GNN architecture shown in Figure 2 is built with three Chebyshev graph convolutional layers using PyTorch Geometric [41]. Each of the graph convolutional layers is followed by Rectified Linear Unit (ReLU) activation functions. The degree of polynomial approximation for the Chebyshev filters differ for each graph convolutional layer, such that the depth of the Chebyshev polynomial approximates by order 3 for the first layer, a degree of order 4 for the second layer, and a degree of order 5 for the third layer. After passing through all three of the graph convolutional layers, they are followed by a mean pooling layer, which leads to a dropout layer with a dropout rate of 0.2, before then being passed to the last linear layer that produces the vector output values that are used for the fault location prediction. Using the Adam optimizer [42] to minimize the cross entropy loss function used with multi-class classification, a set learning rate of 0.001 was found to be effective, and the training data was processed with batch sizes of 32.

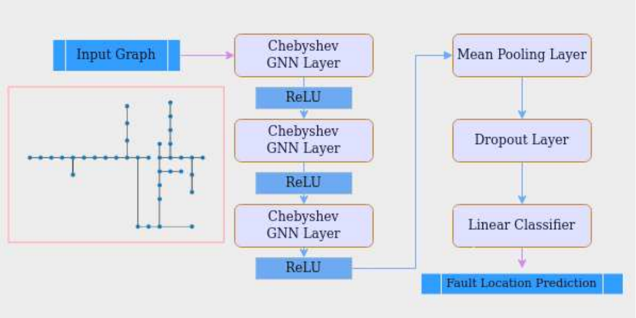

FIGURE 2. The EGNN model architecture used for fault location prediction.

C. EXPLANATION EVALUATION FRAMEWORK

Multiple explanation methods were applied to the EGNN, making it essential to assess the effectiveness of each method in generating explanations that accurately reflect the underlying logic of the EGNN model. To enhance the explainability of our fault location detection EGNN model, we propose the application of the EGNN Evaluation Framework, detailed below. This framework involves reporting the Fidelity+, Fidelity-, and Characterization Score performance metrics for each explanation method. By using these metrics, we can comprehensively measure and compare the effectiveness of each explanation approach.

1) FIDELITY+ AND FIDELITY-SCORES

The Fidelity metric, which was originally defined by [33], was extended in [11] to also account for whether the explanation method focused on explaining the phenomenon or the model itself. When using the Fidelity metric, two different scores can be generated, namely Fidelity+ and Fidelity-. They both are designed to indicate how well the subgraphs generated by the explanation methods can reflect either the patterns in the phenomenon or the inner-workings of the model itself, and Fidelity+ and Fidelity- each approach this from different perspectives [11], [33].

With some exceptions, explanation methods included in our study are designed to generate an edge mask M_E and a node feature mask M_{NF}. For each input graph, we already defined **n** as the number of nodes and **d** as the number of features for each node, such that the node feature matrix for the input graph is defined as $X \in \mathbb{R}^{\mathbf{n}\times\mathbf{d}}$, with each Adjacency matrix defined as $A \in \{0,1\}^{\mathbf{n}\times\mathbf{n}}$. Each G_S subgraph is built to include an adjacency matrix A_S and a node feature matrix X_S, which are each derived from Hadamard products of the graph input data and the explanation masks such that $A_S = M_E \odot A$ and $X_S = M_{NF} \odot X$ define G_S as a subset of the entire input graph.

Though M_E and M_{NF} have similar dimensions to A and X respectively, such that $M_E \in \mathbb{R}^{\mathbf{n}\times\mathbf{n}}$ and $M_{NF} \in \mathbb{R}^{\mathbf{n}\times\mathbf{d}}$ when generated by an explanation method, both masks can be further transformed to be either a *soft mask* or a *hard mask*, depending on the preferred form of explanation [11]. Whereas soft masks are normalized so that their weight values are between 0 and 1, such that $M_E^{soft} \in [0,1]^{\mathbf{n}\times\mathbf{n}}$ and $M_{NF}^{soft} \in [0,1]^{\mathbf{n}\times\mathbf{d}}$, hard masks are further transformed such that their values are only 0 and 1, such that $M_E^{hard} \in \{0,1\}^{\mathbf{n}\times\mathbf{n}}$ and $M_{NF}^{hard} \in \{0,1\}^{\mathbf{n}\times\mathbf{d}}$. Additionally, these explanation masks can also be further transformed for different levels of TopK, a threshold that determines the size of the explanation subgraph G_S by adjusting the k number of edges based on mask weights.

For examining how well an explanation method performs at reflecting the underlying logic of a GNN model, given that our GNN model is trained for the task of graph classification, Fidelity+ and Fidelity- are defined as:

$$\mathrm{fid}_+ = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i^{G_{C\setminus S}} = \hat{y}_i) \tag{8}$$

$$\mathrm{fid}_- = 1 - \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i^{G_S} = \hat{y}_i) \tag{9}$$

where N is the total number of instances in a given sample of graph data inputs, $\hat{y}_i$ is the predicted label generated by the trained GNN model for each i_{th} graph input, $\hat{y}_i^{G_S}$ is the label predicted by the GNN for each i_{th} subgraph G_S which is generated by the explanation method for each graph input, and $\hat{y}_i^{G_{C\setminus S}}$ is the label predicted by the GNN for each i_{th} complement of the subgraph G_S which are denoted as $G_{C\setminus S}$.

Conceptually, Fidelity+ and Fidelity- reflect different ways to evaluate the quality of the explanations provided by the subgraphs that are generated from the explanation method. Since Fidelity- essentially reflects how consistently the explanation subgraph G_S points to the prediction generated by the model, Fidelity- indicates how well an explanation method can generate *sufficient* explanations. Conversely, Fidelity+ reflects how consistently the complement to the explanation subgraph $G_{C\setminus S}$ points to the model prediction, such that Fidelity+ indicates how well an explanation method can generate *necessary* explanations. Based on the assumptions taken by the calculations for Fidelity+ and Fidelity-, an explanation method with Fidelity+ scores near 1 and Fidelity- scores near 0 are considered to be more necessary and more sufficient, respectively [11]. For ease of visualization and more convenience with reporting of experimental results, Fidelity- scores are reported here as (1 – Fidelity-).

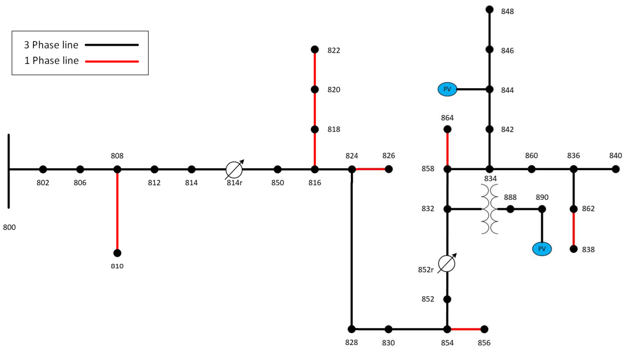

FIGURE 3. IEEE 34 bus system modified with added PV units.

2) CHARACTERIZATION SCORE

Once Fidelity+ and Fidelity- are both calculated, Characterization Score can also be used as an additional evaluation metric, which considers the strength of an explainability method indicated by Fidelity+ and Fidelity- simultaneously. Characterization Score is defined as:

$$\mathrm{charact} = \frac{w_+ + w_-}{\frac{w_+}{\mathrm{fid}_+} + \frac{w_-}{1-\mathrm{fid}_-}} \tag{10}$$

where w_+ is defined as the weight of importance that can be attributed to the fid_+ score (Fidelity+), and w_- is defined as the weight of importance attributed to the fid_- score (Fidelity−). For our purposes, the Characterization Score is calculated with w_+ and w_- both set to a value of 0.5, which assumes equal importance for both Fidelity+ and Fidelity-.

3) EVALUATION FRAMEWORK

The explanation evaluation framework we propose for assessing explanation methods in the context of using EGNN (or any explainable GNN model) to predict fault event locations in a power grid allows for evaluating the effectiveness and consistency of these methods. By simultaneously reporting the Fidelity+, Fidelity-, and Characterization Scores, we can determine how often the method provides effective explanations. Reporting all these performance metrics provides clearer insights into how consistently an explanation subgraph captures the key node features and edges of input graph data that influence the EGNN's decision-making process.

IV. CASE STUDY

A. BENCHMARK SYSTEMS AND DATASETS

For this work, simulations of short-circuit fault events were generated using the IEEE 34 and IEEE 123 radial test feeder systems [43], [44] and the more recently developed 342-node Low Voltage Networked (LVN) Test System [45]. OpenDSS [46] and its official python API *py-dss-interface* [47] were used to implement modified versions of these test systems, and for each system the fault simulations had fault events generated for every possible bus location. For all three systems tested here, they were modified to have photovoltaic (PV) units added at various bus locations. To simulate how grid system loads and power generation from PV units can vary throughout typical day cycles, different solar irradiance levels for each PV unit and changes in the base system load were introduced across different fault scenarios [21]. In addition to having various fault simulation scenarios, different 1-phase, 2-phase, and 3-phase fault-type events were also introduced, including various line-to-ground, line-to-line, and line-to-neutral faults.

Similar to the rates of differentiation applied in [21], the irradiance levels simulated for each PV unit ranged between 0 - 1 kW/m^2, and the base load of the entire system was adjusted between multiples of 0.4 and 1.2. By differentiating combinations of fault event conditions for fault types, fault locations, levels of system loading, and irradiance levels impacting PV energy production, large enough sample sizes of power system simulation data could be generated for use as training and testing data. Similar to noise injection described by [48], Gaussian noise was added to the testing data under the assumption that noise with a standard deviation of 3% makes the results more realistic.

1) IEEE 34 BUS SYSTEM

The IEEE 34 bus system was modified to have PV units added to bus locations 890 and 844 [21], as shown in Figure 3. Each PV unit had a nominal voltage rating of 4.16 kV, with the bus 890 PV unit having a power capacity of 270 kW and the bus 844 PV unit having a power capacity of 245 kW. The training and testing data applied to our graph and explanation model for the IEEE 34 case study comprised a total of 26,820 fault scenarios. The fault scenario simulation data is transformed for graph classification, such that each fault scenario instance has a corresponding label for the supervised learning process. The IEEE 34 system's 34 buses are grouped into 24 unique fault locations by assigning the same label to buses within 1,000 ft of each other or connected via transformers or regulators. This approach considers the topological proximity of bus locations for each fault location label, ensuring models do not need to differentiate between fault locations with minimal separation. While most bus locations in the IEEE 34 system are linked by 3-phase lines, some (e.g., buses 810, 818, 820, 822, 826, 838, 856, and 864) are connected only by 1-phase lines. For these locations, only 1-phase fault scenarios are used in simulations.

2) IEEE 123 BUS SYSTEM

For the IEEE 123 bus system, PV units were installed in bus locations 79, 95, 250, 300, and 450 [49], and they were each installed with a power capacity of 450 kW and a voltage rating of 4.16 kV. The IEEE 123 system includes not only 3-phase lines but also 1-phase lines, 2-phase lines, and closed switches, all of which are represented through node adjacency. While a small minority of buses are connected exclusively via 2-phase lines, nearly half of all buses are connected via one-phase lines. For these cases, only applicable fault simulation scenarios are applied. To generate the 38,450 fault scenarios for the IEEE 123 system, bus nodes sharing regulators or closed switches were grouped into common labels, as the distance between such nodes is negligible. This process resulted in 120 distinct fault location labels. Figure 4 illustrates the modified IEEE 123 system with added PV units, where 1-phase, 2-phase, and 3-phase lines are color-coded as shown.

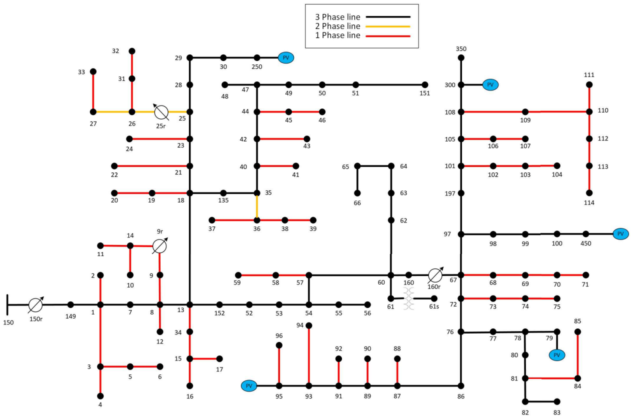

FIGURE 4. IEEE 123 bus system modified with added PV units.

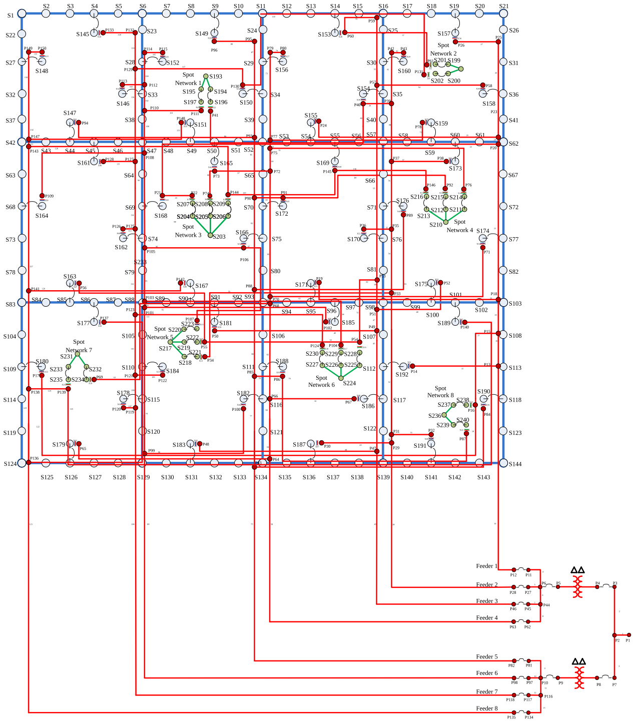

FIGURE 5. The 342 LVN System [44], [45] visualized to show the 48 buses from the 8 spot networks [50]. For this study, PV units were installed at each of the spot network bus locations.

3) 342-NODE LVN BUS SYSTEM

The 342-node LVN system is also referred to as the IEEE 390 system, in reference to the 48 additional buses that manage different networks of spot loads. This LVN system was modified to have 48 PV units, each with a power capacity of 300 kW and a 0.48 kV voltage rating, added to the 48 secondary buses S193 - S240 that support 8 networks of spot loads present in the LVN system [45]. Shown in Figure 5, the 342 LVN system is constructed to have a primary and secondary set of connections. All lines and buses colored red in Figure 5 comprise the primary section, while every other bus connection comprises the secondary section. Similar with other benchmarks, for the the 342-node LVN system, any bus locations that share the same closed switch or transformer are grouped together within the same classification label, and any buses within the primary section of the 342 LVN system that aren't directly connected via a transformer or closed switch to buses in the secondary section are all grouped within the same fault location label, such that there are a total of 186 fault location classification labels encoded for the 31,200 samples generated. In addition to using 3-phase line connections between each bus to represent the topology, closed switches are also represented by graph data topology as connections between bus locations. Though there has been previous work that used the 342 LVN system for binary detection of power grid system outages [51], to the best of the author's knowledge there exists no previous work that uses the 342 LVN bus system as a benchmark for fault location prediction and there are no other known instances of any type of fault detection being applied to the 342 LVN bus system when the system has been modified with added PV units.

B. MODEL PERFORMANCE COMPARATIVE STUDY

1) OVERVIEW AND IMPLEMENTATION OF FAULT DETECTION MODELS

Along with our GNN model, we also tested k Nearest Neighbors (kNN), Multi-Layer Perceptron (MLP), and Naive Bayes models for fault detection with each of our case studies. Aside from our GNN, each of these models were implemented using Scikit-learn [52]. For our kNN model, K (which reflects clustering distance between data samples based on their features) was set to 1. Our MLP was trained for 200 epochs, similar to the number of epochs that were spent training our GNN model. Our Naive Bayes model was implemented as a Gaussian variant. For each of these models, alongside our GNN, the Testing Accuracy, F-1 Measure, and AUC were reported for each them along an 80 % - 20 % training-to-testing data ratio. For our purposes, we report the Macro-F1 score for each of the model prediction results. In our case studies the AUC scores for each class label are calculated against the rest of the data, and the unweighted mean of each AUC score for each class label is reported as the AUC metric.

For the labeling of the fault locations being detected for each bus system, the IEEE 34 bus system was assigned 24 classes, the IEEE 123 bus system was assigned 120 classes, and the 342 LVN bus system was assigned 187 classes, so from a machine learning perspective each of these bus systems are represented by multi-class datasets. Such numbers of classes in these datasets may generally signify some class imbalance by default, but the fact that all 390 buses of the 342 LVN bus system were aggregated into 187 classes makes the dataset considerably more imbalanced than the other bus systems. When trying to assess model performance with imbalanced data, testing accuracy can be misleading if used as the only performance metric [53], and for multi-class data the sensitivity, specificity, and precision rates can provide a comparison of each model that takes model performance by class into account. Given the task of predicting the bus locations of fault events, we report scores for AUC, F-1 Measure, and Testing Accuracy for each model, in order to make our comparative study more representative of model performance.

TABLE 1. Model comparative study for IEEE 34 bus system.

Models	AUC	F-1 Measure	Accuracy
EGNN	**0.9991**	**0.9693**	**0.9607**
k-NN	0.9816	0.9636	0.9544
MLP	0.9988	0.9592	0.9450
Naive Bayes	0.5051	0.0689	0.0087

TABLE 2. Model comparative study for IEEE 123 bus system.

Models	AUC	F-1 Measure	Accuracy
EGNN	**0.9999**	0.9822	**0.9905**
k-NN	0.9943	**0.9890**	0.9826
MLP	0.9998	0.9677	0.9486
Naive Bayes	0.5035	0.0016	0.0507

TABLE 3. Model comparative study for 342 LVN bus system.

Models	AUC	F-1 Measure	Accuracy
EGNN	**0.9994**	**0.9046**	**0.8552**
k-NN	0.9337	0.8684	0.7606
MLP	0.9948	0.6877	0.6865
Naive Bayes	0.8134	0.2268	0.1811

2) MODEL COMPARATIVE STUDY PERFORMANCE RESULTS

The Accuracy, F-1 Measure, and AUC scores are averages taken from 5 different sampled testing scenarios, with Table 1, Table 2, and Table 3 showing performance results for the IEEE 34 bus system, the IEEE 123 bus system, and the 342-node LVN bus system respectively. Though the GNN was trained for 200 epochs to result in the performance reported for the IEEE 34 and IEEE 123 bus systems, for the 342 LVN system our GNN was trained for 350 epochs, likely necessitated by the large number of classes and greater complexity inherent to the 342 LVN system. The confusion matrix showing the EGNN model performance for the IEEE

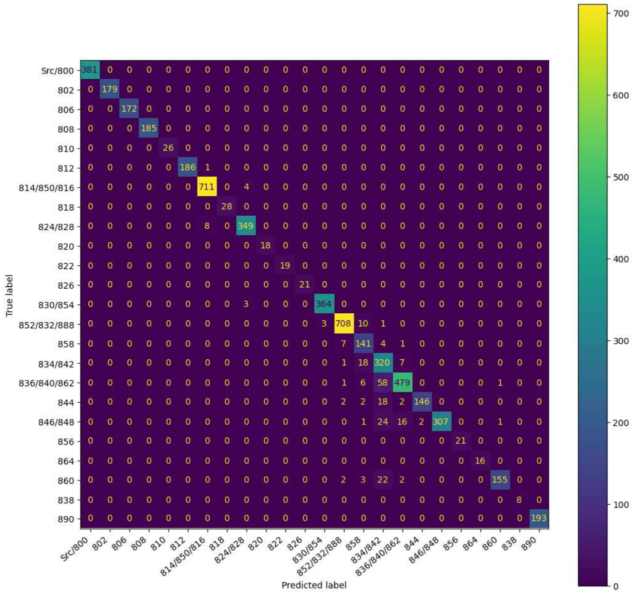

FIGURE 6. Confusion matrix showing EGNN classification performance for IEEE 34 bus system.

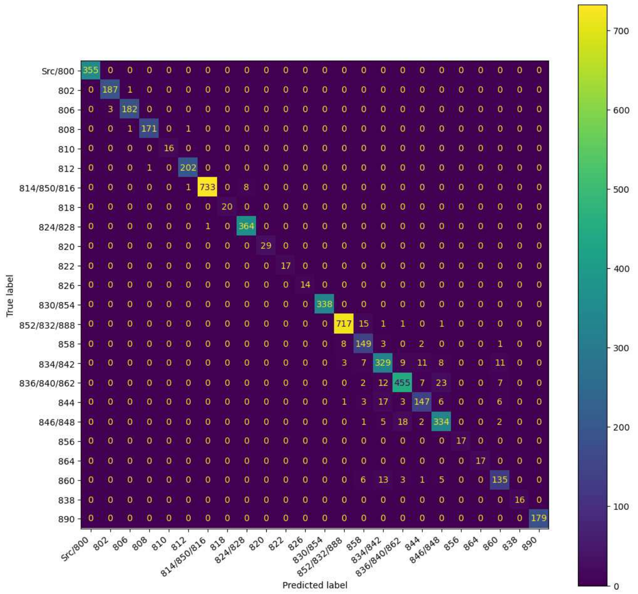

FIGURE 7. Confusion matrix showing kNN classification performance for IEEE 34 bus system.

34 bus system is shown in Figure 6, and the confusion matrix showing the EGNN model performance for the 342 LVN bus system is shown in Figure 10, where the vertical axis represents the true labels for each class and the horizontal axis represents the label predictions that were actually generated.

Looking at the performance metric results, the EGNN model generally outperformed all of the other models. The only exception is the k-NN model for the 123 bus system, which has acquired a slightly higher F1-Measure than the EGNN model, despite the EGNN model having the highest AUC and Testing Accuracy among these models. Given the 123 bus system, the k-NN model's higher F-1 Measure score might suggest that the k-NN model may have generated predictions with higher precision for certain bus locations, but the fact that the k-NN model had lower AUC and Testing Accuracy scores than the EGNN model may suggest that the EGNN generally did better at making classifications that were considerably more sensitive and specific for a larger majority of fault locations in the 123 bus power system.

To show model classification performance in more detail, Figures 6 - 9 show the resulting confusion matrices for each comparing model working with the IEEE 34 bus system. Figure 6 shows the EGNN model results, Figure 7 shows the k-NN model results, Figure 8 shows the MLP model results, and Figure 9 shows the Naive Bayes model results. As indicated by Table 1, we can see from Figures 6 - 9 that the EGNN saw the lowest amount of misclassification across the largest number of fault location classes.

Examining the confusion matrices for each model applied to the 342 LVN system, shown in Figures 10-13, reveals several noteworthy performance details. The 342 LVN bus system comprises primary and secondary bus connections, with an imbalanced class distribution for fault event locations. Fault locations were labeled such that primary bus locations received labels ranging roughly from 0 to 80, with class 14 representing the disproportionately large group. From the confusion matrices, it is evident that the EGNN significantly outperforms the other models in predicting fault locations. In contrast, the other models struggle with classification challenges posed by the larger class 14 and the imbalanced labeling of primary bus locations. Specifically, the kNN and MLP models (Figures 11 and 12, respectively) exhibit high

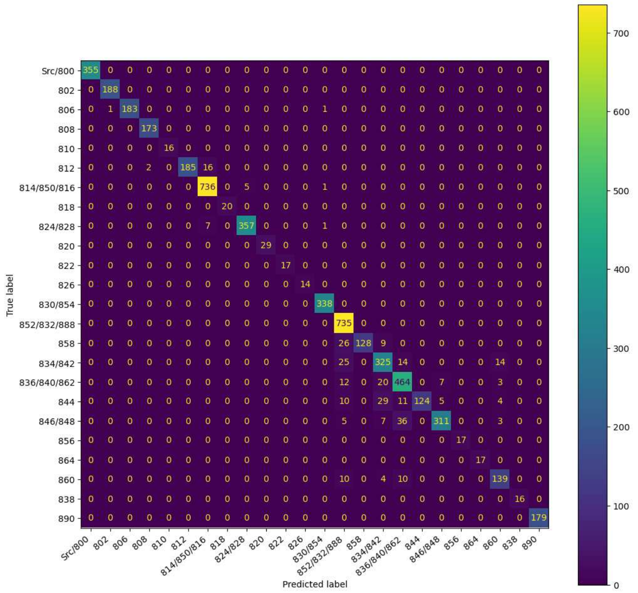

FIGURE 8. Confusion matrix showing MLP classification performance for IEEE 34 bus system.

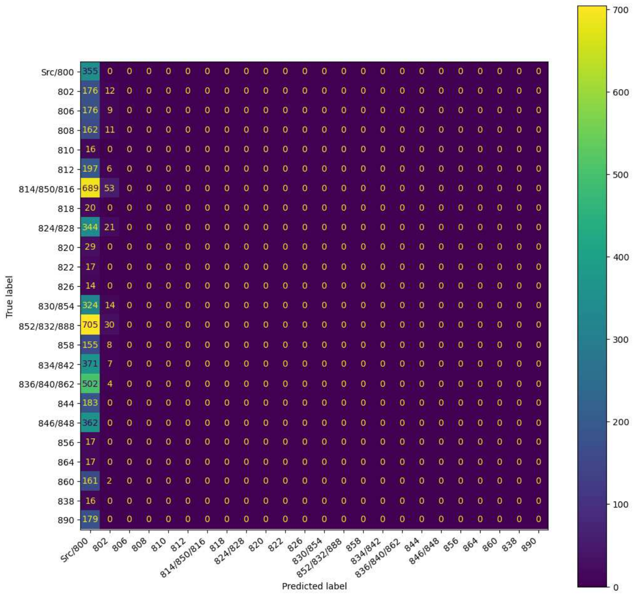

FIGURE 9. Confusion matrix showing Naive Bayes classification performance for IEEE 34 bus system.

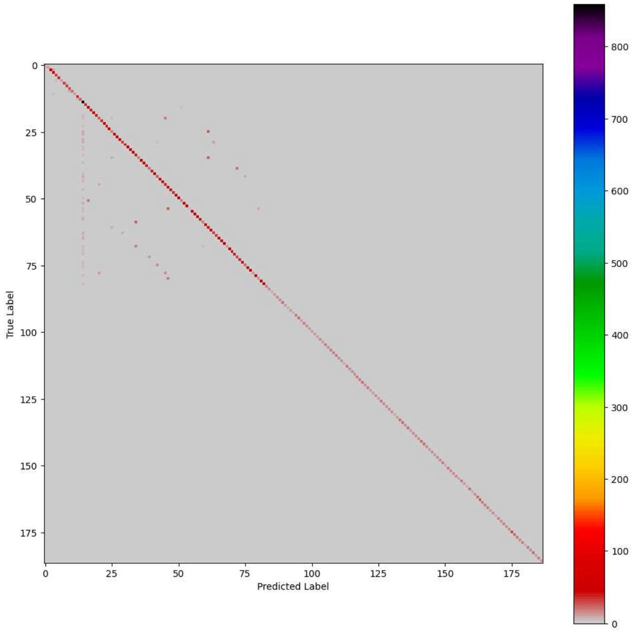

FIGURE 10. Confusion matrix showing EGNN classification performance for the 342 LVN bus system.

false positive and false negative rates for class 14. The Naive Bayes model, consistently the worst performer, demonstrates the most severe inability to predict fault locations within the primary distribution bus locations, as shown in Figure 13.

C. EXPLANATION METHOD COMPARATIVE STUDY

We exam our proposed explanation evaluation framework in this section. The implementation environment is in the Pytorch Geometric [41] and Captum [54]. In addition to several gradient-based explanation methods, including Guided Backpropagation [55], Integrated Gradients [56], Saliency [57], Deconvolution [58], and Input X Gradient [59], our comparative study of explanation methods also included the perturbation-based explanation methods GNNExplainer [40] and GraphMaskExplainer [60]. Whereas gradient-based methods attempt to estimate importance values of input features based directly on training gradient values, perturbation-based methods attempt to show what

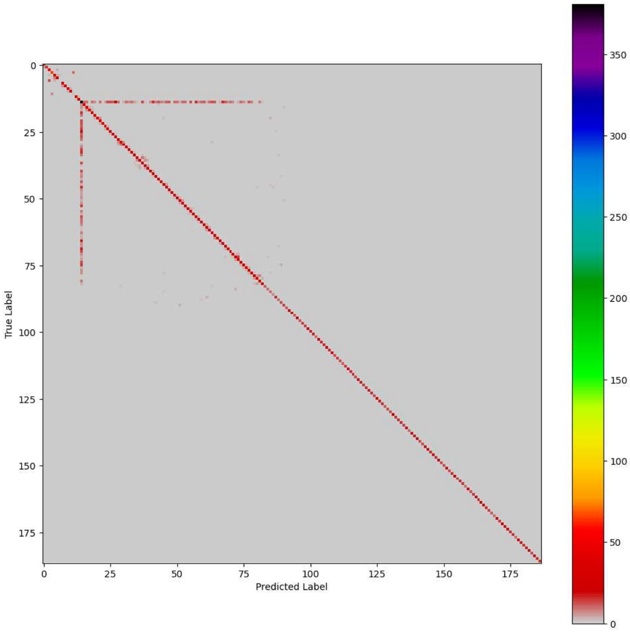

FIGURE 11. Confusion matrix showing kNN classification performance for the 342 LVN bus system.

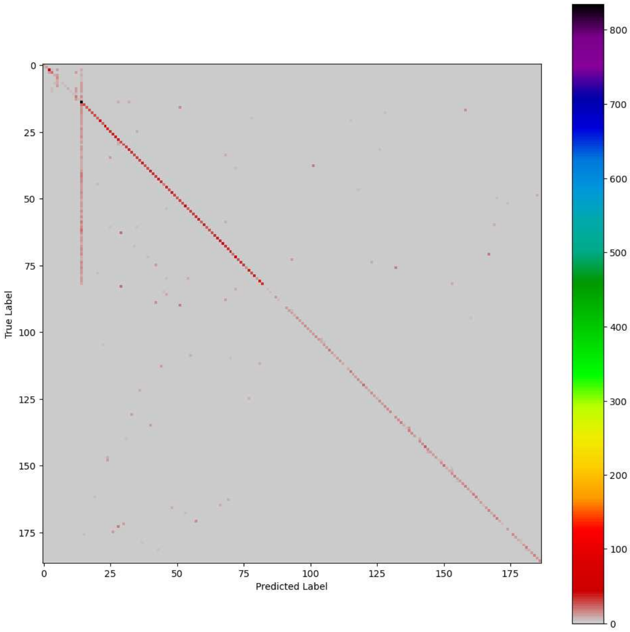

FIGURE 12. Confusion matrix showing MLP classification performance for the 342 LVN bus system.

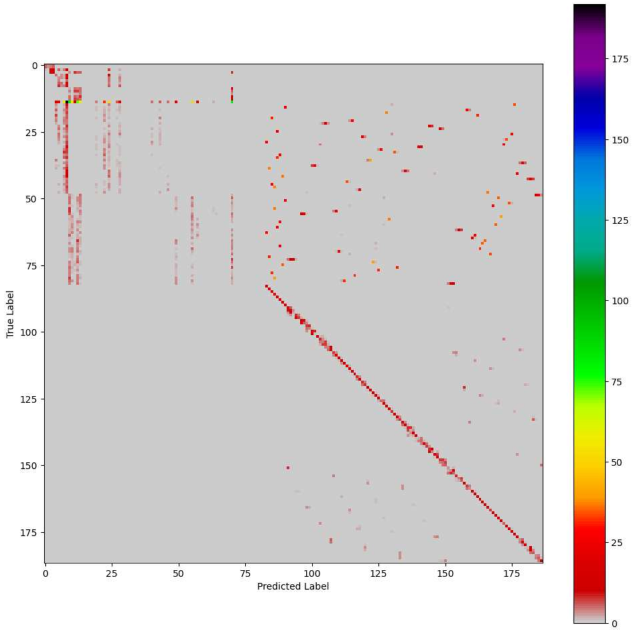

FIGURE 13. Confusion matrix showing Naive Bayes classification performance for the 342 LVN bus system.

TABLE 4. Explanation method comparative study for IEEE 34.

Methods	Fidelity+	Fidelity-	Characterization
GraphMask Explainer	0.8985	0.0694	0.1287
GNNExplainer	0.9048	0.0731	0.1351
Random	0.9377	0.0664	0.1239
Deconvolution	0.9193	0.0534	0.1008
Guided Backpropagation	0.9223	0.0454	0.0863
InputXGradient	0.9574	0.0519	0.0984
Integrated Gradients	**0.9696**	**0.0733**	**0.1361**
Saliency	0.9418	0.0603	0.1132

input features are most important by analyzing how model predictions are influenced when random modifications are applied to input features [33].

1) PARAMETERS AND APPROACH FOR EGNN EVALUATION

The Fidelity+, Fidelity-, and Characterization evaluation metrics are reported at TopK values of 10 to comprehensively compare the performance of these evaluation methods for explanation subgraphs with 10 edges. These results are tested using hard masks to identify the most effective methods. As a baseline, the same mask conditions and evaluation metrics are also applied to a "Random" method, which generates random binary values (0 and 1) for each hard mask element, enabling a comparison of all explanation methods against random results. Since random guesses are unlikely to provide meaningful model insights, any explanation method performing no better than Random will be deemed largely ineffective. The evaluation metrics reported in the following sections were tested using 6 different random samples, each containing 1,000 instances of graph input data. As noted earlier, Fidelity- scores are presented as 1 - Fidelity-.

TABLE 5. Explanation method comparative study for IEEE 123.

Methods	Fidelity+	Fidelity-	Characterization
GraphMask Explainer	0.9896	0.0162	0.0318
GNNExplainer	0.7522	0.0132	0.0257
Random	0.8685	0.0148	0.0290
Deconvolution	0.9057	0.0073	0.0143
Guided Backpropagation	0.9013	0.0084	0.0165
InputXGradient	0.9489	0.0100	0.0197
Integrated Gradients	**0.9948**	**0.0164**	**0.0322**
Saliency	0.9378	0.0066	0.0130

TABLE 6. Explanation method comparative study for 342 LVN.

Methods	Fidelity+	Fidelity-	Characterization
GraphMask Explainer	**0.9933**	0.0067	0.0132
GNNExplainer	0.8393	0.0067	0.0131
Random	0.8381	0.0069	0.0135
Deconvolution	0.8370	**0.0079**	**0.0157**
Guided Backpropagation	0.8411	0.0065	0.0127
InputXGradient	0.8373	0.0071	0.0139
Integrated Gradients	0.8797	0.0074	0.0145
Saliency	0.8301	0.0075	0.0149

2) EXPLANATION METHOD EVALUATION FRAMEWORK RESULTS

Based on the explanation method performance metric results shown in Tables 4, 5, and 6, which show the Fidelity+, Fidelity-, and Characterization scores resulting from the explanation subgraphs generated by each explanation method, it was generally found that overall explanation method performance seemed to be noticeably lower for the IEEE 123 and 342 LVN bus systems when compared to the explanation performance observed for the IEEE 34 bus system, particularly when taking Fidelity- and Characterization scores into account. This means that the explanation methods were generally less effective at generating sufficient explanations for the IEEE 123 and 342 LVN bus systems. This may be attributed to the greater complexity of larger systems, as more complex bus systems inherently have a higher number of classes due to the increased number of possible fault locations. Nonetheless, from a model performance and fault location detection perspective, our EGNN demonstrated strong scalability with these larger and more intricate bus systems.

For both the IEEE 34 and IEEE 123 bus systems, the Integrated Gradients method demonstrates the best-performing explanation results. For the IEEE 34 bus system, while GraphMask Explainer and GNNExplainer—both perturbation-based methods—generally outperformed most other gradient-based methods aside from Integrated Gradients in terms of Fidelity- (indicating how often an explanation method successfully identifies input features sufficient to reproduce the model's prediction, such as fault location predictions made by the EGNN), these perturbation-based methods showed the lowest Fidelity+ scores. This suggests that, for this system, perturbation-based methods struggle to generate explanations capturing the portions of the input graph uniquely necessary for the model's prediction.

TABLE 7. Calculation times for explanation methods.

Methods	Calculation Time (Seconds)
GraphMask Explainer	19.8
GNNExplainer	3.41
Random	0.046
Deconvolution	0.076
Guided Backpropagation	0.076
InputXGradient	0.063
Integrated Gradients	0.871
Saliency	0.075

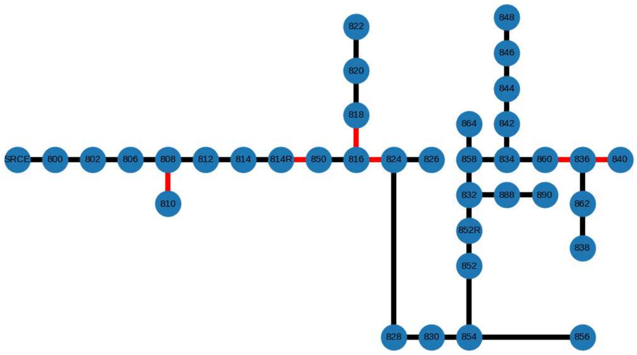

FIGURE 14. Integrated gradients edge explanation for fault at 802.

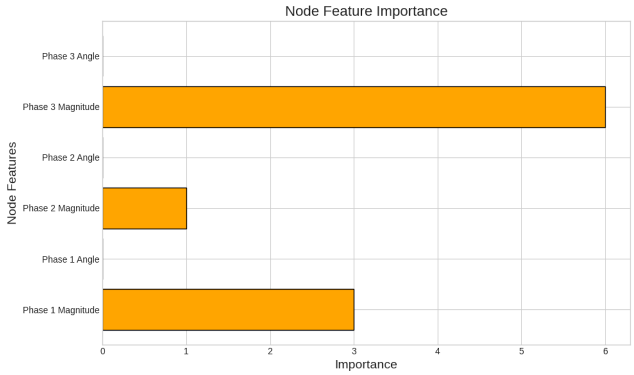

FIGURE 15. Integrated gradients node feature explanation.

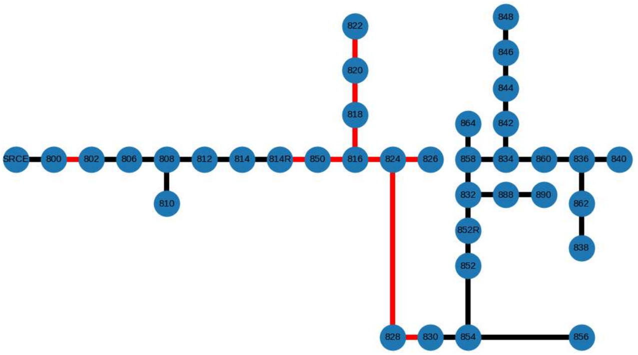

FIGURE 16. GraphMask explainer edge explanation for fault at 830/854.

The generally poorer performance of explanation methods for the more complex IEEE 123 and 342 LVN bus systems suggests these methods may work best on simpler power grid sections or grids labeled with fewer fault location classes. Even for the simpler IEEE 34 bus system, most gradient-based methods (except Integrated Gradients) performed no better than random guesses. These findings highlight the importance of the EGNN Evaluation Framework for fault detection, as practitioners need to identify methods that may struggle to explain EGNN predictions. Among the methods studied, GNNExplainer, GraphMask Explainer, and especially Integrated Gradients showed better and more consistent performance for the IEEE 34 and 123 bus systems.

We have also reported the computation times for each explanation method in Table 7, measured in seconds. Each value represents the average of five sample runs using identical hardware. As expected, the Random method had the shortest computation time since it simply generates random numbers. Among the remaining methods, most gradient-based approaches were the fastest, with computation times for Integrated Gradients, Guided Backpropagation, Deconvolution, Saliency, and InputXGradient ranging from slowest to fastest within this group.

D. USAGE AND DEMONSTRATION OF EXPLANATION

The EGNN explanation methods are designed to clarify the GNN model's logic by visualizing the spatial and temporal features present in power grid data. Spatial aspects highlight the significance of line connections in the grid, represented by the explanation edge mask, while temporal aspects indicate the importance of node features, such as voltage angles and magnitudes, at specific time frames, shown in the explanation node mask. These explanation subgraphs are fully utilized in the visualizations. To implement these methods, we used PyTorch Geometric, Captum, NetworkX, and Matplotlib, as mentioned earlier. The visualizations from Integrated Gradients are shown in Figures 14 and 15, those from GraphMask Explainer in Figures 16 and 17, and those from GNNExplainer in Figures 18 and 19.

1) EXPLAINS SPATIAL ASPECTS

As discussed earlier, the data from each power system case study represent 1-phase, 2-phase, and 3-phase connections between bus locations. Since GNN models can leverage graph topology, the explanation methods for our EGNN are able to identify and express the importance of different parts of this topology. This means that the edge masks generated by each explanation method highlight the most significant line connections between bus locations, which the EGNN model uses to make fault location predictions. Figures 14, 16, and 18 demonstrate how these spatial aspects can be visualized for the IEEE 34 bus system using NetworkX and Matplotlib. In these visualizations, the edge masks highlight line connections in red, indicating their importance to the model's prediction, while black lines are considered less important. The red lines are part of the explanation subgraph, representing the key connections for the specific fault

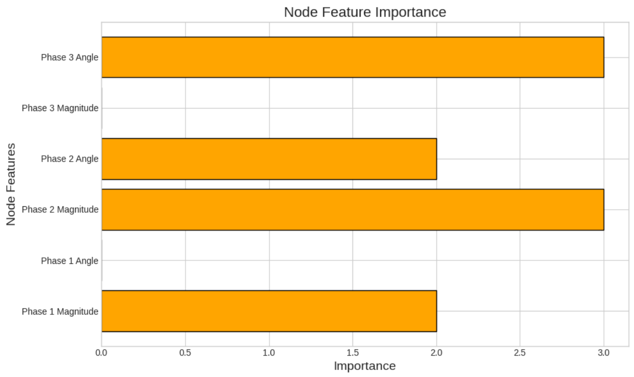

FIGURE 17. GraphMask explainer node feature explanation.

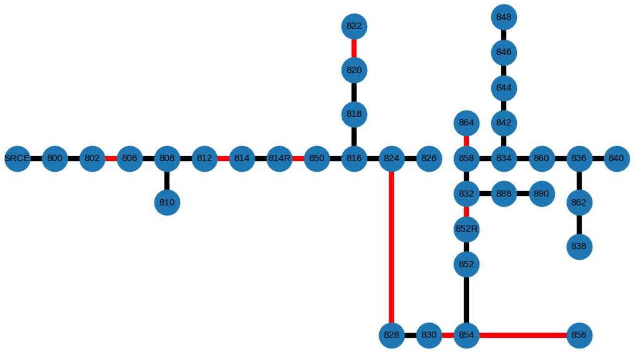

FIGURE 18. GNNExplainer edge explanation for fault at 830/854.

scenario, while the black lines belong to a complementary subgraph, which is less relevant to the explanation.

2) EXPLAINS TEMPORAL ASPECTS

The temporal aspect of the EGNN is determined by the importance of node features, specifically the voltage phasor data measured at each bus location, as emphasized by the node mask in the explanation subgraph. For the IEEE 34 bus system, this temporal aspect directly reflects the voltage angle and magnitude data for each of the three phases measured at each bus. Figures 15, 17, and 19 illustrate how these temporal aspects are visualized for the IEEE 34 bus system using Matplotlib. These visualizations, which complement the edge explanations shown on the same page, highlight the node features deemed most important by the corresponding EGNN explanation method. In the context of power grids, each bus location is treated as a node, and these node features represent the voltage data recorded by PMUs at each location. The importance values for each of the voltage magnitude and angle readings, shown in the figures, reflect how frequently these features were selected by the node mask as crucial for a specific fault scenario. Bar charts are used for readability, aligning with the edge explanation visuals on the same page.

3) DISCUSSION OF EGNN EXPLANATION VISUALIZATIONS

The visualizations demonstrate how fault location predictions made by the EGNN model can be explained in a comprehensive spatiotemporal context. For example, Figures 14 and 15 illustrate the results from Integrated Gradients, which explain what features of the IEEE 34 bus system the EGNN model relied on most when predicting a fault at bus location 802. In Figure 14, the red lines represent the bus connections that Integrated Gradients identified as most significant for the fault location prediction. Figure 15 shows that, for this particular fault event, Integrated Gradients emphasized the voltage magnitude data from all three phases as being more important than the voltage angles. Notably, the voltage magnitudes at Phase 3 had the most influence on the model's decision, as indicated by the higher importance scores for that phase compared to the others.

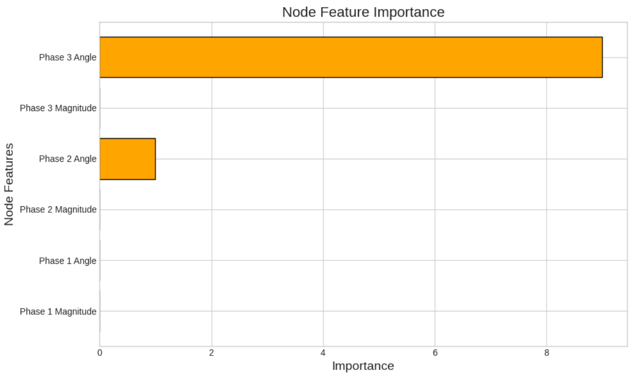

FIGURE 19. GNNExplainer node feature explanation.

The explanations offer a way to interpret the underlying logic of the EGNN model, providing greater transparency by revealing which aspects of the power grid system data the model considers most crucial when making predictions. Similarly, Figures 16-19 showcase the explanations from GraphMask Explainer and GNNExplainer, demonstrating how these methods highlight different features of the power grid data and contribute to a better understanding of the model's decision-making process for various fault scenarios. Ultimately, these visualizations help improve transparency and interpretability of the EGNN model in the context of power grid fault location detection.

V. CONCLUSION

Short-circuit fault events present significant challenges for utility providers and power grid operators. To reduce the time required for fault detection, research suggests leveraging machine learning models that utilize voltage data across the grid. While GNN models excel at learning from the network topology of power grids to make accurate predictions, their lack of transparency can hinder trust in their results. To address this issue, we introduce the Explainable GNN (EGNN), which incorporates explanation techniques into GNN models for fault location detection, clarifying the rationale behind their predictions.

Furthermore, we present an EGNN Evaluation Framework to systematically evaluate the efficacy of multiple explanation approaches. This comprehensive evaluation identifies the most effective techniques for various power grid configurations. Additionally, we showcase how these methods generate clear, graphic, and context-rich explanations that reveal the EGNN model's underlying logic. These insights not only enhance the interpretability of GNN-based fault location systems but also bolster their trustworthiness, paving the way for more transparent and reliable power grid operation.

REFERENCES

[1] M. Mousa, S. Abdelwahed, and J. Klüss, "Review of diverse types of fault, their impacts, and their solutions in smart grid," in *Proc. SoutheastCon*, 2019, pp. 1–7.

[2] J. D. Glover, T. J. Overbye, and M. S. Sarma, *Power System Analysis & Design*. Boston, MA, USA: Cengage Learning, 2011.

[3] S. Motta, J. Ihonen, and J. Kiviluoma, "A new method for analysing financial damages caused by grid faults on individual customers," *Electr. Power Syst. Res.*, vol. 207, Jun. 2022, Art. no. 107839.

[4] M. Shafiullah and M. A. Abido, "A review on distribution grid fault location techniques," *Electr. Power Compon. Syst.*, vol. 45, no. 8, pp. 807–824, May 2017.

[5] S. Das, S. Santoso, A. Gaikwad, and M. Patel, "Impedance-based fault location in transmission networks: Theory and application," *IEEE Access*, vol. 2, pp. 537–557, 2014.

[6] M. Izadi and H. Mohsenian-Rad, "Synchronous waveform measurements to locate transient events and incipient faults in power distribution networks," *IEEE Trans. Smart Grid*, vol. 12, no. 5, pp. 4295–4307, Sep. 2021.

[7] A. Ghaderi, H. A. Mohammadpour, H. L. Ginn, and Y.-J. Shin, "High-impedance fault detection in the distribution network using the time-frequency-based algorithm," *IEEE Trans. Power Del.*, vol. 30, no. 3, pp. 1260–1268, Jun. 2015.

[8] Y. Wang, C. Chen, J. Wang, and R. Baldick, "Research on resilience of power systems under natural disasters—A review," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1604–1613, Mar. 2016.

[9] H. Jiang, J. J. Zhang, W. Gao, and Z. Wu, "Fault detection, identification, and location in smart grid based on data-driven computational methods," *IEEE Trans. Smart Grid*, vol. 5, no. 6, pp. 2947–2956, Nov. 2014.

[10] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI open*, vol. 1, pp. 57–81, Jan. 2020.

[11] K. Amara, R. Ying, Z. Zhang, Z. Han, Y. Shan, U. Brandes, S. Schemm, and C. Zhang, "GraphFramEx: Towards systematic evaluation of explainability methods for graph neural networks," 2022, *arXiv:2206.09677*.

[12] R. G. Bosso, "Explainable graph learning for power grid fault detection," Master's thesis, Florida Atlantic University, Boca Raton, FL, USA, 2024.

[13] O. A. Alimi, K. Ouahada, and A. M. Abu-Mahfouz, "A review of machine learning approaches to power system security and stability," *IEEE Access*, vol. 8, pp. 113512–113531, 2020.

[14] M. S. Ibrahim, W. Dong, and Q. Yang, "Machine learning driven smart electric power systems: Current trends and new perspectives," *Appl. Energy*, vol. 272, Aug. 2020, Art. no. 115237.

[15] S. R. Fahim, S. K. Sarker, S. M. Muyeen, S. K. Das, and I. Kamwa, "A deep learning based intelligent approach in detection and classification of transmission line faults," *Int. J. Electr. Power Energy Syst.*, vol. 133, Dec. 2021, Art. no. 107102.

[16] J. Hosseinzadeh, F. Masoodzadeh, and E. Roshandel, "Fault detection and classification in smart grids using augmented K-NN algorithm," *Social Netw. Appl. Sci.*, vol. 1, no. 12, p. 1627, Dec. 2019.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[18] A. G. Rameshrao, E. Koley, and S. Ghosh, "A LSTM-based approach for detection of high impedance faults in hybrid microgrid with immunity against weather intermittency and N-1 contingency," *Renew. Energy*, vol. 198, pp. 75–90, Oct. 2022.

[19] M. Hossain, R. Khan, N. Islam, S. Sarker, S. Fahim, and S. Das, "Deep learning techniques for transmission line fault diagnosis: A comparative evaluation," in *Proc. Int. Conf. Autom., Control Mechatronics Ind. 4.0 (ACMI)*, Jul. 2021, pp. 1–5.

[20] W. Liao, B. Bak-Jensen, J. R. Pillai, Y. Wang, and Y. Wang, "A review of graph neural networks and their applications in power systems," *J. Mod. Power Syst. Clean Energy*, vol. 10, no. 2, pp. 345–360, Mar. 2022.

[21] M. MansourLakouraj, R. Hossain, H. Livani, and M. Ben-Idris, "Application of graph neural network for fault location in PV penetrated distribution grids," in *Proc. North Amer. Power Symp. (NAPS)*, Nov. 2021, pp. 1–6.

[22] H. Tong, R. C. Qiu, D. Zhang, H. Yang, Q. Ding, and X. Shi, "Detection and classification of transmission line transient faults based on graph convolutional neural network," *CSEE J. Power Energy Syst.*, vol. 7, no. 3, pp. 456–471, May 2021.

[23] D. Chanda and N. Yahya Soltani, "A heterogeneous graph-based multi-task learning for fault event diagnosis in smart grid," 2023, *arXiv:2309.09921*.

[24] A. Varbella, K. Amara, B. Gjorgiev, M. El-Assady, and G. Sansavini, "PowerGraph: A power grid benchmark dataset for graph neural networks," 2024, *arXiv:2402.02827*.

[25] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.

[26] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.

[27] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[28] K. Chen, J. Hu, Y. Zhang, Z. Yu, and J. He, "Fault location in power distribution systems via deep graph convolutional networks," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 1, pp. 119–131, Jan. 2020.

[29] A. Pandey and S. R. Mohanty, "Graph convolutional network based fault detection and identification for low-voltage DC microgrid," *J. Mod. Power Syst. Clean Energy*, vol. 11, no. 3, pp. 917–926, 2023.

[30] P. Veli ković, G. Cucurull, A. Casanova, A. Romero, P. Lió, and Y. Bengio, "Graph attention networks," *stat*, pp. 1–12, 2017.

[31] W. Hu, B. Liu, J. Gomes, M. Zitnik, P. Liang, V. Pande, and J. Leskovec, "Strategies for pre-training graph neural networks," 2019, *arXiv:1905.12265*.

[32] Y. Shi, Z. Huang, S. Feng, H. Zhong, W. Wang, and Y. Sun, "Masked label prediction: Unified message passing model for semi-supervised classification," 2020, *arXiv:2009.03509*.

[33] H. Yuan, H. Yu, S. Gui, and S. Ji, "Explainability in graph neural networks: A taxonomic survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 5, pp. 5782–5799, May 2023.

[34] N. Burkart and M. F. Huber, "A survey on the explainability of supervised machine learning," *J. Artif. Intell. Res.*, vol. 70, pp. 245–317, Jan. 2021.

[35] R. Machlev, L. Heistrene, M. Perl, K. Y. Levy, J. Belikov, S. Mannor, and Y. Levron, "Explainable artificial intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities," *Energy AI*, vol. 9, Aug. 2022, Art. no. 100169.

[36] C. Xu, Z. Liao, C. Li, X. Zhou, and R. Xie, "Review on interpretable machine learning in smart grid," *Energies*, vol. 15, no. 12, p. 4427, Jun. 2022.

[37] Y. Liu, N. Zhang, D. Wu, A. Botterud, R. Yao, and C. Kang, "Searching for critical power system cascading failures with graph convolutional network," *IEEE Trans. Control Netw. Syst.*, vol. 8, no. 3, pp. 1304–1313, Sep. 2021.

[38] A. Verdone, S. Scardapane, and M. Panella, "Explainable spatio-temporal graph neural networks for multi-site photovoltaic energy production," *Appl. Energy*, vol. 353, Jan. 2024, Art. no. 122151.

[39] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS ONE*, vol. 10, no. 7, Jul. 2015, Art. no. e0130140.

[40] R. Ying, D. Bourgeois, J. You, M. itnik, and J. Leskovec, "GNNExplainer: Generating explanations for graph neural networks," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, pp. 9244–9255.

[41] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch geometric," in *Proc. ICLR Workshop Represent. Learn. Graphs Manifolds*, 2019.

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[43] W. H. Kersting, "Radial distribution test feeders," *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 975–985, Aug. 1991.

[44] K. P. Schneider, B. A. Mather, B. C. Pal, C.-W. Ten, G. J. Shirek, H. Zhu, J. C. Fuller, J. L. R. Pereira, L. F. Ochoa, L. R. de Araujo, R. C. Dugan, S. Matthias, S. Paudyal, T. E. McDermott, and W. Kersting, "Analytic considerations and design basis for the IEEE distribution test feeders," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3181–3188, May 2018.

[45] K. Schneider, P. Phanivong, and J.-S. Lacroix, "IEEE 342-node low voltage networked test system," in *Proc. IEEE PES Gen. Meeting Conf. Expo.*, Jul. 2014, pp. 1–5.

[46] R. C. Dugan and T. E. McDermott, "An open source platform for collaborating on smart grid research," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, Jul. 2011, pp. 1–7.

[47] P. Radatz, E. Viana, and P. Londero. (2024). *Py-DSS-Interface*. [Online]. Available: https://pypi.org/project/py-dss-interface/

[48] N. Bhusal, R. M. Shukla, M. Gautam, M. Benidris, and S. Sengupta, "Deep ensemble learning-based approach to real-time power system state estimation," *Int. J. Electr. Power Energy Syst.*, vol. 129, Jul. 2021, Art. no. 106806.

[49] J. Petinrin, J. Agbolade, and O. O. Petinrin, "Application of open distribution system simulator (opendss) in distribution feeders with renewable energy," *Technol. (IJOSEET)*, vol. 4, no. 5, pp. 37–47, 2019.

[50] L. J. R. Neiva, F. C. R. Coelho, W. Peres, S. A. Flávio, and L. R. Dias, "Analysis of power flow reversion in distribution transformers due to medium-voltage fault and distributed generation in secondary networks," *J. Control, Autom. Electr. Syst.*, vol. 32, no. 6, pp. 1718–1727, Dec. 2021.

[51] Y. Chen, R. A. Jacob, Y. R. Gel, J. Zhang, and H. V. Poor, "Learning power grid outages with higher-order topological neural networks," *IEEE Trans. Power Syst.*, vol. 39, no. 1, pp. 720–732, Jan. 2024.

[52] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 52, pp. 2825–2830, Jan. 2011.

[53] P. Thölke, Y.-J. Mantilla-Ramos, H. Abdelhedi, C. Maschke, A. Dehgan, Y. Harel, A. Kemtur, L. M. Berrada, M. Sahraoui, T. Young, A. B. Pépin, C. El Khantour, M. Landry, A. Pascarella, V. Hadid, E. Combrisson, J. O'Byrne, and K. Jerbi, "Class imbalance should not throw you off balance: Choosing the right classifiers and performance metrics for brain decoding with imbalanced data," *NeuroImage*, vol. 277, Aug. 2023, Art. no. 120253.

[54] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, "Captum: A unified and generic model interpretability library for pytorch," 2020, *arXiv:2009.07896*.

[55] J. Tobias Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, "Striving for simplicity: The all convolutional net," 2014, *arXiv:1412.6806*.

[56] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3319–3328.

[57] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," 2013, *arXiv:1312.6034*.

[58] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. ECCV*, 2014, pp. 818–833.

[59] A. Shrikumar, P. Greenside, A. Shcherbina, and A. Kundaje, "Not just a black box: Learning important features through propagating activation differences," 2016, *arXiv:1605.01713*.

[60] M. S. Schlichtkrull, N. De Cao, and I. Titov, "Interpreting graph neural networks for NLP with differentiable edge masking," 2020, *arXiv:2010.00577*.

RICHARD BOSSO received the B.S. degree in biology from the University of Central Florida, in 2018, and the M.S. degree in data science and analytics from Florida Atlantic University, in 2024. His work focuses on applying machine learning and data-driven techniques to smart grids and energy systems, leveraging both research and industry experience in data engineering, predictive analytics, and intelligent energy management. His research interests include developing explainable machine learning models, optimizing data pipelines, and enhancing power grid resilience through advanced data analytics.

COREY CHANG received the bachelor's degree in electrical engineering from Florida Atlantic University. He is currently pursuing the master's degree in electrical engineering. From 2023 to 2024, he conducted undergraduate research on smart grids with Florida Power and Light (FPL) Center for Intelligent Energy Technologies (InETech), where he explored advanced control strategies and grid integration techniques for modern power systems. His professional research interests include power electronics, renewable energy, and the development of efficient and sustainable energy conversion technologies.

MAHDI ZARIF received the Ph.D. degree in electrical engineering from Ferdowsi University of Mashhad, in 2012. He is currently a Postdoctoral Researcher with the Institute for Sensing and Embedded Network Systems Engineering (I-SENSE), Florida Atlantic University, working on smart grids and smart cities. Previously, he was a Postdoctoral Associate with the University of Miami, focusing on energy analytics and sustainability, for two years. His research interests include power system analysis, renewable energy, smart grids, data analysis, deep learning, and energy management.

YUFEI TANG (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Rhode Island, Kingston, RI, USA, in 2016.

He is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, and a Faculty Fellow with the Institute for Sensing and Embedded Network Systems Engineering, Florida Atlantic University, Boca Raton, FL, USA. His research interests include machine learning, smart grid, and renewable energy. He was a recipient of the IEEE International Conference on Communications Best Paper Award, in 2014; the National Academies Gulf Research Program Early-Career Research Fellowship, in 2019; and the National Science Foundation CAREER Award, in 2022. He is an Associate Editor for IEEE Journal of Oceanic Engineering.