Future Energy
Open Access Journal
ISSN 2832-0328

August 2025| Volume 04 | Issue 03 | Pages 35-47

Journal homepage: https://fupubco.com/fuen

https://doi.org/10.55670/fpll.fuen.4.3.4

Future Publishing LLC

Article

Transfer learning for power system fault location using artificial neural networks

Stefanos Petridis[1], Petros Iliadis[1], Angelos Saverios Skembris[2], Rakopoulos Dimitrios[2*], Elias Kosmatopoulos[1]

[1]Department of Electrical and Computer Engineering, Democritus University of Thrace, Xanthi, Greece
[2]Sustenergo CERTH Spin-off P.C., Kozani, Greece

ARTICLE INFO

Article history:
Received 26 May 2025
Received in revised form
03 July 2025
Accepted 12 July 2025

Keywords:
Artificial neural networks, Fault location, Transfer learning, Power distribution systems, IEEE test feeders, Classification

*Corresponding author
Email address:
rakopoulos@certh.gr

DOI: 10.55670/fpll.fuen.4.3.4

ABSTRACT

This paper investigates the application of transfer learning techniques to artificial neural networks (ANNs) for fault detection in power distribution systems, formulated as a classification problem. Comprehensive datasets are developed using multiple IEEE test feeders of varying complexity, including the 13-bus, 34-bus, 37-bus, and 123-bus test feeders. Various fault types are simulated across all three-phase buses in each system. Baseline performance is established by independently training ANNs on each feeder. Subsequently, knowledge learned from the complex 123-bus feeder is transferred to accelerate and improve fault location in simpler networks. The results demonstrate that transfer learning significantly improves both training efficiency and classification performance. Training convergence is accelerated by a factor of 1.68 to 2.56 across target feeders, corresponding to epoch reductions between 40.6% and 61.0%. Additionally, computational time is reduced by 24.0% to 49.5%, further enhancing the practical viability of the proposed approach. These findings suggest that transfer learning offers a powerful strategy to address data scarcity and computational challenges in fault location, enabling utilities to deploy accurate, efficient fault detection systems across diverse distribution networks with minimal retraining effort.

1. Introduction

The rapid identification and localization of faults in power distribution systems is crucial for minimizing outage durations, optimizing repair crew dispatch, and improving system reliability indices. Traditional fault location methods often rely on impedance-based calculations or rule-based systems that may struggle with the increasing complexity and dynamic nature of modern distribution networks. Machine learning techniques, particularly artificial neural networks (ANNs), have shown promising results in addressing these challenges by learning patterns from system measurements to accurately classify fault locations [1]. ANNs are capable of learning complex relationships between electrical measurements and fault parameters, offering better accuracy and noise tolerance than traditional impedance-based or traveling-wave techniques [2]. A significant number of recent works have focused specifically on ANN-based approaches. For example, Pourahmadi-Nakhli and Safavi [3] employed ANNs combined with wavelet features for path characteristic frequency-based fault location. Rafinia and Moshtagh [4] used ANNs and fuzzy logic to locate faults in underground systems based on high-frequency features extracted from current and voltage signals. Dashtdar et al. [5] developed an ANN-based system for fault section identification and localization in distribution networks. Similarly, Aslan and Yagan [6] used ANN models to classify fault types and distances based on the frequency spectra of fault signals. Other studies have enhanced ANN-based models to address the challenges posed by distributed generation. Javadian et al. [7] combined ANN with distributed generation information to determine fault types and locations. Bakkar et al. [8] presented an ANN-based protection framework for smart grids. Shafiullah et al. [9] introduced a wavelet-based extreme learning machine, an ANN variant, for fault location. Lout and Aggarwal [10] proposed an ANN-based method that uses current transients for fault phase selection and location in active networks with spurs. While ANN-based methods have proven effective under known network configurations, they typically require extensive retraining when network topology or operating conditions change, limiting their practical deployment in evolving smart grids. Traditional ANN approaches for fault location typically train separate models for each distribution system [1], ignoring potential knowledge transfer between networks of varying complexity.

Abbreviations

ANN	Artificial Neural Network
PG	Phase-to-Ground
DPG	Double-Phase-to-Ground
IEEE	Institute of Electrical and Electronics Engineers
MLP	Multi-Layer Perceptron
PP	Phase-to-Phase
ReLU	Rectified Linear Unit
SGD	Stochastic Gradient Descent
SPG	Single-Phase-to-Ground

Nomenclature

b	Bias term
C	Number of classes E - Error function
f	Activation function or classifier
I	Current magnitude (A)
N	Number of samples
n	Number of neurons
P	Active power (W)
Q	Reactive power (VAr)
V	Voltage magnitude (V)
wi	Weight of i-th neuron
xi	Input value
y	Output
Z	Impedance (Ω)
η	Learning rate
θ	Phase angle (rad)
μ	Mean
σ	Standard deviation

This methodology becomes increasingly inefficient as utilities manage diverse feeder configurations ranging from simple radial systems to complex meshed networks with distributed generation [11]. Therefore, despite their effectiveness, the deployment of ANNs in real-world settings is hindered by two significant limitations. First, deep learning models typically require large amounts of labelled training data—an unrealistic expectation in power systems where fault events are both rare and sparsely labelled [12]. Second, these models often lack the flexibility to generalize across varying network topologies [13]. In practical utility applications, this poses a major barrier: acquiring sufficient fault data for every unique or reconfigured network is rarely feasible. As a result, while ANNs offer a compelling solution in theory, their practical implementation remains constrained by data availability and transferability challenges.

Transfer learning offers a promising solution to this challenge by enabling models trained under one set of conditions to adapt to new scenarios with minimal additional data. This approach aligns with the intuition that the underlying physical phenomena of fault behavior share common characteristics across different network topologies, despite variations in specific electrical parameters and configurations. Recent studies have successfully applied transfer learning to fault detection tasks: Shakiba et al. [14] used CNNs for transmission line fault diagnosis under varying line characteristics; Yao et al. [15] applied deep transfer learning to improve fault detection across different nuclear power plant conditions; and Asutkar and Tallur [16] demonstrated efficient domain adaptation in machine fault diagnosis. These works show that transfer learning can significantly reduce training costs, improve generalization, and accelerate model deployment, making it a highly attractive tool for fault detection in dynamic and distributed energy systems. However, few studies have yet explored its potential specifically for distribution networks with high variability — a gap this paper aims to address. The remainder of this paper is organized as follows: In Section 2, the Fault Location problem is described and formulated. A brief description of neural networks is also provided, along with their use in similar tasks. In Section 3, the transfer learning approach is presented. In Section 4, the methodology for creating comprehensive fault datasets across multiple test feeders is described in detail. The data manipulation, the ANN architectures used, and the training procedures for establishing baseline performance as well as for transfer learning are presented in Section 5. The experimental results are analyzed in Section 6, and the baseline and transfer learning performance are compared. The findings of the experiments are summarized in Section 7, and their implications for practical applications are discussed.

2. Fault location with neural networks

In this section, a brief description of the fault location problem is provided. Subsequently, a small introduction to ANNs is included, along with the way they can be applied to the fault location problem.

2.1 Fault location problem formulation

In this paper, it is assumed that voltage phasor measurements are available for all connected phases at all buses of the distribution network. That is, for a certain distribution network bus, *i*, a vector $[V_i^a, \theta_i^a, V_i^b, \theta_i^b, V_i^c, \theta_i^c]^T \in \mathbb{R}^6$ exists where:

- V_i^a, V_i^b and V_i^c are the voltage magnitudes of phases *a*, *b* and *c*, respectively, on bus *i*.
- θ_i^a, θ_i^b and θ_i^c are the voltage angles of phases *a*, *b* and *c*, respectively, on bus *i*.

For a distribution network containing *n* buses, a single data point, *k*, that includes all buses' measurements can be represented as a single vector, $\mathrm{x}_k \in \mathbb{R}^{6n}$. The entire dataset of all *m* data points measured for the examined distribution network can be represented as a single matrix $\mathrm{x}_k \in \mathbb{R}^{6n \times m}$. The fault location problem is, then, formulated as a classification task where the classifier's objective is to locate the faulty bus if one is present. That is, given a data point x_k, the faulty bus $\hat{y}_k$ is determined by the following:

$$\hat{y}_k = f(\mathrm{x}_k) \tag{1}$$

where f is the classification method.

2.2 Introduction to artificial neural networks

ANNs are computational models inspired by the structure and functioning of biological neural networks in the human brain. They have emerged as powerful machine learning tools for solving complex problems across various domains, including pattern recognition, function approximation, classification, and prediction tasks [17]. The key strength of neural networks lies in their ability to learn representations from data without explicit programming of rules, making them particularly effective for problems where

mathematical formulation is difficult or unknown. The fundamental idea behind neural networks is to simulate the behavior of interconnected neurons that process and transmit information. Similar to biological neural networks, artificial neural networks learn by adjusting the connections (weights) between artificial neurons in response to input stimuli, enabling them to recognize patterns and make predictions based on previously unseen data. An artificial neuron is called the perceptron, graphically depicted in Figure 1, and is the fundamental building block of neural networks. The mathematical representation of a perceptron of n inputs and one output, y, is the following:

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \quad (2)$$

where x_i represents the i^{th} input value, w_i the weight of the i^{th} neuron that is applied to the input x_i, b the bias term, and $f(\cdot)$ the activation function.

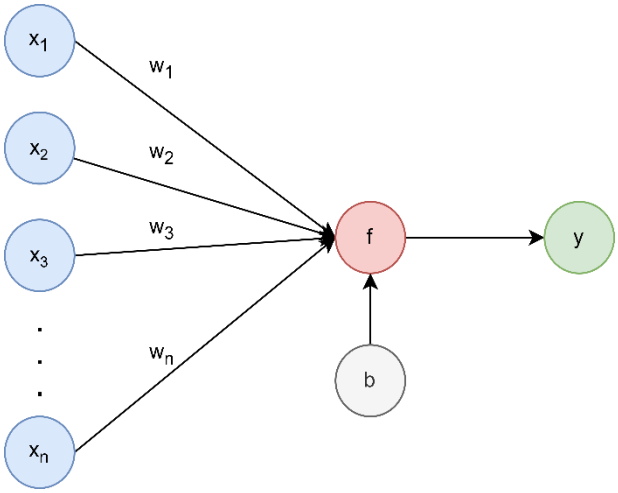


Figure 1: Schematic representation of a perceptron, showing inputs x_i, weights w_i, bias b, activation function f, and output y.

A multi-layer neural network, shown in Figure 2, also known as a multi-layer perceptron (MLP), consists of an input layer, one or more hidden layers, and an output layer. Each layer contains multiple neurons (perceptrons), with each neuron in a layer connected to all neurons in the subsequent layer, forming a fully connected structure. The key advantage of multi-layer neural networks is their ability to learn hierarchical representations. Lower layers typically learn to detect simple features, while higher layers combine these features to recognize more complex patterns. The universal approximation theorem states that a feedforward network with a single hidden layer containing a finite number of neurons can approximate any continuous function on compact subsets of $\mathbb{R}^n$, given sufficient neurons and appropriate activation functions [18].

2.3 Neural networks for classification tasks

Classification represents one of the primary applications of ANNs, wherein the objective is to assign input data to predefined categories or classes [17]. In the context of power systems, classification tasks encompass fault detection, fault type identification, and fault location—the focus of this study.

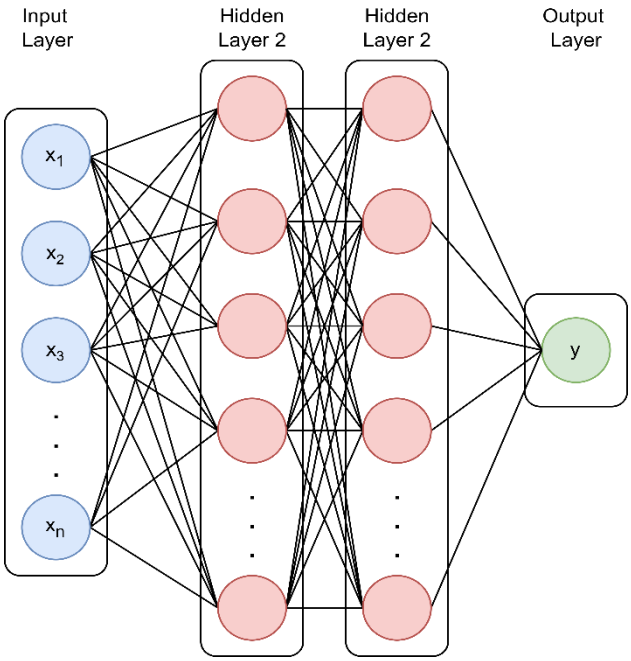


Figure 2. Schematic representation of a multilayer ANN, with n inputs, two hidden layers, and one output

When configured for classification, the architecture of an ANN must accommodate the discrete nature of the output space. For a classification problem with C classes, the output layer typically contains C neurons for multi-class classification or a single neuron for binary classification. The final layer employs specific activation functions conducive to classification tasks. For the purposes of fault detection in distribution networks, multi-class classification is performed with n + 1 different classes, where n is the number of buses in the network. The additional class corresponds to normal operating conditions (no fault detected). The activation function used in the output layer is called Softmax [19], which is commonly used in multiclass classification problems, transforming the ANN's output into a probability distribution across all classes:

$$\sigma(\mathrm{z})_j = \frac{e^{z_j}}{\sum_{k=1}^{n+} e^{z_k}} \quad (3)$$

where z_j represents the input to the softmax function for class j, and the output $\sigma(\mathbf{z})_j$ is the probability that the input belongs to class j.

During training, classification networks typically minimize cross-entropy loss functions [20]. For multi-class problems, the categorical cross-entropy is defined as:

$$L = -\sum_{i=1}^{N} \sum_{j=1}^{C} y_{ij} \log(\hat{y}_{ij}) \quad (4)$$

where N is the number of samples, C is the number of classes, y_{ij} is a binary indicator (0 or 1) if class j is the correct classification for sample i, and $\hat{y}_{ij}$ is the predicted probability that sample i belongs to class j.

For the fault location problem formulated in Section 2.1 and for the purposes of this paper, the ANN classifier, f, maps the

voltage phasor measurements, x_k, to the predicted fault location, $\hat{y}_k$. The network architecture used consists of:

- An input layer with $6n$ neurons, corresponding to the voltage magnitude and angle measurements for all phases at all buses
- Two hidden layers containing neurons with rectified linear unit (ReLU) as their activation function:

$$\text{ReLU}(x) = max(0, x) \quad (5)$$

- An output layer with $n + 1$ neurons (representing n possible fault locations plus a "no-fault" state).

This architectural configuration allows the neural network to learn the complex, nonlinear relationships between voltage measurements and fault locations across the distribution network. The model captures patterns in voltage sags, phase shifts, and other disturbances that characterize different fault locations, enabling accurate classification even in complex distribution systems with unbalanced loading and varying operational conditions.

3. Transfer learning with artificial neural networks

Transfer learning represents a paradigm in machine learning wherein knowledge gained while solving one problem is applied to a different but related problem. In the context of ANNs, transfer learning enables leveraging pre-trained models' capabilities to improve performance on target tasks with limited data availability or computational resources. This methodology is particularly valuable in power system applications, where obtaining comprehensive labeled datasets for diverse network topologies can be both time-intensive and computationally expensive [21].

3.1 Theoretical foundations of transfer learning

The fundamental premise of transfer learning is that representations learned for one task may be beneficial for another related task. Formally, given a source domain $\mathcal{D}_S$ with learning task $\mathcal{T}_S$ and a target domain $\mathcal{D}_T$ with learning task $\mathcal{T}_T$, transfer learning aims to improve the performance of the target predictive function $f_T(\cdot)$ in $\mathcal{D}_T$ using the knowledge from $\mathcal{D}_S$ and $\mathcal{T}_S$, where $\mathcal{D}_S \neq \mathcal{D}_T$ or $\mathcal{T}_S \neq \mathcal{T}_T$. In neural networks, transfer learning typically manifests through the following approaches:

- Feature extraction: Using the pre-trained network as a fixed feature extractor, where the early layers capture domain-independent features.
- Fine-tuning: Initializing a new model with parameters from a pre-trained model and then updating all or a subset of these parameters on the target dataset.
- Domain adaptation: Modifying the network architecture to minimize the discrepancy between source and target domains while maintaining performance on the source task.

The efficacy of transfer learning is predicated on the degree of similarity between the source and target domains. Greater similarity typically facilitates more effective knowledge transfer, while significant domain shift may necessitate more extensive adaptation techniques [22]. The aforementioned approaches can be implemented by the following methodologies:

3.1.1 Layer-specific transfer

In layer-specific transfer learning, certain layers of the pre-trained model are frozen while others are fine-tuned. Typically, early layers capturing low-level features are preserved, while later layers are retrained to adapt to the target task. This approach is formalized as:

$$\theta_T = \{\theta_{S,1:k}, \theta_{T,k+1:n}\} \quad (6)$$

where θ_T represents the parameters of the target model, $\theta_{S,1:k}$ denotes the parameters of the first k layers from the source model, and $\theta_{T,k+1:n}$ represents the newly initialized or fine-tuned parameters for the remaining layers.

3.1.2 Weight initialization and regularization

Transfer learning can be implemented through weight initialization, where the parameters of the target model are initialized with values from the source model and then updated via backpropagation:

$$\theta_T^{(0)} = \theta_S^{(*)} \quad (7)$$

where $\theta_T^{(0)}$ represents the initial parameters of the target model and $\theta_S^{(*)}$ denotes the optimized parameters of the source model. Additionally, regularization techniques can be employed to prevent significant deviation from the source model's weights:

$$\mathcal{L}_{reg}(\theta_T) = \mathcal{L}(\theta_T) + \lambda \|\theta_T - \theta_S^{(*)}\|^2 \quad (8)$$

where $\mathcal{L}(\theta_T)$ is the primary loss function, λ is the regularization parameter, and $\|\theta_T - \theta_S^{(*)}\|^2$ penalizes divergence from the source model's parameters.

3.2 Transfer learning for fault location in power distribution networks

In the context of fault location in power distribution networks, transfer learning offers several advantages:

- Reduction in the volume of training data required for new network topologies
- Decreased computational resources needed for model training
- Improved generalization capabilities across diverse operating conditions
- Accelerated deployment of fault location systems for newly commissioned networks

For the specific application of transferring knowledge from models trained on a larger feeder to other smaller feeders, the process involves recognizing common patterns in voltage sag propagation and phase angle shifts during fault conditions. Despite differences in network topology, operational parameters, and loading conditions across different test feeders, fundamental electromagnetic principles governing fault behavior remain consistent.

3.3 Metrics for evaluating transfer learning effectiveness

The efficacy of transfer learning for fault location can be quantified through several metrics:

- Transfer Ratio (TR): The ratio of performance achieved through transfer learning to that obtained with full training on the target domain:

$$\text{TR} = \frac{\text{Performance}_{\text{transfer}}}{\text{Performance}_{\text{full training}}} \quad (9)$$

- Data Efficiency (DE): The proportion of target domain data required to achieve equivalent performance to full training:

$$\text{DE} = \frac{\text{Data size required with transfer learning}}{\text{Data size required for full training}} \quad (10)$$

- Convergence Acceleration (CA): The reduction in training iterations needed to reach a specific performance threshold:

$$\text{CA} = \frac{\text{Iterations for convergence without transfer}}{\text{Iterations for convergence with transfer}} \quad (11)$$

These metrics provide quantitative measures for assessing the benefits of transfer learning in terms of data requirements, computational efficiency, and model performance.

4. Dataset creation

4.1 Overview

This section details the methodology employed to generate comprehensive datasets for fault location classification across multiple IEEE test feeders [23]. The datasets were specifically designed to train and evaluate ANNs for power system fault location tasks. The data generation process leveraged OpenDSS [24], an open-source electric power distribution system simulator, to model various fault scenarios across four standard IEEE test feeders of increasing complexity.

4.2 IEEE test feeders

The study utilized four standardized, non-symmetrical IEEE test feeders, each representing different network topologies and complexities commonly encountered in distribution systems. These test feeders serve as benchmarks in the power systems community and provide a standardized basis for comparing different methodologies. The test feeders utilized in this study include:

- IEEE 13-bus Test Feeder: A small, relatively simple feeder with both overhead and underground lines operating at 4.16 kV. This feeder includes unbalanced spot and distributed loads, shunt capacitors, and an in-line transformer.
- IEEE 34-bus Test Feeder: A medium-sized, long, and lightly loaded feeder operating at 24.9 kV. This feeder features two voltage regulators, an in-line transformer, unbalanced loading, and both spot and distributed loads.
- IEEE 37-bus Test Feeder: A delta-configured feeder with relatively short line segments, characterized by underground cables operating at 4.8 kV, featuring unbalanced loading and constant PQ loads.
- IEEE 123-bus Test Feeder: A large, complex feeder operating at 4.16 kV. This feeder includes multiple voltage regulators, shunt capacitors, overhead and underground lines, multiple laterals, and unbalanced loading with various load models.

Table 1 summarizes the key characteristics of each test feeder, including the number of buses, three-phase buses, voltage levels, and other distinguishing features. It needs to be mentioned that in the second column of the table, the bus number shown on each feeder is larger than the one in its corresponding name. This occurs because OpenDSS models regulators, transformers, and distributed loads as additional buses in the power system.

Table 1. Characteristics of IEEE test feeders used for dataset generation

Test Feeder	Total Buses	3-Phase Buses	Voltage Level	Special Features
IEEE 13-bus	16	11	4.16 kV	Inline transformer, unbalanced loads
IEEE 34-bus	37	29	24.9 kV	Voltage regulators, long lines
IEEE 37-bus	39	39	4.8 kV	Delta configuration, underground cables
IEEE 123-bus	132	71	4.16 kV	Multiple voltage regulators, switch operations

4.3 Fault simulation methodology

The dataset generation process involved systematically simulating faults at each bus within the test feeders. Three common fault types were considered (Figure 3) to ensure comprehensive coverage of realistic fault scenarios:

- Single-Phase-to-Ground (SPG): Represents a fault where one phase comes into contact with ground, a common occurrence in distribution systems due to events such as tree contacts or insulation failure.
- Phase-to-Phase (PP): Represents a fault where two phases make contact without ground involvement, often caused by conductor slapping or vegetation.
- Double-Phase-to-Ground (DPG): Represents a more severe fault where two phases make contact with ground simultaneously.

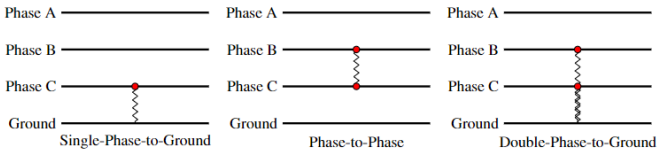


Figure 3. Schematic representation of the three fault types simulated in this study

For each three-phase bus in every test feeder, 300 faults were simulated, distributed equally among the three fault types. The fault impedance was randomly varied within a realistic range of 0.05 to 5 ohms to account for various fault resistance scenarios encountered in real-world situations. This relatively low impedance range was selected to represent bolted or low-impedance faults, which are generally more severe and critical for system protection. Additionally, to enable the network to distinguish between normal operating conditions and fault conditions, 400 data points representing normal operation without any faults were included in each dataset. These no-fault scenarios were generated by varying load levels and power factors within typical operating ranges to represent diverse normal operating conditions.

4.4 Data collection process

The fault simulation and data collection process consisted of the following steps, executed via Python using the OpenDSSDirect interface:

- System Initialization: Each test feeder was loaded into OpenDSS, and a base case power flow solution was obtained.
- Three-Phase Bus Identification: All buses in the network were examined to identify those with three phases, as these represented the locations where all fault types could be applied.
- Fault Application: For each three-phase bus and fault type combination:
  a) The system was reset to its base case.
  b) A fault with randomly selected parameters (phase selection, fault impedance) was applied.
  c) A power flow solution was obtained.
- No-Fault Scenarios: For the 400 no-fault data points:
  a) The system was reset to its base case.
  b) Load levels were randomly varied within ±20% of nominal values.
  c) Power factors were randomly varied within typical ranges (0.85-0.98).
  d) A power flow solution was obtained under these normal operating conditions.
- Feature Extraction: After each simulation (both fault and no-fault), per-unit voltage magnitudes and angles were collected from all buses in the network. These voltage measurements serve as the feature vectors for the fault location classification task.
- Label Assignment: For fault scenarios, the faulted bus name was recorded as the classification label. For no-fault scenarios, a special "no-fault" label was assigned.

4.5 Dataset characteristics

The resulting datasets exhibit the following key characteristics:

- Feature Dimensionality: Each feature vector consists of per-unit voltage measurements from all buses in the network. The dimensionality varies by test feeder, ranging from relatively small (for the 13-bus feeder) to significantly larger (for the 123-bus feeder).
- Class Balance: The dataset for each feeder contains an equal number of samples for each three-phase bus location, ensuring balanced representation across all potential fault locations. Additionally, each dataset includes 400 no-fault scenarios.
- Fault Type Distribution: Within each fault location, there is an approximately equal distribution of fault types (SPG, PP, DPG).
- Dataset Size: The total number of samples per feeder is determined by the equation:

$$Nsamples = (N3\phi buses \times 300) + 400 \quad (12)$$

where $N_{3\phi buses}$ is the number of three-phase buses in the feeder, each bus has 300 fault scenarios (approximately 33 for each fault type), and 400 represents the no-fault scenarios. Table 2 summarizes the key statistics for each generated dataset.

Table 2. Statistics of generated fault datasets

Dataset	3-Phase Buses	Faults per Bus	No-Fault Samples	Total Samples
IEEE 13-bus	11 of 16	300	400	3,700
IEEE 34-bus	29 of 37			9,100
IEEE 37-bus	39 of 39			12,100
IEEE 123-bus	71 of 132			21,700

5. Baseline and transfer learning methodology

In this section, the baseline and transfer training methodology for fault classification in IEEE distribution test feeders is described. The ANN models were trained independently on four standard IEEE test feeders: the 13-bus, 34-bus, 37-bus, and 123-bus feeders. For each feeder, a separate classifier was trained to pinpoint the location of faults. Then, the ANN trained on 123-bus feeder data was utilized for transfer learning purposes.

5.1 Data preprocessing

Prior to training, the datasets were split into training, validation, and test sets using an 80%, 10%, and 10% ratio, respectively. This partitioning ensures sufficient data for training while reserving independent sets for validation during training and final performance evaluation. Data normalization was applied to improve the convergence and performance of the ANNs. The training data for each feeder was normalized using the StandardScaler function of Python's scikit-learn library, which transforms features to have zero mean and unit variance, following the following formula:

$$X_{normalized} = \frac{X-\mu}{\sigma} \quad (13)$$

where μ is the mean and σ is the standard deviation of each feature in the training set. The validation and test sets were normalized using the same scaling parameters derived from the training set to prevent data leakage. This standardization process is critical for neural networks as it ensures all features contribute equally to the model training regardless of their original scales.

5.2 Neural network architecture

The baseline neural network architecture consists of a fully connected feedforward network with two hidden layers. The architecture was kept consistent across all feeders with the exception of the first hidden layer size, which was adjusted based on the complexity of the feeder:

- Input Layer: Dimensionality matched to the number of features in each feeder dataset
- First Hidden Layer: 40 neurons for the 13-bus, 34-bus, and 37-bus feeders; 256 neurons for the 123-bus feeder
- Second Hidden Layer: 20 neurons for all feeders
- Output Layer: Dimensionality matched to the number of potential fault locations (classes) for each feeder

Glorot uniform initialization [25], also known as Xavier initialization, was used for all weight matrices to improve training convergence. This initialization method samples weights from a uniform distribution with limits based on the number of input and output units in the weight tensor.

5.3 Baseline training process

The networks were trained using the Adam optimizer with a learning rate of 0.001 and categorical cross-entropy as the loss function, which is appropriate for multi-class classification problems. The training process was configured to use mini-batch gradient descent with a batch size of 32 samples. To prevent overfitting and optimize training efficiency, an early stopping mechanism was implemented with a patience parameter of 50 epochs. This approach monitored the validation loss and stopped training when no improvement was observed for 50 consecutive epochs, restoring the model weights to the best-performing configuration.

5.4 Transfer learning process

The transfer learning methodology leverages knowledge from the more complex 123bus feeder model to improve classification performance and training efficiency on the simpler feeders (13-bus, 34-bus, and 37-bus). This approach is particularly valuable in power systems applications, where obtaining comprehensive labeled datasets for all network topologies can be resource-intensive. The dimensional mismatch between source and target feeders is addressed by the following methodology:

- Input Adaptation Layer: A dense layer with ReLU activation that transforms the input features from the target feeder's dimensionality to match the expected input dimension of the 123-bus model:

$$x_{adapted} = \mathrm{ReLU}(W_{adapter} x_{input} + b_{adapter}) \quad (14)$$

where x_{input}, the input vector of the target distribution network, $W_{adapter}$, the weights of the new input layer, $b_{adapter}$ its bias vector and $x_{adapted}$ its output vector.

- Transferred Hidden Layers: The first two dense layers from the pre-trained 123-bus model are reused without modification to preserve the learned feature representations:

$$h_1 = \mathrm{ReLU}(W_1 x_{adapted} + b_1 \quad (15)$$

$$h_2 = \mathrm{ReLU}(W_2 h_1 + b_2) \quad (16)$$

where W_1, W_2 the weights, b_1, b_2 the bias vectors and h_1, h_2 the output vectors of the first and second layer respectively.

- New Output Layer: A dense layer with Softmax activation that maps the second hidden layer's output to the appropriate number of fault location classes for the target feeder:

$$y_{pred} = \mathrm{Softmax}(W_{out} h_2 + b_{out}) \quad (17)$$

where, W_{out}, the weights of the output layer, b_{out} its bias vector and y_{pred}, output vector of the ANN.

This architecture enables effective knowledge transfer while accommodating the structural differences between source and target networks. Additionally, the transfer learning process utilized a two-phase training strategy to optimize knowledge transfer while allowing adaptation to the target feeder's characteristics:

Phase 1: Feature Extraction (First 2500 epochs)

- The pre-trained hidden layers were frozen
- Only the input adaptation layer and the new output layer were trained
- A learning rate of 0.0001 was used with the Adam optimizer
- This phase allowed the model to adapt its input and output mappings while preserving the intermediate feature representations learned from the more complex 123-bus feeder

Phase 2: Fine-Tuning (Subsequent 2500 epochs)

- All hidden layers were unfrozen.
- The learning rate was reduced by a factor of 10 (to 0.00001) to prevent catastrophic forgetting.
- This fine-tuning phase allowed for more subtle adjustments to the entire network, optimizing the transferred knowledge for the specific characteristics of the target feeder

The categorical cross-entropy loss function was used throughout both training phases, consistent with the baseline training approach. Furthermore, the implementation included several key technical details to ensure effective knowledge transfer:

- Weight Initialization: The input adaptation and output layers used Glorot uniform initialization for stable gradient flow during training
- Batch Processing: Training was performed using full-batch gradient descent due to the moderate size of the datasets
- Training Duration: A fixed budget of 5000 total epochs (2500 for each phase) was allocated to each transfer learning experiment
- Early Stopping: Similar to the baseline training, an early stopping mechanism with a patience parameter of 50 epochs was employed to prevent overfitting and optimize training efficiency
- Performance Monitoring: Training and testing accuracy were monitored throughout the process to evaluate the effectiveness of knowledge transfer.

6. Results and discussion

This section presents and assesses the result stemming from the experiments conducted for the purposes of this study.

6.1 Baseline model training evaluation

Table 3 presents the quantitative results from the baseline model training. Contrary to intuitive expectations, training duration did not correlate linearly with network complexity. The IEEE 34-bus feeder required the highest number of epochs (928) to converge, despite representing an intermediate level of complexity. Meanwhile, the IEEE 123-bus feeder—the most complex topology with 71 three-phase buses—achieved convergence in only 305 epochs. The IEEE 13-bus and IEEE 37-bus feeders required intermediate epoch counts (415 and 613, respectively) to reach convergence. These results suggest that convergence behavior in fault location ANNs is influenced not only by network size but also by the electrical characteristics and topology of the specific distribution system. Classification accuracy varied substantially across the four test feeders. The IEEE 37bus feeder achieved the highest test accuracy (96.25%) and F1-score (0.9609), followed by the IEEE 13-bus feeder (91.67%,

0.9167), the IEEE 123-bus feeder (87.45%, 0.8508), and the IEEE 34-bus feeder (74.52%, 0.7330).

Table 3. Performance metrics for baseline neural network training on IEEE test feeders

Test Feeder	Epochs	Time (s)	Dataset	Loss	Accuracy/F1-Score
IEEE 13-bus	415	53.75	Train Validation	0.1155 0.1260	0.9260/0.9251 0.9083/0.9062
			Test	0.1203	0.9167/0.9167
IEEE 34-bus	928	183.21	Train Validation	0.5278 0.5193	0.7518/0.7371 0.7758/0.7609
			Test	0.5386	0.7452/0.7330
IEEE 37-bus	613	151.98	Train Validation	0.0900 0.1125	0.9634/0.9618 0.9463/0.9421
			Test	0.0958	0.9625/0.9609
IEEE 123-bus	305	242.06	Train Validation	0.2086 0.2092	0.8784/0.8551 0.8782/0.8559
			Test	0.2128	0.8745/0.8508

The superior performance of the IEEE 37-bus model merits particular attention, as it outperformed models for both simpler and more complex networks. This could be attributed to several distinguishing characteristics of this feeder:

- Delta configuration rather than wye configuration
- Uniform underground cable construction throughout the network
- Relatively short line segments with consistent impedance characteristics
- Balanced three-phase construction (all buses are three-phase)

These features likely produce more consistent and distinctive voltage signatures during fault conditions, facilitating more effective classification. The relatively poor performance of the IEEE 34-bus model (74.52% accuracy) can be attributed to its distinctive characteristics that complicate fault location:

- Long, lightly loaded feeder with distributed loads
- Presence of voltage regulators introducing nonlinear behavior
- Mix of one-, two-, and three-phase laterals
- Significant voltage drops along the feeder length

These characteristics create more complex and potentially similar fault signatures at different locations, increasing classification difficulty.

Training time did not directly correlate with epoch count due to differences in per-epoch computational requirements. Despite requiring the fewest epochs, the IEEE 123-bus feeder demanded the longest training time (242.06 seconds) due to its significantly larger feature space and more complex neuron structure. The computational complexity scales with the number of neurons and connections in the network, which increases substantially with the dimensionality of input features. The IEEE 13-bus feeder, with its compact structure (16 buses total), required only 53.75 seconds for training, approximately 21% of the time needed for the IEEE 123-bus model despite requiring 36% more epochs.

All models demonstrated robust generalization, as evidenced by the relatively small divergence between training and test metrics. This indicates that the selected architecture and regularization techniques (particularly early stopping) were effective in preventing overfitting. The consistent performance across datasets suggests that the models captured genuine patterns in the fault signatures rather than memorizing training examples. The IEEE 37-bus feeder exhibited particularly impressive generalization, with nearly identical performance across training, validation, and test datasets. This further supports the hypothesis that its uniform construction produces more consistent and learnable fault patterns. Confusion matrices, shown in Figure 4 were utilized to visualize the ANN predictions. Predictions that fall onto the first diagonal of the confusion matrix represent true positives, indicating that the predicted class is the correct one. Several patterns are noteworthy:

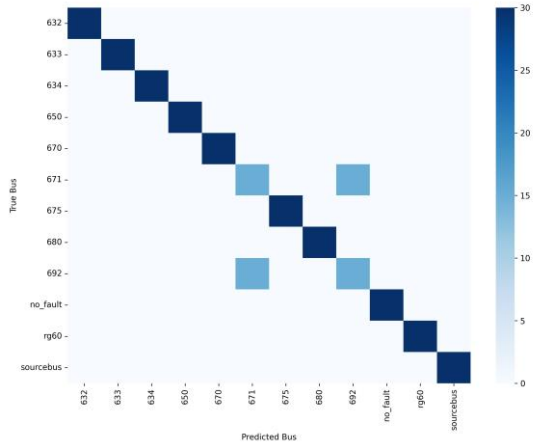
- Across all feeders, misclassifications predominantly occur between electrically adjacent buses or buses with similar electrical characteristics.
- The IEEE 13-bus and IEEE 37-bus feeders exhibit strong diagonal patterns, indicating high classification accuracy across most fault locations.
- The IEEE 34-bus feeder shows more pronounced off-diagonal elements, particularly in certain network regions, suggesting clusters of buses with similar fault signatures.
- The ”no-fault” class (representing normal operating conditions) is classified with high accuracy (95%) across all feeders, indicating robust discrimination between normal operation and fault conditions.
- In the IEEE 123-bus feeder, misclassifications exhibit spatial correlation, with higher confusion rates between buses within the same lateral branches.

These baseline results establish that neural networks can effectively locate faults across various network topologies, with performance generally correlating with network complexity and electrical characteristics.

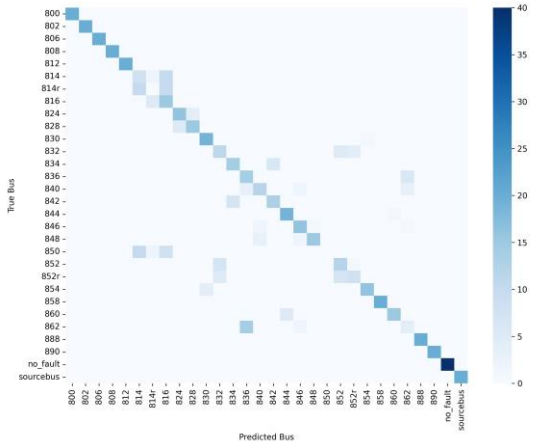
6.2 Transfer learning effectiveness

Table 4 presents a comparative analysis between baseline training and transfer learning approaches, while Table 5 quantifies the efficiency gains through two key metrics: Transfer Ratio (accuracy of transfer model divided by accuracy of baseline model) and Convergence Acceleration (baseline epochs divided by transfer learning epochs). Figures 5 and Figure 6 further visualize these performance metrics. The data reveals several significant patterns regarding knowledge transferability between distribution networks.

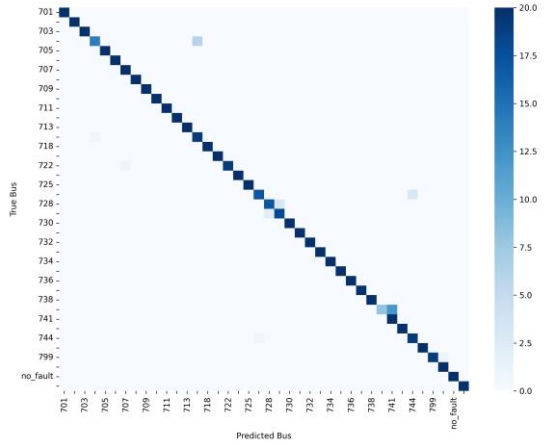
As quantified in Table 5, transfer learning demonstrated substantial improvements in training efficiency across all target feeders. The Convergence Acceleration metric—defined as the ratio of baseline epochs to transfer learning epochs—ranged from 1.68 for the IEEE 34-bus feeder to 2.56 for the IEEE 13-bus feeder. In practical terms, this represents a 61% reduction in required epochs for the IEEE 13-bus model, a 40.6% reduction for the IEEE 34-bus model, and a 60% reduction for the IEEE 37-bus model.



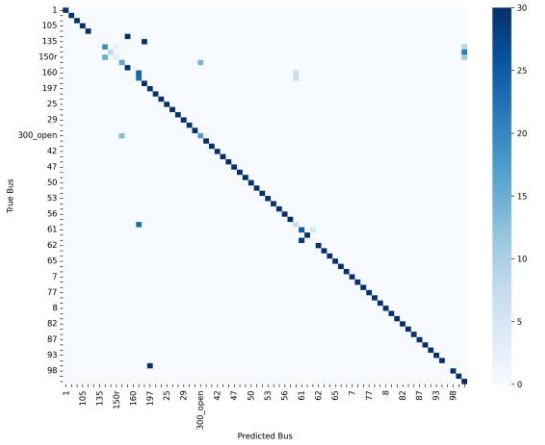
(a) IEEE 13-bus feeder



(b) IEEE 34-bus feeder



(c) IEEE 37-bus feeder



(d) IEEE 123-bus feeder

Figure 4. Confusion matrices for baseline fault location classification on test datasets for all four IEEE test feeders. Color intensity indicates classification frequency, with diagonal elements representing correct classifications. Note the stronger diagonal dominance in the 13-bus and 37-bus feeders compared to the 34-bus and 123-bus feeders.

Table 4. Performance comparison between baseline training and transfer learning

Test Feeder	Training	Epochs	Time (s)	Dataset	Loss	Accuracy/F1-Score
IEEE 13-bus	Baseline	415	53.75	Train Validation Test	0.1155 0.1260 0.1203	0.9260/0.9251 0.9083/0.9062 0.9167/0.9167
	Transfer	162	27.96	Train Validation	0.1230 0.1306	0.9201/0.9048 0.9056/0.8876
				Test	0.1249	0.9194/0.9124
IEEE 34-bus	Baseline	928	183.21	Train Validation Test	0.5278 0.5193 0.5386	0.7518/0.7371 0.7758/0.7609 0.7452/0.7330
	Transfer	551	139.24	Train Validation	0.5643 0.5716	0.7349/0.7283 0.7274/0.7210
				Test	0.5652	0.7500/0.7453
IEEE 37-bus	Baseline	613	151.98	Train Validation Test	0.0900 0.1125 0.0958	0.9634/0.9618 0.9463/0.9421 0.9625/0.9609
	Transfer	245	76.77	Train Validation	0.1375 0.1447	0.9361/0.9309 0.9300/0.9212
				Test	0.1361	0.9300/0.9241

Table 5. Transfer learning efficiency metrics

Test Feeder	Transfer Ratio	Convergence Acceleration	Time Reduction (%)	Epoch Reduction (%)
IEEE 13-bus	1.003	2.56	48.0	61.0
IEEE 34-bus	1.006	1.68	24.0	40.6
IEEE 37-bus	0.967	2.50	49.5	60.0

These findings suggest that feature representations learned from the IEEE 123-bus feeder contained generalizable knowledge about fault signatures that could accelerate learning in target networks. The magnitude of convergence improvement appeared inversely related to the topological similarity between source and target networks. The IEEE 13-bus feeder, with its compact radial structure, differs substantially from the more complex IEEE 123-bus network, yet it exhibited the greatest Convergence Acceleration (2.56). This counterintuitive result suggests that simpler networks may benefit more from transfer learning, as they can more readily adapt generalizable features learned from complex networks to their simpler topologies. The impact of transfer learning on classification accuracy varied across the three target feeders, as reflected in the Transfer Ratio metric in Table 5. For the IEEE 13-bus feeder, transfer learning yielded a Transfer Ratio of 1.003, indicating a marginal improvement in test accuracy (91.94% versus 91.67%) alongside a slight decrease in F1-score (91.24% versus 91.67%). The IEEE 34-bus feeder, which had the poorest baseline performance, showed a slightly higher Transfer Ratio of 1.006, achieving 75.00% accuracy compared to 74.52% in baseline training, and a 1.23 percentage point improvement in F1-score (74.53% versus 73.30%).

Notably, the IEEE 37-bus feeder exhibited a Transfer Ratio of 0.967, reflecting a modest performance decline with transfer learning as test accuracy decreased from 96.25% to 93.00%. This aligns with our baseline findings that identified the IEEE 37-bus feeder's unique characteristics (delta configuration, underground cables, balanced construction) as particularly conducive to neural network learning. The feature representations learned from the wye-configured, overhead IEEE 123-bus network may not have captured the distinctive electrical behaviors of the delta-configured IEEE 37-bus system, resulting in a less optimal starting point for learning. Despite this variability, all transfer learning models maintained reasonable generalization capability, with consistent performance across training, validation, and test datasets. This suggests that transfer learning preserves the robustness of the original training approach while significantly reducing computational requirements.

As documented in Table 5, transfer learning delivered consistent computational savings across all target feeders. Time reductions ranged from 24.0% for the IEEE 34-bus feeder (139.24 seconds versus 183.21 seconds) to 49.5% for the IEEE 37-bus feeder (76.77 seconds versus 151.98 seconds) and 48.0% for the IEEE 13-bus feeder (27.96 seconds versus 53.75 seconds). These efficiency gains could prove significant for practical applications, particularly when training needs to be performed on resource-constrained systems or when models must be periodically retrained to accommodate network modifications.

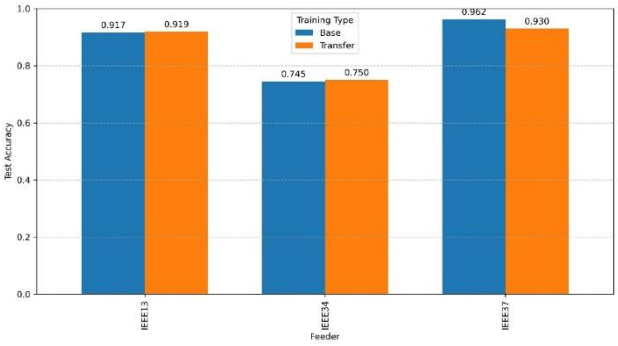


Figure 5. Comparison of test accuracy between baseline training and transfer learning across IEEE test feeders. The transfer learning approach shows comparable or slightly improved accuracy for the IEEE 13-bus and 34-bus feeders, while experiencing a modest decline for the IEEE 37-bus feeder.

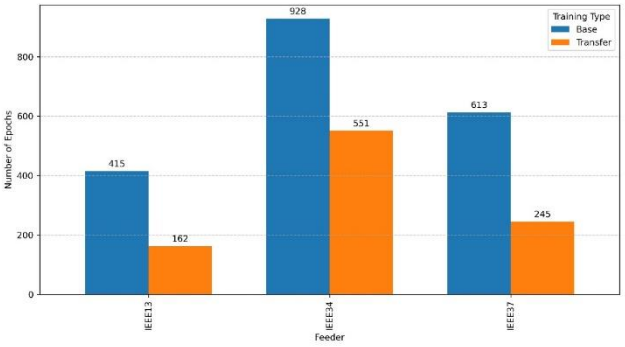
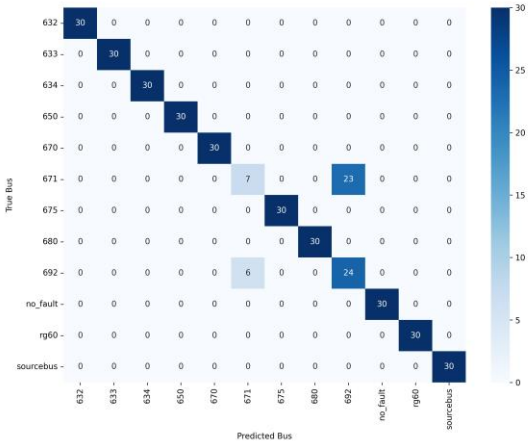
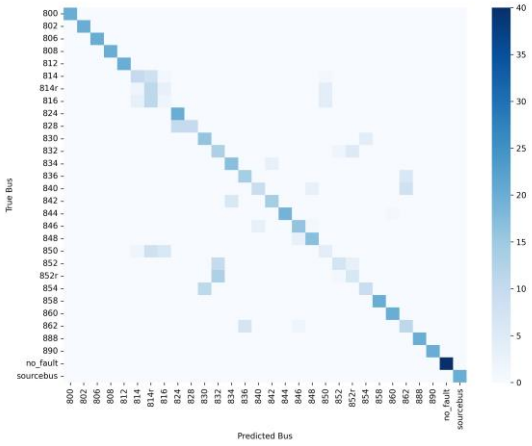


Figure 6. Comparison of training epochs required for convergence between baseline training and transfer learning across IEEE test feeders. Transfer learning consistently reduces the number of epochs required, with reductions of 61% for IEEE 13-bus, 41% for IEEE 34-bus, and 60% for IEEE 37-bus feeders.

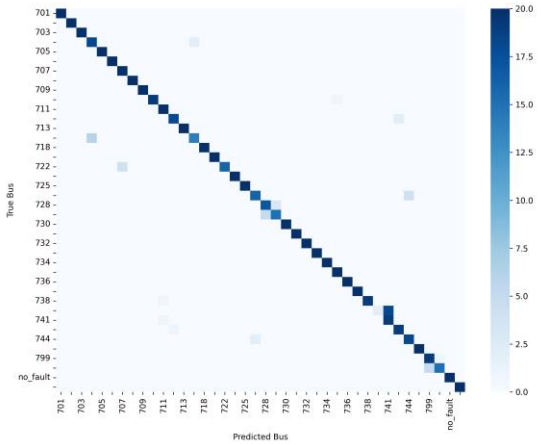
The computational benefits of transfer learning exhibit a compound effect: not only do the models converge in fewer epochs, but the initial training phase (with frozen hidden layers) requires substantially less computation per epoch, as backpropagation is limited to the output layers. This approach enables rapid adaptation to new network topologies without sacrificing performance, making neural network fault location more practical for real-world distribution system applications. The successful application of transfer learning across diverse network topologies suggests that neural networks learn generalizable representations of fault signatures in electrical distribution systems. These representations appear to capture fundamental physical relationships between voltage measurements and fault locations that transcend specific network configurations. Several factors may explain these generalizable representations:



(a) IEEE 13-bus feeder (Transfer)



(b) IEEE 34-bus feeder (Transfer)



(c) IEEE 37-bus feeder (Transfer)

Figure 7. Confusion matrices for transfer learning fault location classification on test datasets for the three target IEEE test feeders. The transfer learning models preserve the strong diagonal structure indicative of accurate classification, with marginally different error patterns compared to their baseline counterparts.

- Electrical principles governing fault propagation remain consistent across networks despite topological differences
- Relative voltage perturbations during faults follow similar patterns regardless of absolute voltage levels or network size
- Feature hierarchies learned by neural networks may correspond to physical phenomena at different spatial scales within the network
- The nonlinear transformations learned by hidden layers may effectively normalize diverse network characteristics to a common representation space

The variable impact of transfer learning across different target feeders highlights the importance of electrical characterization in knowledge transfer. While topological differences certainly influence transferability, our results suggest that electrical characteristics (configuration, construction type, phase balance) may play a more determinative role in successful knowledge transfer between distribution networks.

Figure 7 presents the prediction results of the transfer learning ANNs in a visual manner, showcasing the similarity in performance with the baseline training ones.

7. Conclusion

This paper investigated the application of transfer learning techniques to artificial neural networks for fault location in power distribution systems. Comprehensive experimentation across multiple IEEE test feeders demonstrated that knowledge gained from training on complex network topologies can be effectively transferred to accelerate and potentially improve fault location in simpler networks. The findings contribute several important insights to the field of power system fault detection and classification. First, the results establish that transfer learning enables significant improvements in training efficiency. The Convergence Acceleration metric, ranging from 1.68 to 2.56 across different target feeders, translates to epoch reductions of 40.6% to 61.0%. This efficiency gain has substantial practical implications for utilities deploying fault location systems across diverse network topologies, as it dramatically

reduces the computational resources and time required for model training. Second, transfer learning was found to maintain or slightly improve classification accuracy for two of the three target feeders (Transfer Ratios of 1.003 and 1.006 for the IEEE 13-bus and 34-bus feeders, respectively). Notably, the IEEE 34-bus feeder—which exhibited the poorest baseline performance due to its challenging electrical characteristics—benefited most from knowledge transfer, suggesting that transfer learning can be particularly valuable for networks with complex fault signatures. Third, the analysis revealed that electrical characteristics may play a more significant role than topological similarity in determining transfer learning effectiveness. The modest performance decline observed for the IEEE 37-bus feeder (Transfer Ratio of 0.967) highlights the importance of considering network configuration (delta versus wye), construction type (overhead versus underground), and phase balance when applying transfer learning across different distribution systems. The computational efficiency improvements demonstrated in this study—with time reductions ranging from 24.0% to 49.5%—further enhance the practical applicability of neural network approaches for fault location. These efficiency gains, combined with the ability to accelerate model training through knowledge transfer, address key barriers to the widespread deployment of intelligent fault location systems in distribution networks. Future research directions emerging from this work include:

- Investigation of transfer learning effectiveness under reduced data availability scenarios, particularly relevant for newly commissioned networks with limited fault history
- Exploration of alternative transfer learning architectures, such as domain adaptation techniques, to better address electrical characteristic differences between source and target networks
- Extension of the knowledge transfer approach to incorporate dynamic network reconfigurations and the presence of distributed energy resources
- Development of hybrid approaches combining physics-informed constraints with transfer learning to enhance generalization across diverse network topologies

In conclusion, this study demonstrates that transfer learning offers a promising approach to address the challenges of data scarcity and computational efficiency in power system fault location. By leveraging knowledge from more complex networks, utilities can deploy accurate fault location systems across diverse distribution feeders with reduced data requirements and training time, ultimately contributing to improved power system reliability through faster fault isolation and service restoration.

Acknowledgements

The PANNEL (Physics-aware neural networks and edge processing for low voltage systems) project has received funding from the European Union, via the oc1-2024-TIS-01 issued and implemented by the ENFIELD project, under the grant agreement No 101120657.

Ethical issue

The authors are aware of and comply with best practices in publication ethics, specifically concerning authorship (avoidance of guest authorship), dual submission, manipulation of figures, competing interests, and compliance with policies on research ethics. The authors adhere to publication requirements that the submitted work is original and has not been published elsewhere in any language.

Data availability statement

The manuscript contains all the data. However, more data will be available upon request from the corresponding author.

Conflict of interest

The authors declare no potential conflict of interest.

References

[1] Majid Jamil, Sanjeev Kumar Sharma, and Rajveer Singh. Fault detection and classification in electrical power transmission system using artificial neural network. SpringerPlus, 4(1):334, December 2015.

[2] Hamed Rezapour, Sadegh Jamali, and Alireza Bahmanyar. Review on artificial intelligence-based fault location methods in power distribution networks. Energies, 16(11):4636, 2023.

[3] Mohammad Pourahmadi-Nakhli and Ahmad A Safavi. Path characteristic frequencybased fault locating in radial distribution systems using wavelets and neural networks. IEEE Transactions on Power Delivery, 26(2):772–781, 2011.

[4] Ali Rafinia and Javad Moshtagh. A new approach to fault location in three-phase underground distribution system using combination of wavelet analysis with ann and fls. International Journal of Electrical Power & Energy Systems, 55:261–274, 2014.

[5] Mohammad Dashtdar, Rahman Dashti, and Hamidreza Shaker. Distribution network fault section identification and fault location using artificial neural network. In 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE), pages 273–278. IEEE, 2018.

[6] Yasin Aslan and Yılmaz Engin Yagan. Artificial neural-network-based fault location for˘ power distribution lines using the frequency spectra of fault data. Electrical Engineering, 99:301–311, 2017.

[7] Sayed Ahmad Mousavi Javadian, Amir Massoud Nasrabadi, Mahmoud Reza Haghifam, and Javad Rezvantalab. Determining fault's type and accurate location in distribution systems with dg using mlp neural networks. In 2009 International Conference on Clean Electrical Power, pages 284–289. IEEE, 2009.

[8] Mustafa Bakkar, Santiago Bogarra, Francisco Corcoles, Ahmed Aboelhassan, Shuai´ Wang, and Jose Iglesias. Artificial intelligence-based protection for smart grids.´ Energies, 15(14):4933, 2022.

[9] GM Shafiullah, MA Abido, and Zeyad Al-Hamouz. Wavelet-based extreme learning machine for distribution grid fault location. IET Generation, Transmission & Distribution, 11(16):4256–4263, 2017.

[10] Kalpana Lout and Raj K Aggarwal. Current transients based phase selection and fault location in active distribution networks with spurs using artificial intelligence. In 2013 IEEE Power & Energy Society General Meeting, pages 1–5. IEEE, 2013.

[11] Zhidi Lin, Dongliang Duan, Qi Yang, Xuemin Hong, Xiang Cheng, Liuqing Yang, and Shuguang Cui. Data-

Driven Fault Localization in Distribution Systems with Distributed Energy Resources. Energies, 13(1):275, January 2020.
[12] Fatemeh Mohammadi Shakiba, Milad Shojaee, S. Mohsen Azizi, and Mengchu Zhou. Transfer Learning for Fault Diagnosis of Transmission Lines, January 2022. arXiv:2201.08018 [cs].
[13] Ivan L. Degano, Leandro Fiaschetti, and Pablo A. Lotito. Location of faults based on´ deep learning with feature selection for meter placement in distribution power grids. International Journal of Emerging Electric Power Systems, 25(5):657–666, October 2024.
[14] Fatemeh Mohammadi Shakiba, Milad Shojaee, S. Mohsen Azizi, and Mengchu Zhou. Transfer learning for fault diagnosis of transmission lines. arXiv preprint arXiv:2201.08018, 2022.
[15] Yuantao Yao, Daochuan Ge, Jie Yu, and Min Xie. Model-based deep transfer learning method to fault detection and diagnosis in nuclear power plants. Frontiers in Energy Research, 10:823395, 2022.
[16] Supriya Asutkar and Siddharth Tallur. Deep transfer learning strategy for efficient domain generalisation in machine fault diagnosis. Scientific Reports, 13(1):6607, 2023.
[17] Richard O. Duda, Peter E. Hart, and David G. Stork. Pattern Classification (2nd Edition). Wiley-Interscience, 2 edition, November 2000.
[18] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. Neural Networks, 2(5):359–366, 1989.
[19] John S. Bridle. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. In Franc¸oise Fogelman Soulie´ and Jeanny Herault, editors,´ Neurocomputing, pages 227–236, Berlin, Heidelberg, 1990. Springer Berlin Heidelberg.
[20] Kevin P. Murphy. Probabilistic Machine Learning: An introduction. MIT Press, 2022.
[21] Stevo Bozinovski. Reminder of the first paper on transfer learning in neural networks, 1976. Informatica (Slovenia), 44(3), 2020.
[22] Claudia Ehrig, Benedikt Sonnleitner, Ursula Neumann, Catherine Cleophas, and Germain Forestier. The impact of data set similarity and diversity on transfer learning success in time series forecasting, 2024.
[23] Resources – IEEE PES Test Feeder. - https://cmte.ieee.org/pes-testfeeders/resources/
[24] OpenDSS - https://www.epri.com/pages/sa/opendss
[25] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, volume 9 of Proceedings of Machine Learning Research, pages 249–256, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR.