Cureus Journal of Computer Science
https://doi.org/10.7759/s44389-026-00386-3

Original Article

Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets

Michael W. Apprey [1], Daniel Kwegyir [1, ✉], Emmanuel A. Frimpong [1], Frank Kulor [2], Kafui T. Agbevanu [3], Evans W. Akpari [2], Peter Asigri [1]

1. *Electrical and Electronic Engineering, Kwame Nkrumah University of Science and Technology, Kumasi, GHA*
2. *Electrical and Electronic Engineering, Ho Technical University, Ho, GHA*
3. *Computer Science, Ho Technical University, Ho, GHA*

Received: August 28, 2026 | Review began: October 04, 2026 | Review ended: October 21, 2026 | Published: November 23, 2026

© **Copyright** 2026
This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

Accurate fault classification in electrical transmission lines is critical for system stability but remains challenging under limited training data, especially in resource-constrained environments. Existing machine learning approaches often require extensive datasets and struggle with multi-target fault identification. In this work, we propose the novel adaptive multi-target ensemble classifier (AMTEC), which integrates per-target feature selection, dynamic classifier selection, F1-score-weighted soft voting, and random forest meta-learning to enhance fault classification using limited data. Evaluated on a publicly available simulated dataset, AMTEC achieved 89.02% overall accuracy with 231 training samples, surpassing prior benchmarks by over 6 percentage points, and maintained 86.87% accuracy at an extreme scarcity of 100 samples. These results indicate AMTEC's data efficiency and suggest its potential for practical deployment in intelligent protection systems within data-scarce power networks, particularly in developing countries.

Categories: AI applications, AI/ML-based decision support systems, Machine Learning (ML)

Keywords: adaptive multi-target ensemble classifier, fault classification, limited data condition, transmission line protection, machine learning

Introduction

Electrical transmission networks serve as the primary grid infrastructure for long-range transport of electrical energy to industries and residential settings [1,2]. The reliable operation of the transmission line is essential for maintaining power quality, system stability, and continuity of supply. However, transmission lines remain highly susceptible to faults resulting from lightning strikes, equipment ageing and malfunction, insulation failures, environmental factors, and operational disturbances [3-5]. These faults can result in severe consequences, including equipment damage, widespread power outages, safety risks, and substantial economic losses, which amount to billions of dollars annually on a global scale [6,7].

Transmission line faults are generally classified into asymmetric faults, including single-line-to-ground (LG), line-to-line (LL), and double-line-to-ground (LLG) faults, as well as symmetric three-phase (LLL) faults. Accurate identification and classification of these fault types are essential for the rapid isolation of faulted sections, minimisation of service interruptions, and prevention of fault propagation across the power network [4,6,8]. Furthermore, the increasing integration of smart grid technologies, distributed generation resources, and variable renewable energy sources has

significantly increased the complexity of transmission system operation. These developments alter conventional fault-current characteristics and introduce a wider range of abnormal operating conditions, thereby posing additional challenges to reliable fault detection and classification [9,10].

Conventional power system operators use distance relays, micro-impedance estimations, line differential comparisons, and phasor measurement unit (PMU) updates for detection and classification. However, these methods may deteriorate under high-impedance faults, where fault resistance elevates impedance beyond relay thresholds, causing undetected or misclassified faults [11,12]. Integration of inverter-based resources, including wind and solar photovoltaic farms, introduces non-sinusoidal currents that compromise phasor-based assumptions and reduce detection accuracy [13,14]. Series-compensated transmission lines introduce capacitive impedance that causes measurement errors, leading to relay under-reach or over-reach [15]. Additionally, power swings, load encroachment, evolving faults, bidirectional power flows, distributed generation, and system topologies further challenge threshold-based protection schemes and render traditional protection less reliable [16,17]. Consequently, these challenges have motivated data-driven methods adaptable to changing grid conditions.

Machine learning (ML) has emerged as a powerful data-driven tool for fault detection. These approaches have the capacity to learn complex fault patterns directly from fault voltage and current data to detect and classify fault [4,18]. They include algorithms such as support vector machines (SVMs), decision trees (DTs), random forests (RFs), ensemble learners, and deep neural networks. Such methods have been reported in the literature to achieve classification accuracies exceeding 99% under controlled conditions with large and well-balanced datasets. However, many existing ML methods for fault classification depend on extensive training datasets that are often scarce in practical power system settings. Consequently, conventional ML models frequently exhibit substantial performance declines when trained on limited data, limiting their real-world effectiveness. Moreover, traditional classifiers often face challenges in simultaneously managing multiple fault targets, ground and phases A, B, and C, resulting in inaccurate or delayed fault classification, increased system downtime, equipment damage risk, and diminished power system reliability [6,7,15,19].

Despite the remarkable success of ML techniques in transmission line fault classification, their practical deployment remains constrained by the limited availability of labelled fault data. In real-world transmission networks, fault events occur relatively infrequently, while the acquisition, storage, and annotation of fault records are costly and time-consuming processes [6,20]. This challenge is particularly pronounced in developing regions, including many countries in sub-Saharan Africa, where rapid grid expansion is often accompanied by limited sensor coverage, inadequate monitoring infrastructure, and sparse historical fault databases. Consequently, developing fault classification models capable of maintaining high performance under severe data scarcity has become an important research priority [21].

Among the studies addressing limited-data fault classification, the multi-target ensemble classifier (MTEC) proposed by [8] represents the current state-of-the-art. Using only 231 training samples, MTEC achieved an overall classification accuracy of 82.92%, demonstrating the potential of ensemble learning for data-constrained environments. Nevertheless, several limitations remain. First, all six extracted features are applied uniformly across all fault targets, without considering target-specific feature relevance. Second, only a single optimal classifier is selected for each target, thereby limiting the potential benefits of ensemble diversity. Third, the adopted hard-voting mechanism assigns equal importance to all participating classifiers regardless of their individual predictive performance. Finally, the evaluation was conducted exclusively using 231 training samples, leaving the behaviour of the framework under more extreme data scarcity conditions unexplored.

Addressing these limitations is of significant practical importance because accurate fault classification forms the foundation of modern power system protection and reliability. In many emerging and resource-constrained power systems, the availability of extensive fault datasets cannot be guaranteed due to economic, operational, and infrastructural constraints. Therefore, there is an increasing need for intelligent fault diagnostic systems that can achieve high classification accuracy using only limited training data. Such capabilities can improve system reliability, reduce equipment damage, minimise outage durations, and facilitate the deployment of intelligent protection schemes in environments where comprehensive monitoring infrastructure is unavailable [8].

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3. DOI https://doi.org/10.7759/s44389-026-00386-3

To address the identified research gaps, this paper proposes the adaptive multi-target ensemble classifier (AMTEC), an enhanced multi-target ensemble framework that extends the MTEC architecture through four complementary improvements: target-specific minimum redundancy maximum relevance (mRMR) feature selection, adaptive top-three classifier selection, F1-score-weighted soft voting, and RF-based stacked meta-learning.

This study makes three principal contributions. First, we propose a novel adaptive multi-target ensemble that incorporates target-specific mRMR feature selection, dynamic top-three ensemble construction, F1-score-weighted soft-probability voting, and RF meta-learner stacking, using a heterogeneous pool of standard ML classifiers as base learners. Second, we conduct a comprehensive evaluation under both a limited-data regime (231 training samples) and an extreme-scarcity regime (100 training samples), enabling direct benchmarking against conventional classifiers and the MTEC framework. Third, we conduct a systematic comparison of AMTEC against seven conventional ML classifiers and the MTEC baseline across all fault targets.

The remainder of the paper is organised as follows. The 'Related works' section reviews the relevant literature on transmission line fault classification and ML-based fault diagnostic techniques. The 'Materials and Methods' section presents the theoretical foundations of the proposed AMTEC framework and describes the research methodology and experimental procedures. The 'Results and Discussion' section reports the performance evaluation and discusses the findings, while the 'Conclusions' section concludes the paper and outlines future research directions.

Related works

Large-Data ML Approaches for Fault Classification

ML techniques have become increasingly popular for transmission line fault classification due to their ability to learn complex nonlinear relationships from electrical measurements and provide rapid fault diagnosis [4,22]. Most existing studies, however, have been developed and evaluated using large simulation-generated datasets containing thousands of training samples. Under such conditions, classification accuracies exceeding 99% are routinely reported, indicating the performance levels achievable when sufficient training data are available.

Ensemble learning methods have demonstrated particularly strong performance in this regard. Bin Akter et al. [4] employed a DT, RF, and K-nearest neighbour (KNN) ensemble using PMU data and reported a combined accuracy of 99.88%, outperforming the individual constituent classifiers. Similarly, the hybrid RF-long short-term memory (LSTM)-tuned KNN framework proposed in [5] achieved 99.96% accuracy on a multi-label transmission line fault dataset, while Kharusi et al. [13] reported perfect classification accuracy using a bagging ensemble combined with forward feature selection on a 2,661-sample transmission network dataset containing inverter-based generation.

Deep learning approaches have produced similarly impressive results. Ebrahimi et al. [15] achieved 99.98% accuracy using a bidirectional LSTM network trained on 22,680 samples from a series-compensated transmission line. Harish et al. [19] reported 99.85% accuracy using a particle swarm optimisation-based weighted extreme learning machine on a 24,654-sample IEEE 39-bus dataset, while Omar et al. [20] obtained 99.55% using an LSTM architecture on a 400-kV transmission line under noisy operating conditions. Likewise, Turanli et al. [6], Tunio et al. [7], Alhanaf et al. [23], Yang and Yang [24], Nayak et al. [25], and Rahman et al. [1] all reported classification accuracies exceeding 99% using various combinations of convolutional neural networks (CNNs), temporal convolutional networks, wavelet transforms, hybrid CNN-LSTM architectures, and multilayer perceptron neural networks.

Evidence from related fault diagnosis domains further supports the effectiveness of ML when abundant data are available. For example, da Costa et al. [26] compared several classifiers for photovoltaic fault diagnosis using 7,200 samples and found that artificial neural networks and SVMs achieved accuracies above 98%, whereas KNN and DTs exhibited noticeably lower performance.

Collectively, these studies demonstrate that modern ML algorithms can achieve near-perfect transmission line fault classification performance when trained using large datasets. However, their reported accuracies are obtained under conditions that are rarely encountered in practical transmission networks, particularly in developing countries where

fault records are often limited. Consequently, the robustness of these methods under severe data scarcity remains largely unexplored.

Fault Classification Under Data Scarcity

In contrast to the extensive literature on large-data fault classification, studies investigating classifier performance under limited training data conditions remain relatively scarce. This issue is particularly important because real transmission systems typically experience faults infrequently, resulting in small and highly imbalanced fault databases.

Several studies have reported substantial performance degradation as training sample sizes decrease. Goni et al. [12] demonstrated a consistent reduction in classification accuracy across multiple classifier families as the number of training samples was reduced. Similarly, Maduako et al. [18] observed a decline in CNN accuracy from above 95% to 82.85% when the training dataset was restricted to 1,198 samples. Xu et al. [27] proposed a heterogeneous ensemble with weighted soft voting for small-sample transmission line fault diagnosis, achieving 93.84% accuracy under a single-target multiclass formulation. Fahim et al. [28] reported an accuracy of 84.40% using only 208 training samples, representing one of the earliest benchmarks near the sample sizes considered in the present study.

Other investigations have highlighted additional challenges associated with data scarcity. Shakiba et al. [29] showed that transfer learning performance deteriorated significantly under domain-shift conditions, achieving only 82.42% accuracy with a LeNet-5 CNN model. Although Ijaz et al. [30] reported 99.09% accuracy using few-shot learning, the approach relied on meta-training across multiple related tasks, an assumption that is generally not satisfied in single transmission line fault classification applications. Furthermore, Thanapol et al. [31] demonstrated that CNN-based models experience substantial generalisation difficulties when trained using fewer than approximately 500 samples.

These findings collectively suggest that the high accuracies reported in large-data studies do not necessarily translate to practical deployment scenarios where labelled fault records are scarce. More importantly, they reveal that, for multi-target and conventional single-target formulations evaluated on small training sets drawn from a single dataset, reported accuracy typically falls in the 82-85% range regardless of classifier architecture. Two reviewed studies report higher values under different conditions: Xu et al. [27] obtained 93.84% with a heterogeneous weighted-voting ensemble in a single-target multiclass formulation, and Ijaz et al. [30] obtained 99.09% using few-shot learning with meta-training across multiple related tasks. Neither setting is directly comparable to the multi-target, single-dataset formulation adopted here, but both are noted because they bound the range of results reported in the limited-data literature.

Multi-Target Ensemble Learning for Fault Diagnosis

Among the limited-data studies, the MTEC proposed by [8] represents the most relevant benchmark for the present work. Unlike conventional single-output classifiers, MTEC addressed the transmission line fault classification problem as a multi-target learning task by simultaneously predicting the four fault indicators corresponding to phases A, B, C, and ground.

Using only 231 training samples, MTEC achieved an overall multi-target classification accuracy of 82.92%, establishing the current state-of-the-art for limited-data multi-target transmission line fault classification on the dataset employed in this study. The results demonstrated the effectiveness of ensemble learning under data-constrained conditions and highlighted the potential of multi-target modelling for fault diagnosis applications.

Despite its success, several limitations remain. First, feature selection was not performed independently for each target, potentially introducing redundant or weakly informative features into individual target models. Second, the framework selected only a single best-performing classifier for each target, limiting the diversity benefits typically associated with ensemble learning. Third, the adopted hard-voting strategy assigned equal importance to all participating classifiers irrespective of their predictive reliability. Finally, the framework was evaluated only at the 231-sample level, leaving its robustness under more severe data scarcity conditions unknown.

Literature Synthesis and Research Gaps

The reviewed literature reveals a clear performance transition as training data availability decreases. Under large-data conditions involving thousands of samples, ML and deep learning classifiers consistently achieve accuracies exceeding 99%. In contrast, performance can deteriorate substantially as the amount of labelled training data is reduced, highlighting the challenge of maintaining reliable fault classification under data-scarce conditions [28]. When the training dataset is reduced to approximately 250 samples or fewer, classification performance generally declines, with reported accuracies falling to around the low-to-mid 80% range across the evaluated classifiers.

Although MTEC [8] currently provides the strongest reported performance within this limited-data regime, several important research gaps remain unresolved. First, within the literature reviewed above, no previous study has investigated target-specific filter-based feature selection within a multi-target ensemble framework for transmission line fault classification. Existing approaches employ a common feature subset for all targets despite the possibility that different fault indicators may depend on different feature combinations.

Second, current limited-data multi-target frameworks rely primarily on a single classifier for each target, thereby underutilising the diversity advantages that ensemble methods can provide. Third, existing voting mechanisms generally treat all classifiers equally despite potentially large differences in target-specific predictive performance. The combination of performance-weighted soft voting and stacked meta-learning remains unexplored for multi-target fault classification under severe data scarcity.

Finally, within the studies reviewed above, no previous study has systematically investigated multi-target fault classification using only 100 training samples, leaving the behaviour of ensemble classifiers under extreme-scarcity conditions largely unknown.

Motivated by these gaps, this study proposes the AMTEC, a data-efficient ensemble framework designed specifically for limited-data transmission line fault classification. AMTEC extends the MTEC architecture through target-specific mRMR feature selection, adaptive top-three classifier selection, F1-score-weighted soft voting, and RF-based stacked meta-learning. Through these enhancements, the proposed framework seeks to improve classification accuracy, increase robustness under severe data scarcity, and provide a practical fault diagnosis solution for resource-constrained power systems.

Materials And Methods

This section provides an overview and architecture of the proposed AMTEC architecture for fault classification. The AMTEC is a hybrid ensemble framework designed to address the limitations of existing multi-target fault classification methods under limited training data conditions. Figure *1* presents the proposed AMTEC flowchart, illustrating the sequential processing pipeline from raw input features to final multi-target predictions. AMTEC extends the MTEC proposed in [8] through four principal innovations: per-target mRMR feature selection, dynamic top-three classifier selection, F1-score-weighted soft probability aggregation, and RF meta-learner stacking. These innovations are integrated into a five-stage processing pipeline that is executed independently for each fault type.

In Stage 1, per-target mRMR feature selection is applied, tailored to the input feature space for each fault type. In Stage 2, seven base classifiers are trained independently on the selected feature subset, and their validation F1-scores, computed on an internal validation split of the training partition, are obtained. In Stage 3, the top three classifiers are dynamically selected based on their validation F1-scores, computed on an internal validation split of the training partition. In Stage 4, probability estimates from the selected classifiers are aggregated using F1-score-weighted averaging to construct the meta-feature representation. Finally, in Stage 5, a 75-tree RF meta-learner is trained and used for inference on the constructed meta-feature representation to generate the final binary prediction.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3. DOI https://doi.org/10.7759/s44389-026-00386-3

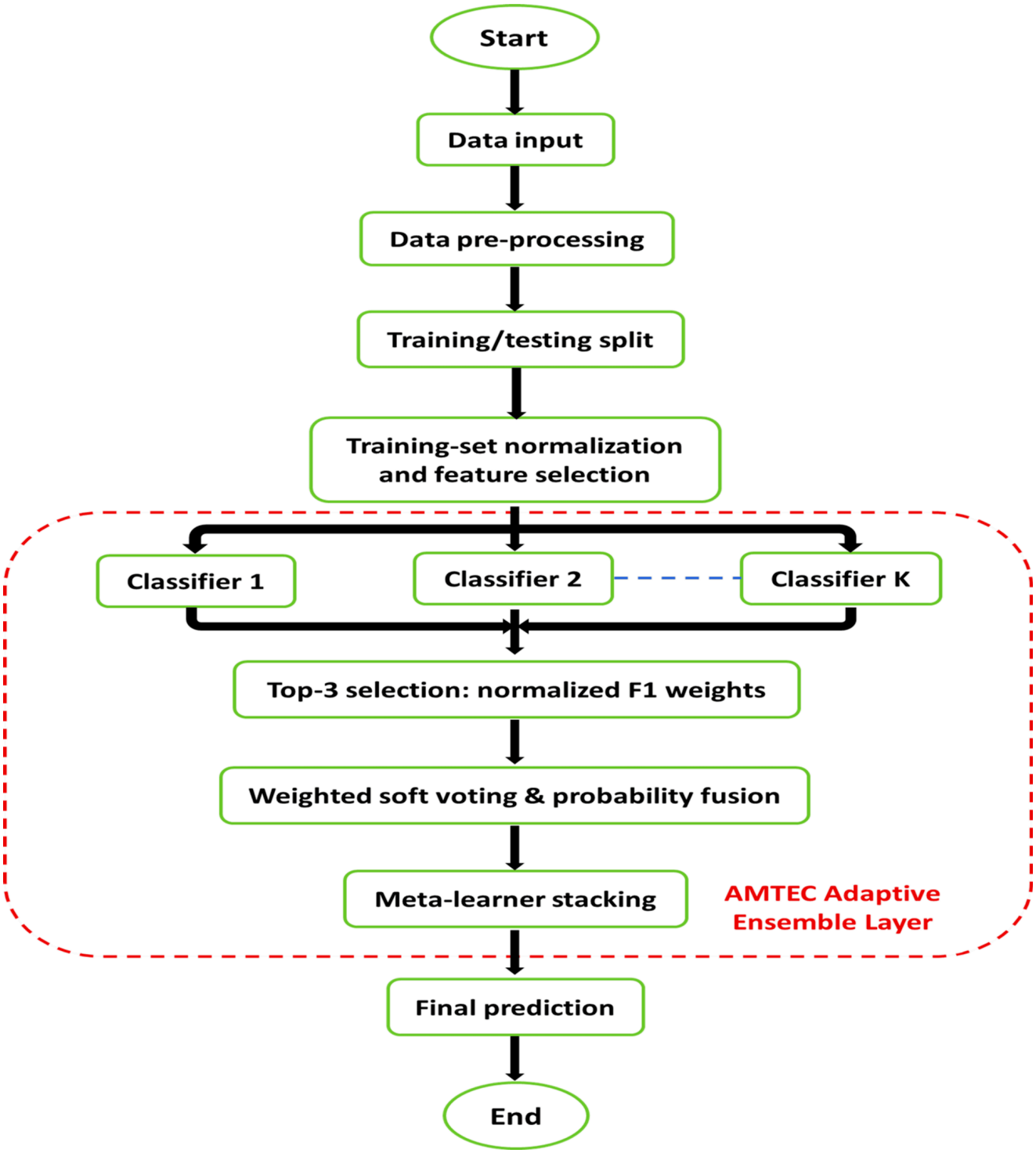


FIGURE 1: AMTEC architecture flowchart diagram

AMTEC, Adaptive Multi-Target Ensemble Classifier

Stage 1: Per-target mRMR feature selection

The proposed AMTEC framework receives six electrical measurements comprising the three-phase currents and voltages. The input feature vector is defined in Equation (1) [8].

$$X = \{ I_a,\ I_b,\ I_c,\ V_a,\ V_b,\ V_c \} \tag{1}$$

where I_a, I_b, and I_c denote the phase currents, while V_a, V_b, and V_c represent the corresponding phase voltages.

Although these six measurements are available for every operating condition, their predictive importance differs across the individual fault targets. Consequently, a feature that is highly informative for one fault type may contribute little information for another. Under limited training data conditions, retaining irrelevant or highly correlated features increases model complexity, inflates estimation variance, and reduces the generalisation capability of the learning algorithm. To mitigate these effects, AMTEC performs feature selection independently for each fault target using the mRMR criterion [32].

The mRMR algorithm seeks an optimal subset of features by simultaneously maximising their statistical relevance to the target class while minimising redundancy among the selected features. These dual objectives preserve discriminative information while avoiding unnecessary duplication of highly correlated measurements. The optimisation criterion is expressed in Equation (2) [32].

$$\mathbf{mRMR} = \arg\max_{x_k} \left[I(x_k; y) - \lambda \frac{1}{|S|} \sum_{x_j \in S} I(x_k; x_j) \right] \tag{2}$$

where $I(x_k; y)$ denotes the mutual information between candidate features x_k and the target fault label y, thereby measuring the predictive relevance of the features. The second term, $\lambda \frac{1}{|S|} \sum_{x_j \in S} I(x_k; x_j)$, represents the average mutual information between the candidate feature and the previously selected subset *S*, weighted by *λ*, thereby quantifying feature redundancy. The operator identifies the candidate feature that maximises the overall selection criterion. In this study, *λ* = 0.3 is used throughout, and the effect of *λ* is examined in the sensitivity analysis reported in this study.

Unlike pure relevance-based feature selection methods, mRMR explicitly balances relevance and redundancy. Consequently, features that provide complementary information are preferred over highly correlated measurements, even when both exhibit strong individual relevance [32]. This enables the proposed AMTEC framework to construct compact, target-specific feature subsets that improve classifier robustness while reducing the risk of overfitting under limited training data conditions.

Stage 2: Base classifier training

For each fault target $j \in \{G, C, B, A\}$, the feature subset selected in Stage 1 is used to train seven heterogeneous base classifiers (*k*), where $k \in \{\mathbf{KNN}, \mathbf{SVC}, \mathbf{RF}, \mathbf{DT}, \mathbf{AdaBoost}, \mathbf{GBM}, \mathbf{GNB}\}$. Accordingly, AMTEC constructs a 7×4 classifier-target matrix comprising 28 independently trained binary classification models. Each model is trained only on the feature subset selected for its corresponding fault target, thereby allowing the learning process to adapt to target-specific voltage and current characteristics. The selected classifiers represent different learning paradigms and were included to promote ensemble diversity through complementary decision boundaries. The seven selected classifiers are as follows.

K-Nearest Neighbours

KNN is a non-parametric, instance-based classifier that assigns a query sample to the class most frequently represented among its nearest training samples. In the proposed framework, the Manhattan distance in Equation (3) is used to quantify similarity [33,34].

$$d(x, x_i) = \sum_{r=1}^{p} |x_r - x_{ir}| \tag{3}$$

where p is the number of selected features. The L_1-based distance is less sensitive than the squared Euclidean metric to isolated large deviations that may arise from fault-induced spikes in individual phase measurements. KNN, therefore, provides a flexible local decision rule without assuming a predefined data distribution [34].

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3. DOI https://doi.org/10.7759/s44389-026-00386-3

Support Vector Classification

Support vector classification (SVC) determines a separating hyperplane that maximises the margin between the two classes. Its soft-margin formulation permits limited classification errors while controlling model complexity, making it suitable for small and potentially overlapping datasets [35-37]. The optimisation problem is expressed in Equation (4).

$$\min_{w,b,\xi} \left[\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N} \xi_i \right] \tag{4}$$

subject to

$$y_i \left[w^T \phi(x_i) + b \right] \geq 1 - \xi_i, \quad \xi_i \geq 0 \tag{5}$$

where C controls the trade-off between margin maximisation and training error, while $\phi(\cdot)$ maps the input into the selected feature space. For subsequent probability fusion, the SVC decision outputs are converted into class-probability estimates using an appropriate calibration procedure.

Random Forest

RF is a bagging-based ensemble that combines multiple DTs trained on bootstrap samples of the training data. At each split, only a random subset of the available features is considered, which reduces correlation among the trees and improves generalisation [38]. The class probability is estimated by averaging the predictions of the individual trees using Equation (6).

$$\hat{P}_{RF}(y = 1 \mid x) = \frac{1}{T}\sum_{t=1}^{T} \hat{P}_t(y = 1 \mid x) \tag{6}$$

where T is the number of trees. RF contributes robustness to nonlinear interactions and reduces the variance associated with a single DT.

Decision Tree

A DT recursively partitions the feature space into increasingly homogeneous regions. At each internal node, the split is selected by minimising an impurity measure [22], taken as the Gini index expressed in Equation (7).

$$G(S) = 1 - \sum_{c=1}^{C} p_c^2 \tag{7}$$

where p_c is the proportion of samples belonging to Class C within node S. The resulting rule-based structure provides interpretable decision boundaries and can capture nonlinear relationships between the selected electrical features and the corresponding fault target.

AdaBoost

AdaBoost constructs a strong classifier by sequentially combining weak learners. Samples that are misclassified at one iteration receive increased weights, causing subsequent learners to focus more strongly on difficult observations [39-41]. The final ensemble decision is given by Equation (8).

$$F(x) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m h_m(x) \right) \tag{8}$$

where $h_m(x)$ is the prediction of the *m*th weak learner and α_m represents its contribution to the ensemble.

Gradient Boosting Machine

Gradient Boosting Machine constructs an additive model by sequentially fitting weak learners to the residual errors produced by the preceding ensemble [42]. Its iterative update is expressed in Equation (9).

$$F_g(x) = F_{g-1}(x) + \nu h_g(x) \tag{9}$$

where $h_g(x)$ is the learner introduced at boosting iteration g, and v is the learning rate. In this study, $v = 0.1$ is used to limit the contribution of each learner and reduce the risk of overfitting. The accumulated decision score is transformed into a class-probability estimate for use in the subsequent probability-fusion stage.

Gaussian Naive Bayes

Gaussian Naive Bayes (GNB) is a probabilistic classifier derived from Bayes' theorem and assumes conditional independence among the selected features given the class label [43]. The predicted class is given in Equation (10).

$$\hat{y} = \arg\max_{y} P(y) \prod_{r=1}^{p} P(x_r \mid y) \tag{10}$$

where each conditional feature distribution is modelled as Gaussian. Despite its simplifying independence assumption, GNB is computationally efficient and often performs competitively when the available training data are limited.

The seven classifiers collectively provide instance-based, margin-based, tree-based, bagging, boosting, and probabilistic learning mechanisms. This heterogeneity is important because no single classifier is expected to provide the most accurate decision boundary for every fault target [44]. Instead, AMTEC exploits differences in classifier behaviour and retains the strongest-performing models dynamically in Stage 3. After training, the classification performance of each classifier-target pair is quantified on an internal validation split of the training partition using the F1-score, defined in Equation (11) [8].

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN} \tag{11}$$

where $TP, FP,$ and FN denote the numbers of true-positive, false-positive, and false-negative predictions, respectively. The F1-score is adopted because it jointly reflects fault-detection sensitivity and false-alarm behaviour and is therefore more informative than accuracy when the target classes are imbalanced.

The F1-scores used for classifier ranking and weight assignment are obtained from an internal validation set rather than from the same samples used to fit the models. These scores are subsequently used in Stage 3 to identify the three highest-performing classifiers for each fault target and in Stage 4 to derive their normalised contribution weights. The held-out test labels remain completely excluded from classifier selection, weighting, and meta-learner construction.

Stage 3: Dynamic top-three classifier selection

Following Stage 2, the seven trained base classifiers are ranked independently for each fault target according to their validation F1-scores. Rather than combining all classifiers, AMTEC dynamically selects the three highest-performing classifiers for each target, thereby retaining only the most reliable learners while reducing the influence of weaker models. The selected classifier subset is defined in Equation (12).

$$C_{\text{top}} = \text{Top-}k(F1_j), \quad k = 3 \tag{12}$$

where C_{top} denotes the set of the three classifiers with the highest F1-scores for fault target j, and $F1_j$ is the vector of F1-scores obtained from the seven base classifiers for that target.

Unlike conventional ensemble methods that assign equal importance to all constituent classifiers, the proposed dynamic selection strategy adapts the ensemble composition to the characteristics of each fault target. Consequently, only the most discriminative classifiers are propagated to the subsequent probability aggregation and meta-learning stages, improving ensemble robustness while reducing the contribution of poorly performing models [45].

Stage 4: F1-Weighted soft probability aggregation

The three classifiers selected in Stage 3 contribute unequally to the ensemble according to their classification performance. Accordingly, their training F1-scores are normalised to produce weights that sum to unity. This is expressed in Equation (13) [27].

$$w_i = \frac{F1_i}{\sum_{k=1}^{3} F1_k}, \quad \sum_{i=1}^{3} w_i = 1 \tag{13}$$

where w_i is the normalised weight assigned to the ith selected classifier based on its training F1-score. The weighted ensemble probability for each sample is then obtained by combining the probability estimates of the three selected classifiers using Equation (14).

$$P_{\text{ensemble}}(x) = \sum_{i=1}^{3} w_i P_i(x) \tag{14}$$

where $P_i(x)$ denotes the posterior probability produced by the ith classifier for a sample. By assigning larger weights to higher-performing classifiers, the proposed aggregation strategy allows more reliable models to exert greater influence while reducing the contribution of weaker classifiers. The aggregated probability serves as one of the four meta-features that are subsequently supplied to the RF meta-learner in Stage 5 for final fault prediction, alongside the three individual selected-classifier probabilities defined in Equation (15).

Stage 5: RF meta-learner stacking

The outputs of the three classifiers selected in Stage 3 are combined with the F1-weighted ensemble probability generated in Stage 4 to form the meta-feature vector for each sample in Equation (15).

$$z(x) = [P_1(x), P_2(x), P_3(x), P_{\text{ensemble}}(x)] \tag{15}$$

where $P_i(x)$ is the fault-class probability produced by the ith selected classifier, and $P_{\text{ensemble}}(x)$ is the corresponding F1-weighted aggregated probability. For N samples, these vectors form the meta-feature matrix in Equation (16):

$$Z \in \mathbb{R}^{N\times 4}. \tag{16}$$

A 75-tree RF is then trained independently for each fault target using the corresponding meta-feature matrix. The final binary prediction for target is expressed in Equation (17).

$$\hat{y}_j = f_{RF}(Z_j; \theta_j) \tag{17}$$

where $f_{RF}(\cdot)$ denotes the RF meta-learner and θ_j represents the parameters learned for fault target j. The RF was selected because bootstrap aggregation reduces prediction variance, while the low-dimensional meta-feature space limits model complexity under restricted training-data conditions. The four independently generated binary outputs are subsequently combined to form the final multi-target prediction vector $\hat{Y} = [\hat{y}_G, \hat{y}_C, \hat{y}_B, \hat{y}_A]$.

To prevent information leakage, the meta-learner must be trained using out-of-fold or validation-set probability estimates generated from the training data. The held-out test samples are used only after the meta-learner has been fitted. Unlike the original MTEC framework, which applies equal-weight majority voting, AMTEC learns a nonlinear combination of classifier probabilities and performance-weighted ensemble information. This enables the final decision rule to adapt to the relative reliability and interaction of the selected classifiers for each fault target.

Experimental setup and performance assessment

Experimental Design

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

A quantitative experimental design was adopted to evaluate the proposed AMTEC framework under progressively constrained training-data conditions. The complete architecture described in the 'Materials and Methods' section was implemented in MATLAB R2023b using the Statistics and Machine Learning Toolbox. Three experimental phases were considered, representing standard, limited-data, and extreme-data conditions, with the limited-data condition (Phase 2) designed to replicate the 231-sample benchmark protocol established in the MTEC study in [8]. Within each phase, the same dataset, preprocessing procedure, classifier pool, target definitions, and evaluation metrics were maintained, and the same random seed was used for all classifiers within a phase to ensure a common train/test partition. Phase 1 and Phase 2 share the same seed (rng(1)); Phase 3 uses a different seed (rng(100)) to generate an independent permutation, as described in the hyperparameter configuration section.

Transmission Line Model and Dataset

The data used in this study are the publicly available Kaggle dataset [46], which was generated from a MATLAB/Simulink model of a 30-km, three-phase overhead transmission line. The network topology described in [46] and used in [8] was reconstructed by the authors in MATLAB R2023b for illustration only, as shown in Figure *2*; no data were generated from this reconstruction, and all experiments use the original Kaggle observations.

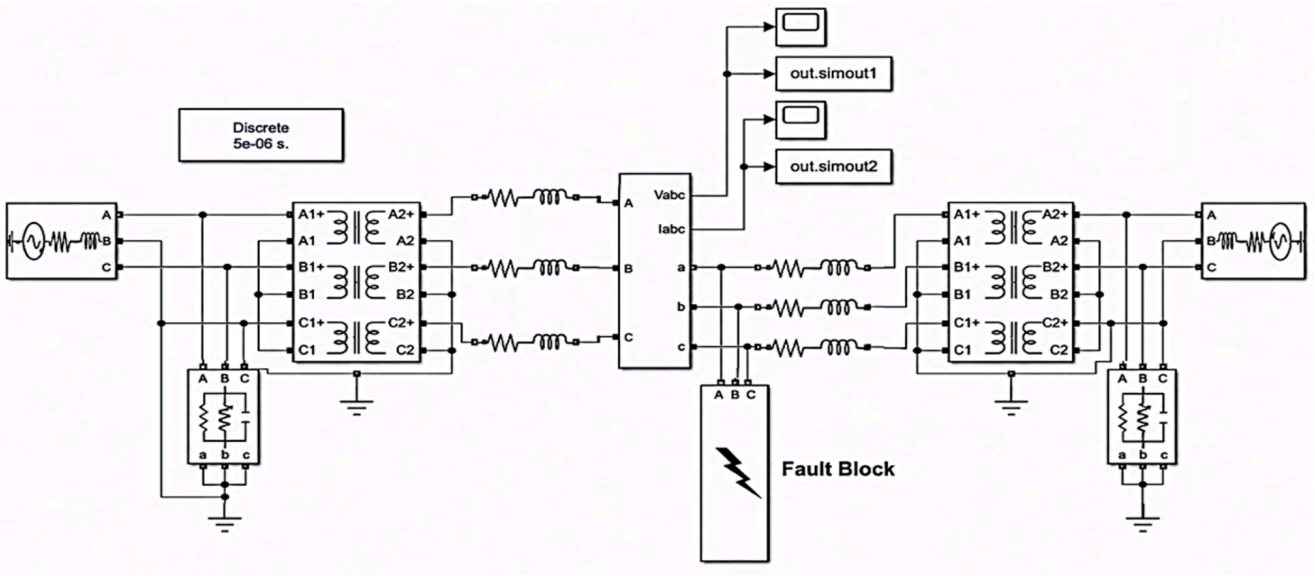


FIGURE 2: Reconstructed MATLAB/Simulink model of the transmission network

Each observation is characterised by three instantaneous phase-current measurements and three instantaneous phase-voltage measurements, forming the six-dimensional input feature vector defined in Equation (1), where I_a, I_b, and I_c denote the phase currents, while V_a, V_b, and V_c represent the corresponding phase voltages. The associated multi-target output vector is defined in Equation (18) [47].

$$Y_i = [y_{iG},\ y_{iC},\ y_{iB},\ y_{iA}] \in \{0,1\}^4 \tag{18}$$

where G indicates ground involvement and A, B, and C indicate the involvement of the respective phases. Each output assumes a binary value, with 1 denoting the presence of the corresponding fault condition and 0 denoting its absence.

The dataset contains 7,861 observations representing normal operation and five transmission line fault conditions: LG, LL, LLG, LLL, and three-phase-to-ground faults. Their corresponding binary label combinations and sample distributions are presented in Table *1*.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

No.	Fault Type	Code [GCBA]	Definition	Samples
1	Normal	0000	No fault	2,365
2	AG	1001	Single line-to-ground	1,129
3	BC	0110	Line-to-line	1,004
4	ABG	1011	Double line-to-ground	1,134
5	ABC	0111	Three-phase	1,096
6	ABCG	1111	Three-phase-to-ground	1,133
Total				7,861

TABLE 1: Dataset composition and fault type distribution

Data Preprocessing and Exploratory Analysis

Before model development, the dataset was subjected to feature normalisation, binary-label verification, missing-value analysis, outlier assessment, and class-distribution analysis. Because the current and voltage variables have different numerical ranges, each feature was transformed using the column-wise min-max normalisation expressed in Equation (19) [12].

$$x'_{ir} = \frac{x_{ir} - x_{r,\min}}{x_{r,\max} - x_{r,\min} + \varepsilon} \tag{19}$$

where x_{ir} is the original value of the feature for sample i, x'_{ir} is its normalised value, $x_{r,\min}$ and $x_{r,\max}$ are the minimum and maximum values of the feature obtained exclusively from the training partition, and $\varepsilon = 10^{-8}$ prevents division by zero when a feature has a constant value.

The resulting training-derived parameters were subsequently applied unchanged to the corresponding test partition to prevent data leakage. The output labels were encoded as binary integers according to Equation (18).

The null-value sparsity matrix presented in Figure *3* confirms that no missing values exist across the six input features and four target labels; therefore, no imputation procedure was required.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

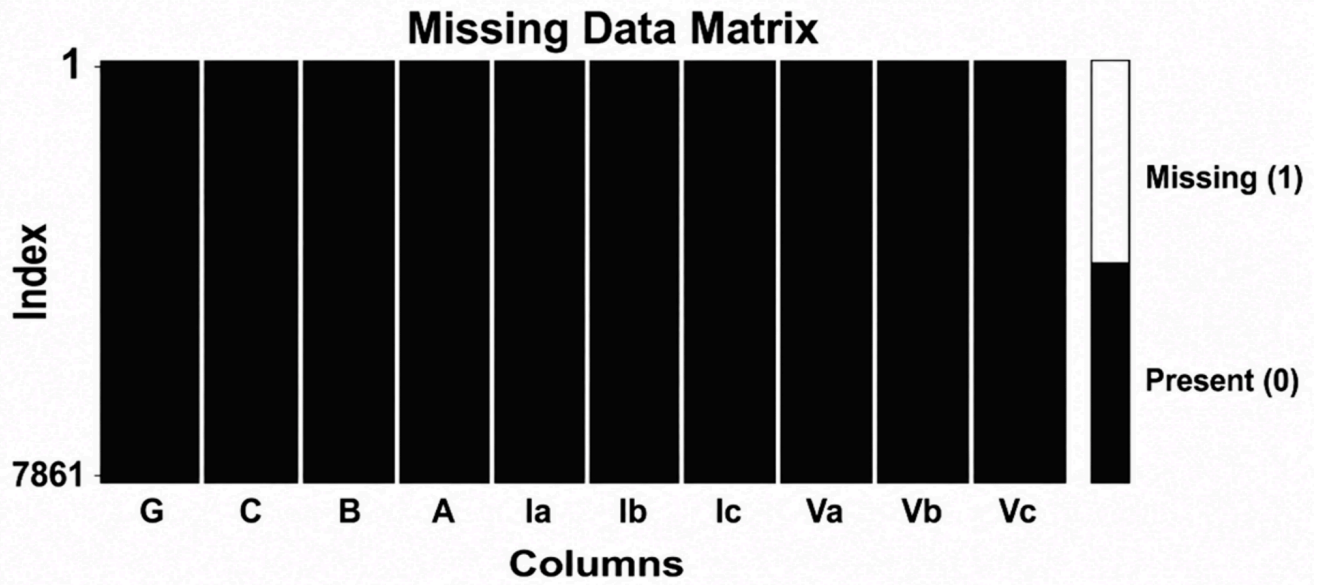


FIGURE 3: Null-value sparsity matrix

The boxplots shown in Figure *4* reveal observations beyond the conventional whisker limits, particularly in the phase-current measurements. These values were retained because they represent physically meaningful fault-induced current excursions rather than erroneous observations.

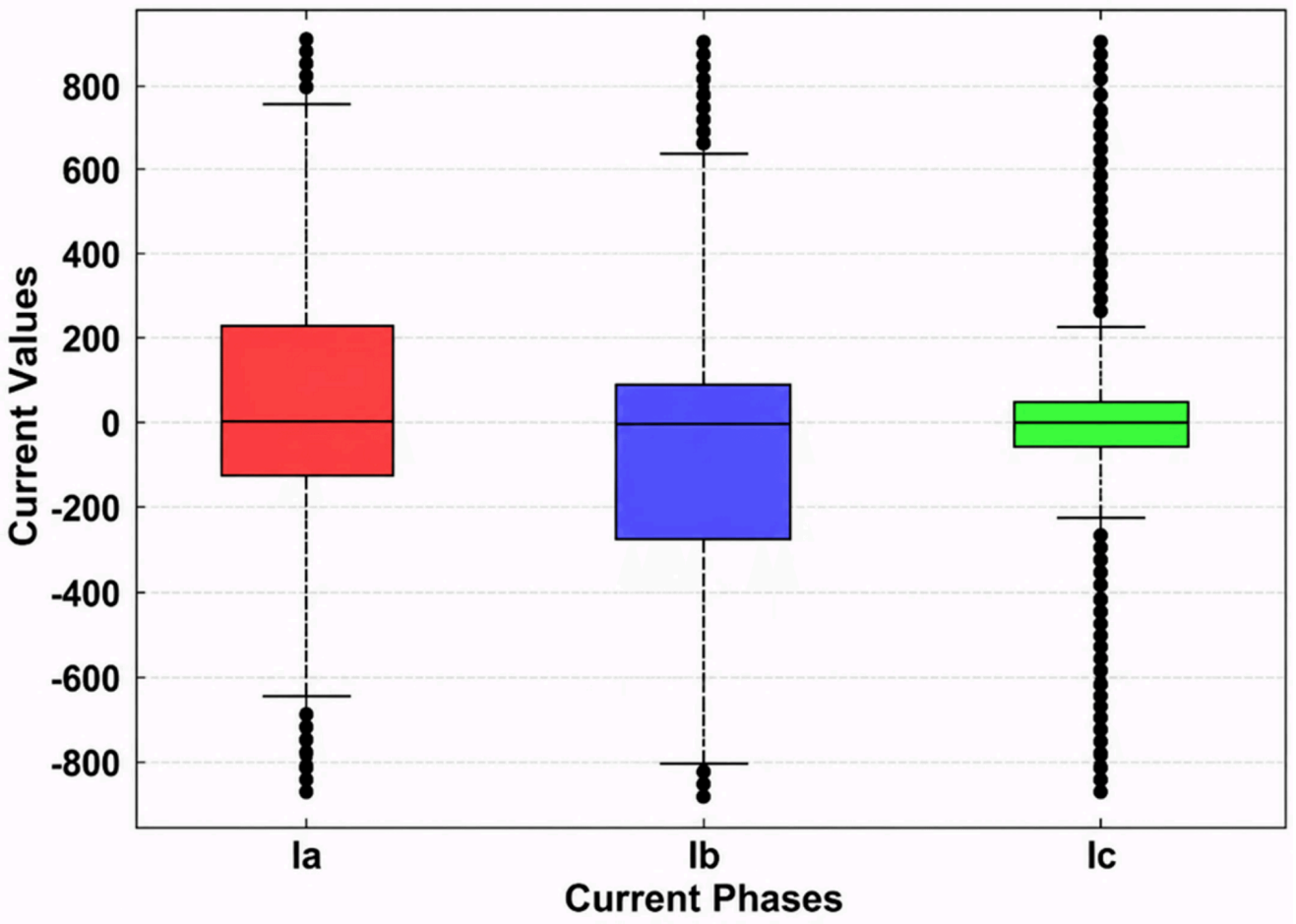


FIGURE 4: Box plots for outlier profiling of electrical current

The target distributions presented in Figure *5* show positive-label proportions of 43.2% for G, 41.1% for C, 55.6% for B, and 57.1% for A. The resulting distributions indicate only moderate target-level imbalance and do not justify additional resampling or synthetic data-generation procedures.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

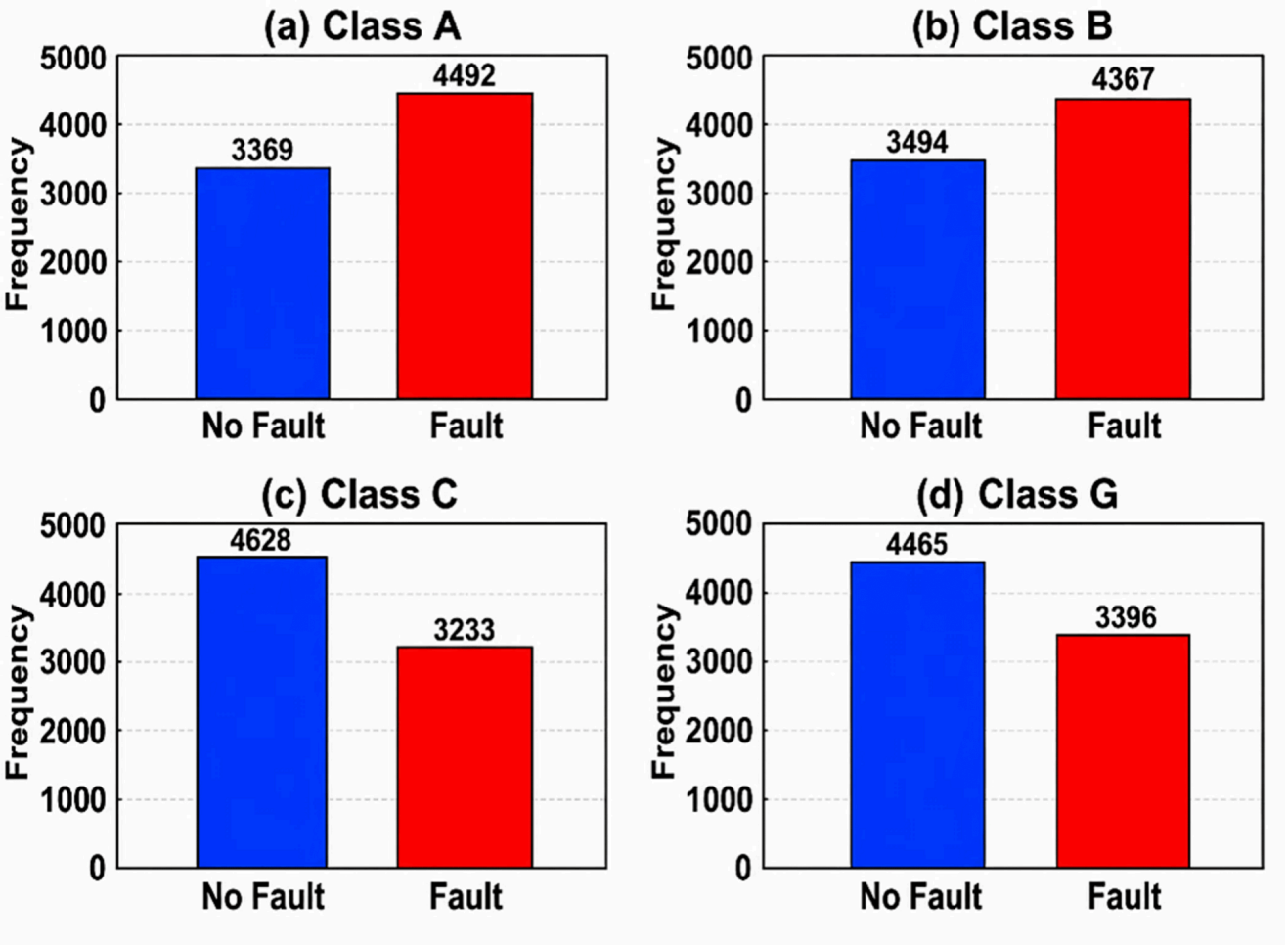


FIGURE 5: Categorical distribution of fault samples across the dataset

Taken together, Figures *3-5* establish that the dataset requires no imputation, no outlier removal, and no resampling before model development, so the preprocessing stage reduces to the training-derived min-max normalisation of Equation (20). Together, these checks indicate that the MATLAB data-handling stage reproduces the data visualisation and preprocessing conditions of the Python study in [8], so the comparison in subsequent phases is not affected by differences in data preparation.

Experimental Data Partitioning

Three experimental phases were defined to assess AMTEC under progressively increasing levels of training-data scarcity. Phase 1 employed a stratified 80:20 holdout partition, producing 6,289 training samples and 1,572 test samples, to validate that the baseline classifier results matched those originally reported by MTEC in [8] before the limited-data and extreme-scarcity phases were conducted. This phase represented the standard-data condition. Phase 2 used only 231 training samples, corresponding to approximately of the complete dataset, while the remaining 7,630 samples were reserved for testing. This phase represented the limited-data benchmark condition. Phase 3 further reduced the training set to 100 samples, equivalent to approximately of the dataset, while 7,761 samples were retained for testing. This phase assessed the robustness of AMTEC under extreme training-data scarcity. An independent random permutation was used for Phase 3 to avoid direct duplication of the Phase 2 training partition. Because Phase 3 uses an independent permutation and seed, the Phase 3 training set is not a nested subset of the Phase 2 training set, and the two phases are evaluated on different held-out partitions. Between-phase comparisons therefore reflect the combined effect of reduced

training size, a different training sample draw, and a different test partition, rather than the effect of training size alone. Within-phase comparisons, in which all classifiers share a common partition and seed, are unaffected. The complete splitting strategy is presented in Table 2.

Phase	Training Samples	Test Samples	Purpose
Phase 1 (Standard)	6,289 (80%)	1,572 (20%)	Full dataset baseline; MATLAB–Python validation
Phase 2 (Limited)	231 (~2.9%)	7,630	MTEC benchmark replication; primary evaluation
Phase 3 (Extreme)	100 (~1.3%)	7,761	First multi-target ensemble evaluation at 100-sample threshold

TABLE 2: Experimental data splitting strategy

MTEC, Multi-Target Ensemble Classifier

Implementation Details and Hyperparameter Configuration

The seven base classifiers were configured to align with the Python/scikit-learn benchmark in [8], so that the MATLAB re-implementation reproduces the MTEC baseline results before AMTEC's ensemble extensions are applied. This ensures that AMTEC and MTEC are compared under equivalent classifier configurations, isolating the effect of AMTEC's architectural innovations from any discrepancy introduced by differing toolbox defaults. The complete configuration of each base classifier is summarised in Table 3.

S/No.	Base Classifier	Hyperparameter Configuration
1.	K-nearest neighbour (KNN)	NumNeighbors = 5; Distance = cityblock; DistanceWeight = equal; Standardize = true. All other options at MATLAB defaults.
2.	Support vector classification (SVC)	RBF kernel; BoxConstraint = 0.05097; KernelScale = auto; Standardize = true; decision scores converted to posterior probabilities by Platt scaling (fitPosterior).
3.	Random forest (base classifier)	TreeBagger with 100 trees.
4.	Decision tree	Gini diversity index (gdi); MinLeafSize = 1; MaxNumSplits = 1000; MinParentSize = 1; Prune = off; MergeLeaves = off; exact categorical splits; all predictors considered at each split.

5.	AdaBoost	AdaBoostM1; NumLearningCycles = 200; LearnRate = 0.05; template tree with MaxNumSplits = 10 and MinLeafSize = 5; target-dependent misclassification costs.
6.	Gradient boosting	LogitBoost; NumLearningCycles = 100; LearnRate = 0.1.
7.	Gaussian Naive Bayes	Gaussian kernel-density likelihoods (DistributionNames = kernel, Kernel = normal), Width = 0.0693; no single-Gaussian assumption imposed.

TABLE 3: Summary of base classifier hyperparameter configuration

RBF, Radial Basis Function

The mRMR feature selection was applied independently to each fault target, retaining five of the six electrical measurements for the ground target, four for Phases C and B, and three for Phase A. These counts were chosen on the basis of the mutual-information analysis presented in Figure *9*, which was computed on the Phase 2 training partition, and were then held fixed across phases, while the specific measurements selected for each target are determined at run time by the mRMR criterion of Equation (2) and are not hard-coded. Because the selection is re-run on each random partition, the exact subset of measurements retained for a given target can vary slightly across partitions; the counts, however, are fixed, and the reported results are based on whatever subset the criterion selected for the partition under evaluation. The redundancy weight was set to $\lambda = 0.3$ throughout.

A single fixed-seed train/test holdout split was used for each experimental phase, with the Phase 2 training size set to 231 samples to match the MTEC benchmark condition. In Phase 1, the split was a stratified 80:20 holdout stratified to Class A; in Phase 2, 231 samples were assigned to training and the remaining 7,630 to testing; and in Phase 3, an independent random permutation was used to assign 100 samples to training and the remaining 7,761 to testing. The same fixed test partition was used for AMTEC and for all seven baseline classifiers within each phase, so that every reported comparison is made on a common held-out set. Any bias that a single split introduces into absolute performance estimates therefore applies to all models alike and does not distort their relative ranking. To support reproducibility, the MATLAB random number generator was deliberately seeded: rng(1) was used for Phase 1 and Phase 2 to guarantee identical holdout partitions and sample permutations across executions, while a separate seed, rng(100), was used for Phase 3 to ensure an independent permutation that avoids any unintended overlap with the earlier phases. The results are therefore reproducible, but they represent a single partition, and their sampling variability is not quantified; repeating each phase over multiple seeds is identified as a recommendation for future work.

Implementation Procedure

The computational implementation followed the AMTEC workflow illustrated in Figure *6*. For each experimental phase, the training and test partitions were first transformed using the training-derived normalisation parameters defined in Equation (19). The four target variables specified in Equation (18) were then processed independently. Internal validation predictions generated from the training partition were used to calculate classifier F1-scores, rank the candidate classifiers, estimate the probability-aggregation weights, and construct the meta-features required by the RF meta-learners.

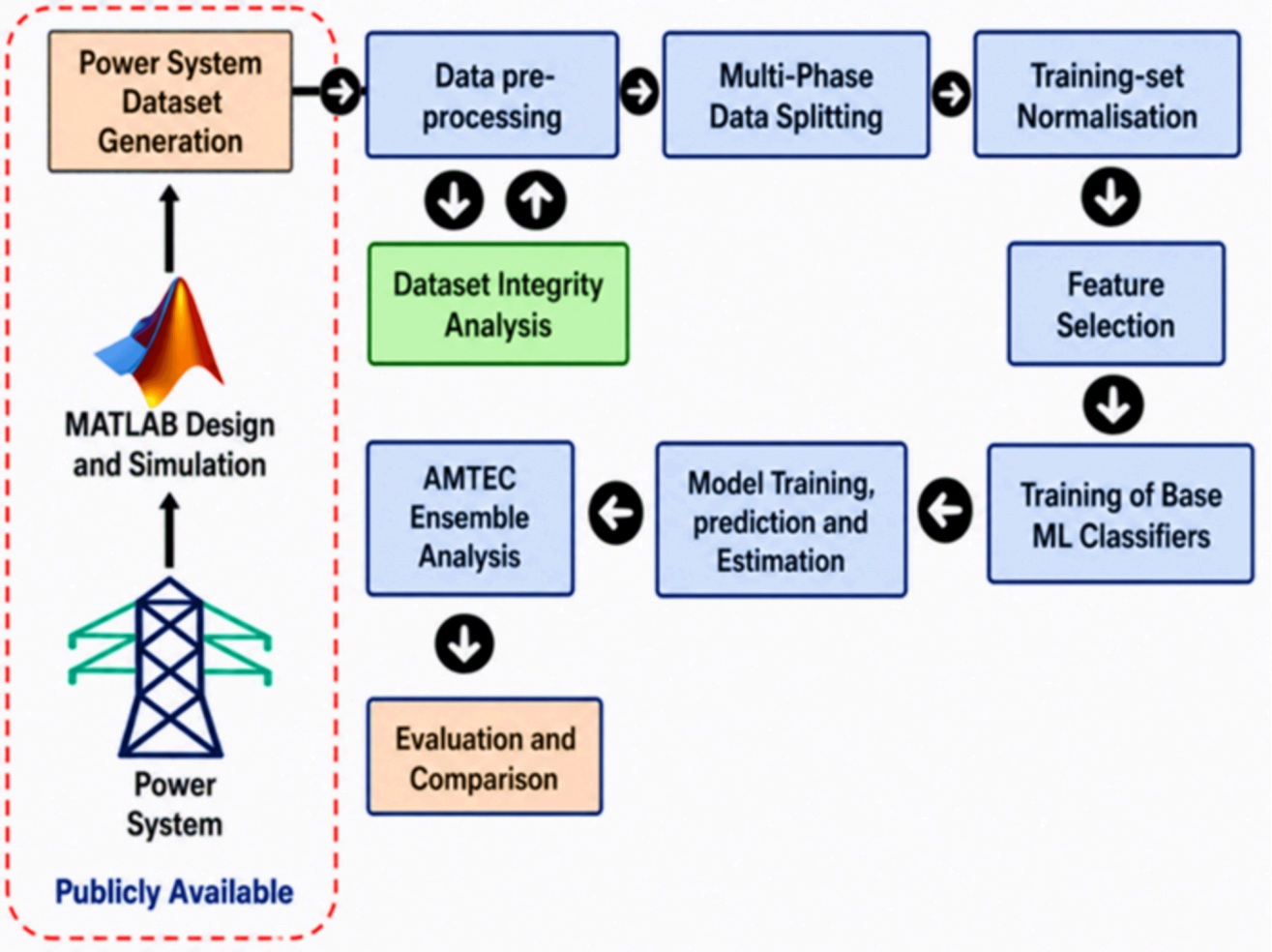


FIGURE 6: Computational workflow of the AMTEC experiments

AMTEC, Adaptive Multi-Target Ensemble Classifier; ML, Machine Learning

After model development, the fitted target-specific feature selectors, base classifiers, selected classifier subsets, aggregation weights, and meta-learners were applied to the corresponding held-out test partition. The final four-dimensional prediction for the sample *i* was formed as Equation (20).

$$\hat{\mathbf{Y}}_i = \left[\hat{y}_{iG},\ \hat{y}_{iC},\ \hat{y}_{iB},\ \hat{y}_{iA}\right] \tag{20}$$

where $\hat{y}_{ij} \in \{0, 1\}$ denotes the predicted binary output for target $j \in G, C, B, A$. The test labels were excluded from feature selection, classifier ranking, weight estimation, meta-feature construction, and meta-learner training.

Performance Evaluation Metrics

AMTEC performance was evaluated using strict multi-target overall accuracy together with five target-specific binary classification metrics: accuracy, precision, recall, F1-score, and specificity [48]. The principal evaluation criterion was strict multi-target overall accuracy. Under this measure, a test sample is classified as correct only when all four predicted outputs simultaneously match the corresponding true labels. It is defined using Equation (21).

$$\text{Overall Accuracy} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\left(Y_i = \hat{Y}_i\right) \tag{21}$$

where N is the number of test samples, $\mathbf{Y_i}$ and $\widehat{\mathbf{Y}}_\mathbf{i}$ are the true and predicted multi-target vectors, respectively, and I(·) is an indicator function that returns 1 when the two vectors match exactly and 0 otherwise. This measure is more stringent than individual target accuracy because an incorrect prediction for any one of the four outputs renders the complete sample prediction incorrect.

Accuracy measures the proportion of all predictions, both positive and negative, that a classifier assigns correctly, computed as the sum of true positives and true negatives divided by the total number of samples. In the present multi-target formulation, it is calculated independently for each fault target (G, C, B, and A) as a one-versus-rest binary problem and then macro-averaged across the four targets to yield a single value per classifier. For each target, accuracy was computed using Equation (22) [12]:

$$\mathbf{Accuracy} = \frac{\mathbf{TP} + \mathbf{TN}}{\mathbf{TP} + \mathbf{TN} + \mathbf{FP} + \mathbf{FN}}. \tag{22}$$

Precision is a metric used to assess the effectiveness of a model by determining the frequency with which the model's positive predictions are accurate. A high precision value signifies that when the classifier signals a fault, it is almost invariably accurate, indicating a low false-alarm rate. Precision was calculated using Equation (23) [8]:

$$\mathbf{Precision} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FP}}. \tag{23}$$

Recall, also known as the true positive rate, quantifies the comprehensiveness of a classifier's fault-detection capabilities. A high recall value signifies that the classifier effectively detects most true fault occurrences. Recall was determined using Equation (24) [47]:

$$\mathbf{Recall} = \frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FN}}. \tag{24}$$

The F1-score balances precision and recall via their harmonic mean, providing a single metric that captures the impact of both false positives and false negatives on classifier performance. The preference for the harmonic mean over the arithmetic mean arises from its ability to penalise significant disparities between precision and recall. The F1-score is defined in Equation (11) and is restated here for completeness in Equation (25):

$$\boldsymbol{F1} = \frac{\mathbf{2\,Precision} \times \mathbf{Recall}}{\mathbf{Precision} + \mathbf{Recall}}. \tag{25}$$

Specificity, also referred to as the true negative rate or selectivity, quantifies a classifier's ability to identify correct samples in which the corresponding fault is absent. A high specificity value signifies that the classifier infrequently generates false alarms under healthy (non-fault) operating conditions. Specificity was calculated using Equation (26) [8]:

$$\mathbf{Specificity} = \frac{\mathbf{TN}}{\mathbf{TN} + \mathbf{FP}}. \tag{26}$$

$\boldsymbol{TP}$, $\boldsymbol{TN}$, $\boldsymbol{FP}$, **and** $\boldsymbol{FN}$ in the above equations denote true positives, true negatives, false positives, and false negatives, respectively. Throughout this paper, metric values are reported to four decimal places (equivalently, two decimal places when expressed as a percentage) in tables and in-text summaries; higher-precision values (up to six decimal places) are used only where needed to make an underlying percentage-point difference traceable to its source figures, and in all such cases the higher-precision value rounds consistently to the four-decimal value reported elsewhere.

Benchmarking Strategy

The proposed AMTEC framework was benchmarked against the published MTEC results under comparable experimental settings. Within each phase, AMTEC and all seven baseline classifiers used the same dataset, training and test partitions, classifier pool, target representation, and metric definitions, so that comparisons among these eight methods are controlled.

The benchmark strategy used in this study is based on a previously published result rather than a re-implemented, matched-codebase reproduction. The MTEC baseline accuracy of 0.8292 was adopted directly from [8], which reports results on the same publicly available Kaggle dataset [46], using the same seven-classifier pool, and the same 231-sample training constraint. The original MTEC implementation is in Python/scikit-learn, and its source code is not publicly available for direct re-implementation in the MATLAB environment used for AMTEC.

The Phase 3 experiment extended the assessment to a 100-sample training condition, thereby examining multi-target fault classification under a more severe level of data scarcity. In all phases, the held-out test partitions were used exclusively for final performance assessment, while classifier selection, F1-based weighting, and meta-learner construction were performed using only information obtained from the corresponding training partitions.

Because the original MTEC framework was implemented in Python/scikit-learn while AMTEC was implemented in MATLAB, the comparison was interpreted in terms of predictive performance rather than exact software-level numerical equivalence. To ensure consistency across all reported results, Phase 1, Phase 2, and Phase 3 performance metrics (Tables *4*-*9*) are reported to six decimal places as directly computed, while the summary AMTEC-versus-MTEC comparison (Table *10*) is reported to four decimal places, consistent with the precision at which the MTEC reference value was adopted. When values are referenced in the text or converted to percentage form, the tabulated value is first rounded to four decimal places and then multiplied by 100 and rounded to two decimal places; percentage-point (pp) differences cited throughout the results and discussion are calculated as the direct arithmetic difference between two such rounded percentage values, distinct from relative percentage change, and both conventions are applied uniformly throughout the manuscript.

Results And Discussion

The results reported below were obtained under the experimental setup described in the 'Materials and Methods' section: the AMTEC framework was implemented in MATLAB R2023b using the Statistics and Machine Learning Toolbox and evaluated on the 7,861-sample transmission line fault dataset described in Table *1*, under the three progressively constrained training-data conditions described in the 'Experimental Data Partitioning' subsection (Phase 1: 6,289 training samples; Phase 2: 231 training samples; Phase 3: 100 training samples). In Phases 2 and 3, AMTEC was benchmarked against seven conventional classifiers (KNN, SVC, RF, DT, AdaBoost, gradient boosting, and GNB). For Phase 2 (the 231-sample condition), AMTEC was additionally benchmarked against the published MTEC baseline. Performance was assessed using strict multi-target overall accuracy, and target-specific metrics are reported for the targets discussed in each phase: Class A for Phase 1, all four targets for Phase 2, and Class G for Phase 3.

Phase 1 results: Standard 80/20 split

Table *4* presents the per-target Class A evaluation metrics for Phase 1, together with overall accuracy. Overall accuracy ranged from 0.6667 (GNB) to 0.8925 (DT), in the order DT, RF (0.8836), AdaBoost (0.8632), gradient boosting (0.8391), KNN (0.8168), SVC (0.7595), and GNB (0.6667). Near-perfect Class A F1-scores (≥0.9983) were achieved by RF, DT, AdaBoost, and gradient boosting. This overall accuracy ranking is the same as that of the Python implementations of the same seven classifiers in [8]. The largest differences from the values reported in [8] were for KNN (+0.0062) and DT (−0.0071), and both classifiers kept their respective ranks. These results are consistent with a correct MATLAB implementation of the baseline classifiers.

Algorithm	Overall Accuracy	A Accuracy	A Precision	A Recall	A F1-Score	A Specificity
KNN	0.816794	0.997455	0.996667	0.998886	0.997775	0.995549
SVC	0.759542	0.956107	0.957001	0.966592	0.961773	0.942136

Random forest	0.883588	1.000000	1.000000	1.000000	1.000000	1.000000
Decision tree	0.892494	0.998728	1.000000	0.997773	0.998885	1.000000
AdaBoost	0.863232	1.000000	1.000000	1.000000	1.000000	1.000000
Gradient boosting	0.839059	0.998092	1.000000	0.996659	0.998327	1.000000
Gaussian NB	0.666667	0.928117	0.934662	0.939866	0.937257	0.912463

TABLE 4: Standard 80/20 split (Class A per-target metrics)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes

Phase 2 results - Limited-data regime (231 training samples)

Overall Accuracy and Class G Performance

Table *5* presents the overall accuracy and Class G per-target metrics for all classifiers under the 231-sample constraint. With one exception, the conventional classifiers showed substantial performance declines compared to Phase 1, with overall accuracy drops of about 4-17 pp. The largest decline was for DT (17.0 pp), followed by KNN (10.5 pp), RF (7.7 pp), AdaBoost (7.3 pp), gradient boosting (7.2 pp), and SVC (4.0 pp). GNB was the sole exception: although it remained the weakest classifier overall (0.7105), its accuracy rose by 4.4 pp (0.6667-0.7105) relative to Phase 1, reflecting the reduced sensitivity of its low-parameter, conditional-independence model to the smaller sample size. Among baselines, RF attained the highest overall accuracy at 0.8063, followed by AdaBoost (0.7906) and gradient boosting (0.7667). KNN and GNB exhibited the lowest overall accuracies at 0.7115 and 0.7105, respectively, a reversal of Phase 1 rankings that illustrates the redistribution of classifier performance under reduced data conditions. AMTEC achieved an overall accuracy of 0.8902, representing improvements of 8.39 pp over the best individual classifier (RF) and 6.10 pp over the MTEC baseline of 0.8292. For Class G, AMTEC's specificity (0.9120) exceeded every baseline, from 5.07 pp above RF (0.8613) to 10.35 pp above KNN (0.8085), reducing false ground-fault alarms.

Algorithm	Overall Accuracy	Accuracy	Precision	Recall	F1-Score	Specificity
KNN	0.711533	0.831324	0.773322	0.861440	0.815006	0.808481
SVC	0.719528	0.835125	0.805164	0.814950	0.810027	0.850426
Random forest	0.806291	0.858322	0.823667	0.854452	0.838777	0.861258
Decision tree	0.722936	0.834993	0.788800	0.843209	0.815098	0.828762
AdaBoost	0.790564	0.845740	0.786450	0.881799	0.831400	0.818391
Gradient boosting	0.766710	0.847444	0.795828	0.869341	0.830961	0.830837
Gaussian NB	0.710485	0.813499	0.781325	0.788210	0.784753	0.832680

AMTEC (proposed model)	0.890170	0.892792	0.881989	0.867517	0.874694	0.911961

TABLE 5: Overall accuracy and Class G per-target metrics (231 training samples)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Class C Per-Target Performance (231 Samples)

Table 6 presents Class C metrics for Phase 2. AMTEC achieved near-perfect classification: accuracy 0.9995, precision 0.9997, recall 0.9990, F1-score 0.9994, and specificity 0.9998. The best baseline (RF) achieved Phase C accuracy of 0.9661 and an F1-score of 0.9570. AMTEC's F1-score improvement of 4.24 pp over the best baseline reflects the combined effect of mRMR feature selection identifying the most relevant Phase C current features and the dynamic top-three ensemble composition concentrating the ensemble around classifiers that distinguish Phase C voltage distortions most reliably.

Algorithm	Accuracy	Precision	Recall	F1-Score	Specificity
KNN	0.910616	0.909820	0.868622	0.888744	0.939920
SVC	0.922805	0.925777	0.882972	0.903868	0.950601
Random forest	0.966055	0.998959	0.918367	0.956970	0.999332
Decision tree	0.948493	0.960081	0.912628	0.935753	0.973520
AdaBoost	0.957929	0.975025	0.921237	0.947368	0.983534
Gradient boosting	0.953080	0.946050	0.939413	0.942720	0.962617
Gaussian NB	0.923329	0.898469	0.917092	0.907685	0.927681
AMTEC (proposed model)	0.999476	0.999681	0.999043	0.999362	0.999777

TABLE 6: Class C per-target metrics (231 training samples)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Class B Per-Target Performance (231 Samples)

Table 7 presents Class B metrics. AMTEC achieved an accuracy of 0.9983, perfect precision of 1.0000, recall of 0.9969, an F1-score of 0.9985, and specificity of 1.0000. The improvements over the best baselines (AdaBoost: accuracy 0.9797, F1-score 0.9814; RF: accuracy 0.9755, F1-score 0.9775) are 1.86 pp in accuracy and 1.70 pp in F1-score relative to AdaBoost, and 2.28 pp in accuracy and 2.10 pp in F1-score relative to RF. AMTEC's perfect precision (1.0000) and specificity (1.0000) indicate that no false Phase B alarms were generated on this test partition.

Algorithm	Accuracy	Precision	Recall	F1-Score	Specificity
KNN	0.941284	0.966339	0.926720	0.946115	0.959539
SVC	0.954784	0.984589	0.933318	0.958268	0.981689
Random forest	0.975491	0.998771	0.957116	0.977500	0.998523
Decision tree	0.944037	0.962667	0.935674	0.948978	0.954519
AdaBoost	0.979685	0.997809	0.965598	0.981439	0.997342
Gradient boosting	0.947051	0.952403	0.952403	0.952403	0.940343
Gaussian NB	0.951900	0.998714	0.914703	0.954864	0.998523
AMTEC (proposed model)	0.998296	1.000000	0.996937	0.998466	1.000000

TABLE 7: Class B per-target metrics (231 training samples)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Class A Per-Target Performance (231 Samples)

Table *8* presents the per-target metrics for Class A. Among the baseline classifiers, gradient boosting achieved the highest accuracy (0.9849) and F1-score (0.9870), whereas RF attained an accuracy of 0.9843 and an F1-score of 0.9863. GNB achieved the highest baseline precision (0.9988) but suffered from lower recall (0.9555), indicating a trade-off between false-alarm reduction and fault-detection completeness. AdaBoost and gradient boosting demonstrated strong recall values of 0.9961 and 0.9991, respectively, though their precision lagged GNB.

AMTEC outperformed all baselines, achieving an accuracy of 0.9995, a precision of 0.9991, a recall of 1.0000, and an F1-score of 0.9995. AMTEC is the only configuration that attains near-perfect precision and perfect recall simultaneously: GNB matches its precision (0.9988 vs 0.9991) but at substantially lower recall (0.9555), while gradient boosting matches its recall (0.9991 vs 1.0000) but at substantially lower precision (0.9751). This combination reflects the per-target top-three selection, which retained the classifiers with the highest validation F1-scores for this target, together with the F1-weighted aggregation that assigned them performance-proportional weights. This result has direct safety implications for relay coordination, as undetected Phase A faults can escalate into broader system instability.

Algorithm	Accuracy	Precision	Recall	F1-Score	Specificity
KNN	0.953473	0.930693	0.992426	0.960569	0.901619
SVC	0.948886	0.921932	0.994721	0.956944	0.887870
Random forest	0.984273	0.977893	0.994951	0.986348	0.970058
Decision tree	0.960419	0.944530	0.988754	0.966136	0.922701

AdaBoost	0.983486	0.975500	0.996098	0.985692	0.966697
Gradient boosting	0.984928	0.975134	0.999082	0.986963	0.966086
Gaussian NB	0.973919	0.998800	0.955474	0.976657	0.998472
AMTEC (proposed model)	0.999476	0.999083	1.000000	0.999541	0.998778

TABLE 8: Class A per-target metrics (231 training samples)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Phase 3 results - Extreme-scarcity regime (100 training samples)

Table *9* presents overall accuracy and Class G metrics for Phase 3, the first multi-target ensemble evaluation at the 100-sample threshold. All seven conventional classifiers showed clear accuracy drops of 2.8-6.8 pp from Phase 2. KNN suffered the largest drop (0.7115-0.6433, –6.82 pp) due to sensitivity to the sample-to-dimension ratio. RF maintained the highest baseline accuracy (0.7429), followed by AdaBoost (0.7275). AMTEC achieved 0.8687 overall accuracy, exceeding the best baseline by 12.58 pp and degrading by only 2.15 pp from its Phase 2 performance, compared to 6.34 pp for the best baseline. AMTEC therefore degrades at approximately one-third the rate of RF (2.15 pp vs 6.34 pp, a ratio of 33.9%) as data decrease from 231 to 100 samples, a quantitative indication of architectural robustness within the context of this study.

Critically, AMTEC's 86.87% accuracy at 100 training samples exceeds the best individual classifier accuracy at 231 training samples (RF: 80.63%). This is consistent with the view that ensemble design can partly offset limited training data, subject to the partition caveat noted in the experimental data partitioning subsection: the 100-sample and 231-sample results derive from different permutations and test sets, so the comparison is indicative rather than a like-for-like demonstration.

Algorithm	Overall Accuracy	G Accuracy	G Precision	G Recall	G F1-Score	G Specificity
KNN	0.643345	0.798222	0.753191	0.792301	0.772251	0.802721
SVC	0.688829	0.826311	0.777347	0.837660	0.806377	0.817687
Random forest	0.742945	0.829919	0.812558	0.787825	0.800000	0.861905
Decision tree	0.692308	0.821544	0.810290	0.766040	0.787544	0.863719
AdaBoost	0.727484	0.857879	0.822248	0.855864	0.838719	0.859410
Gradient boosting	0.719495	0.843706	0.814782	0.825724	0.820216	0.857370
Gaussian NB	0.682515	0.784950	0.738379	0.777380	0.757378	0.790703

AMTEC (proposed model)	0.868702	0.885066	0.862577	0.872874	0.867695	0.894331

TABLE 9: Overall accuracy and Class G per-target metrics (100 training samples)

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Graphical analysis of classifier performance

Overall Accuracy Comparison: AMTEC vs Baseline Classifiers

Figure *7* presents a grouped accuracy bar chart comparing the seven conventional classifiers across all three evaluation phases, together with AMTEC for the two data-scarce phases (Phases 2 and 3) in which it was evaluated. Phase 1 is included only to validate the MATLAB baseline implementations against the results reported in [8], so the Phase 1 bars serve as an implementation check rather than as a performance comparison with AMTEC, which was not run at that phase. Three observations follow from the figure: (1) AMTEC achieves the highest accuracy in both Phases 2 and 3, consistently outperforming all other classifiers under data-constrained conditions; (2) the accuracy gap between AMTEC and the best individual classifier widens as data decrease, from 8.39 pp in Phase 2 to 12.58 pp in Phase 3; and (3) in Phase 1, with abundant data, all seven conventional classifiers except GNB perform within 0.133 of the best accuracy, indicating that the baselines converge at high accuracy when training data are plentiful, whereas AMTEC's advantage is evident under data scarcity.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

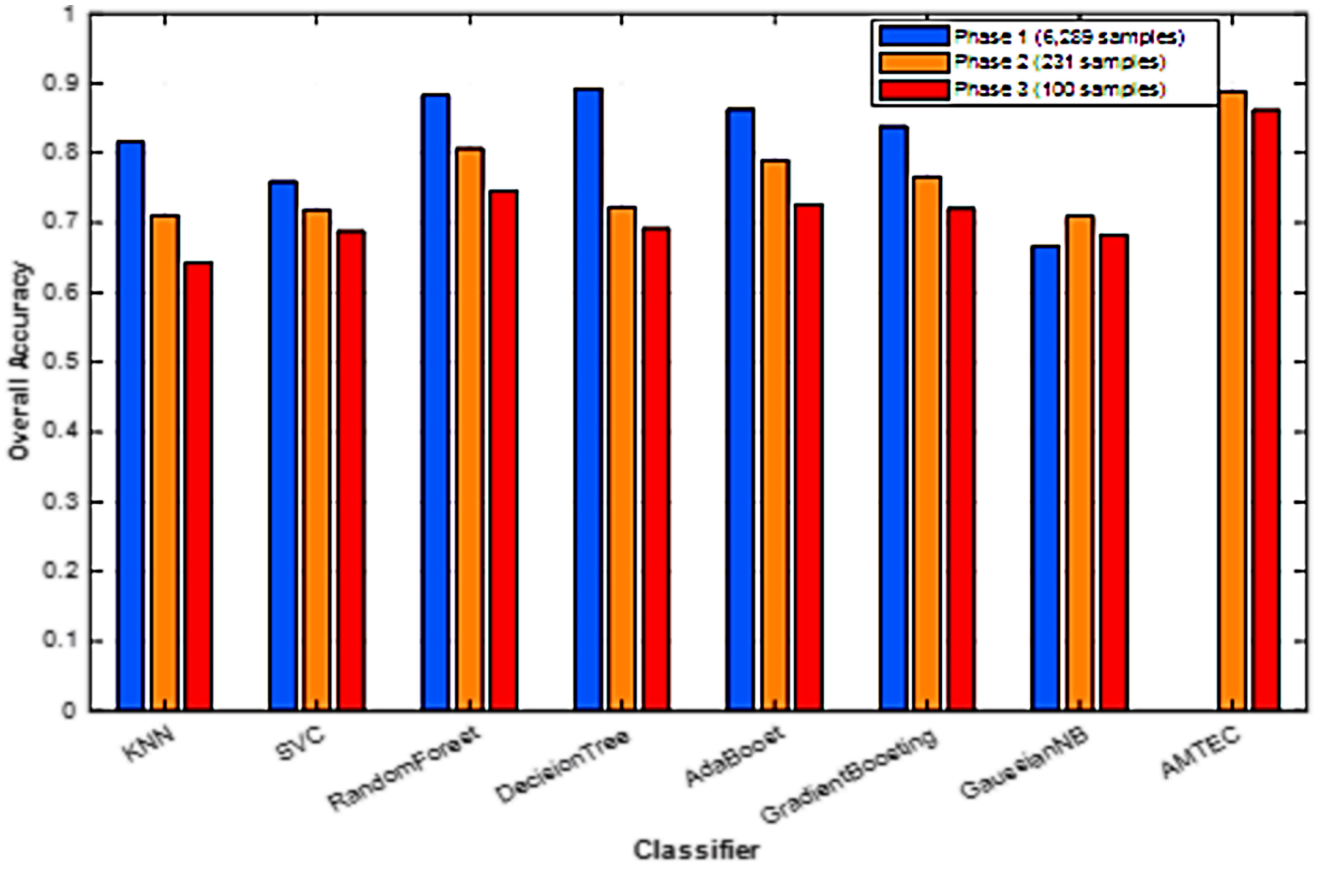


FIGURE 7: Overall accuracy comparison: AMTEC vs baseline classifiers

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Performance Degradation Curves

Figure *8* illustrates performance degradation curves on a logarithmic training-size axis. The seven conventional classifiers are plotted across all three experimental phase points (6,289, 231, and 100 training samples), while AMTEC is plotted only at the two data-scarce phases (231 and 100 samples), for which results were obtained. Six of the seven conventional classifiers (KNN, SVC, RF, DT, AdaBoost, and gradient boosting) exhibit monotonically decreasing accuracy as training size decreases from 6,289 to 100 samples. GNB is the exception: its accuracy rises from 0.6667 at 6,289 samples to 0.7105 at 231 samples before falling to 0.6825 at 100 samples. Within the data-scarce region (100-231 samples), AMTEC's curve consistently lies above all seven individual classifier curves. At 231 samples, AMTEC approaches but does not reach the 90% accuracy threshold (0.8902). Under an independence assumption, 90% strict multi-target accuracy would require approximately 97.4% per-target accuracy (0.9(1/4)), but the target errors in this dataset are not independent, so this figure is indicative rather than a requirement: AMTEC's mean per-target accuracy at 231 samples is 97.25%, yet its strict accuracy is 89.02%, because errors on the ground target dominate the joint outcome. Because AMTEC was not evaluated at the full-data point in the Phase 2 and Phase 3 experiments, the training size at which it would cross 90% cannot be determined from them. Figure *11a*, which used a different random seed, suggests that 90% would be crossed at roughly 300-400 training samples, but this was not evaluated under the headline protocol.

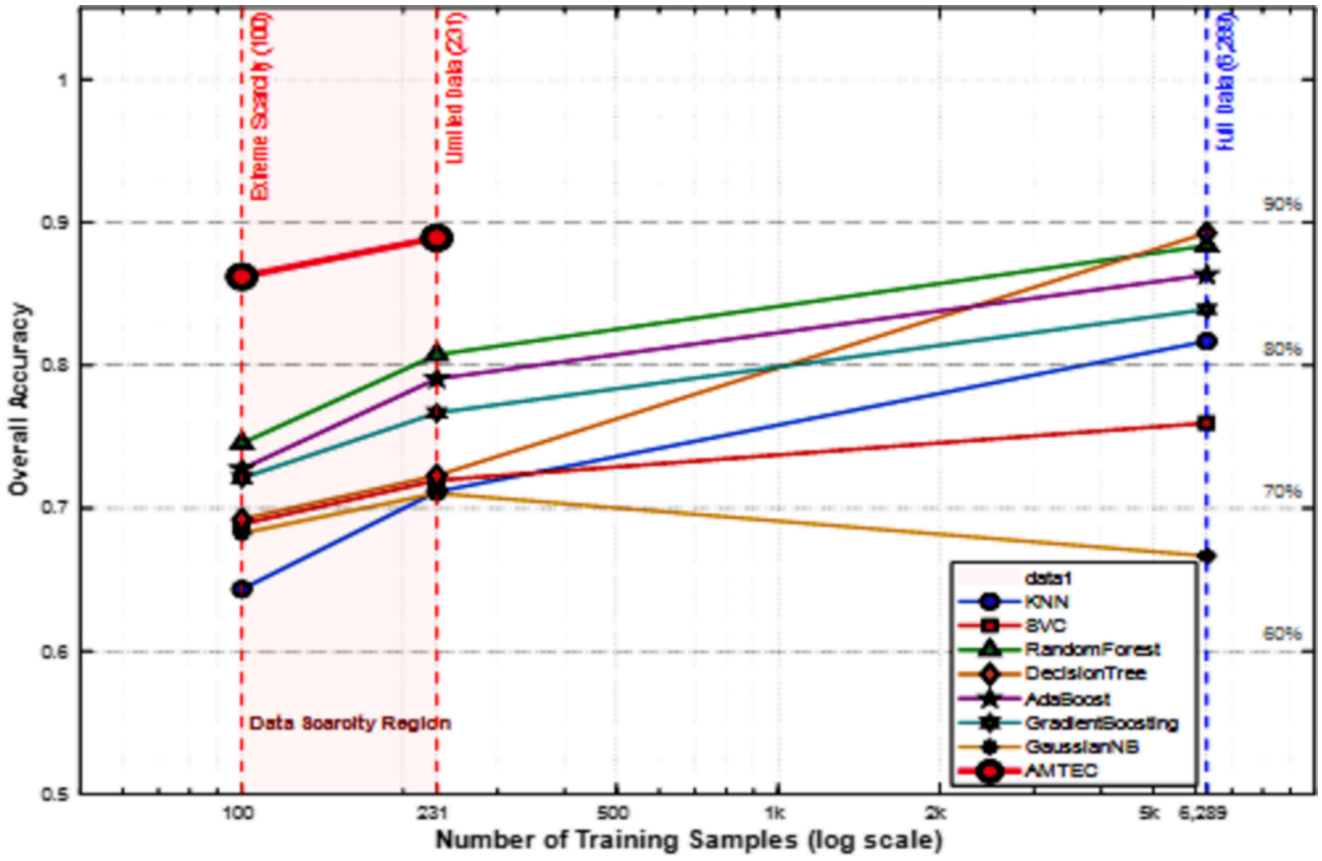


FIGURE 8: Performance degradation under data scarcity

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier

Association Between Electrical Measurements and Fault Targets

Figure 9 reports the normalised mutual-information importance of each of the six electrical measurements (I_a, I_b, I_c, V_a, V_b, V_c) for the four fault targets (*G*, *C*, *B*, *A*), computed on the 231-sample Phase 2 training partition. Within each panel, the six importance values are normalised to sum to unity, so the bars indicate the relative contribution of each measurement to predicting that target. This analysis is descriptive and is distinct from the per-target mRMR feature selection described in Stage 1: the present figure characterises the relevance of each measurement to each target, whereas Stage 1 uses the mRMR criterion of Equation (2) to select the specific subset of measurements retained for each classifier.

For the ground target, the Phase A current I_a is the dominant contributor (0.323), followed by the Phase A voltage V_a (0.277) and the Phase C voltage V_c (0.271), consistent with the need to detect current asymmetry and voltage depression across phases during ground-involved faults. For the Phase C target, the Phase C current I_c dominates (0.382), followed by the Phase B voltage V_b (0.288) and the Phase B current I_b (0.206), indicating that the Phase C current and the adjacent-phase voltage are the most informative measurements for this target. For the Phase B target, the Phase B current I_b carries the largest share (0.325), followed by the Phase C current I_c (0.237), the Phase B voltage V_b (0.216), and the Phase A voltage V_a (0.138), reflecting the dominant role of I_b during Phase B faults and the phase coupling observed in the BC, ABG, ABC, and ABCG conditions in which Phase B is involved. For the Phase A target, the Phase A current I_a is the

dominant contributor (0.428), followed by the Phase A voltage V_a (0.288) and the Phase C voltage V_c (0.241), consistent with the strong dependence of Phase A faults on the I_a-V_a pair and with the phase coupling present in the AG, ABG, ABC, and ABCG conditions in which Phase A is involved.

Overall, the importance profiles vary markedly across targets, with no single measurement dominant for all four; this supports the case for target-specific rather than uniform feature treatment, which is the design adopted by AMTEC's Stage 1 mRMR selection.

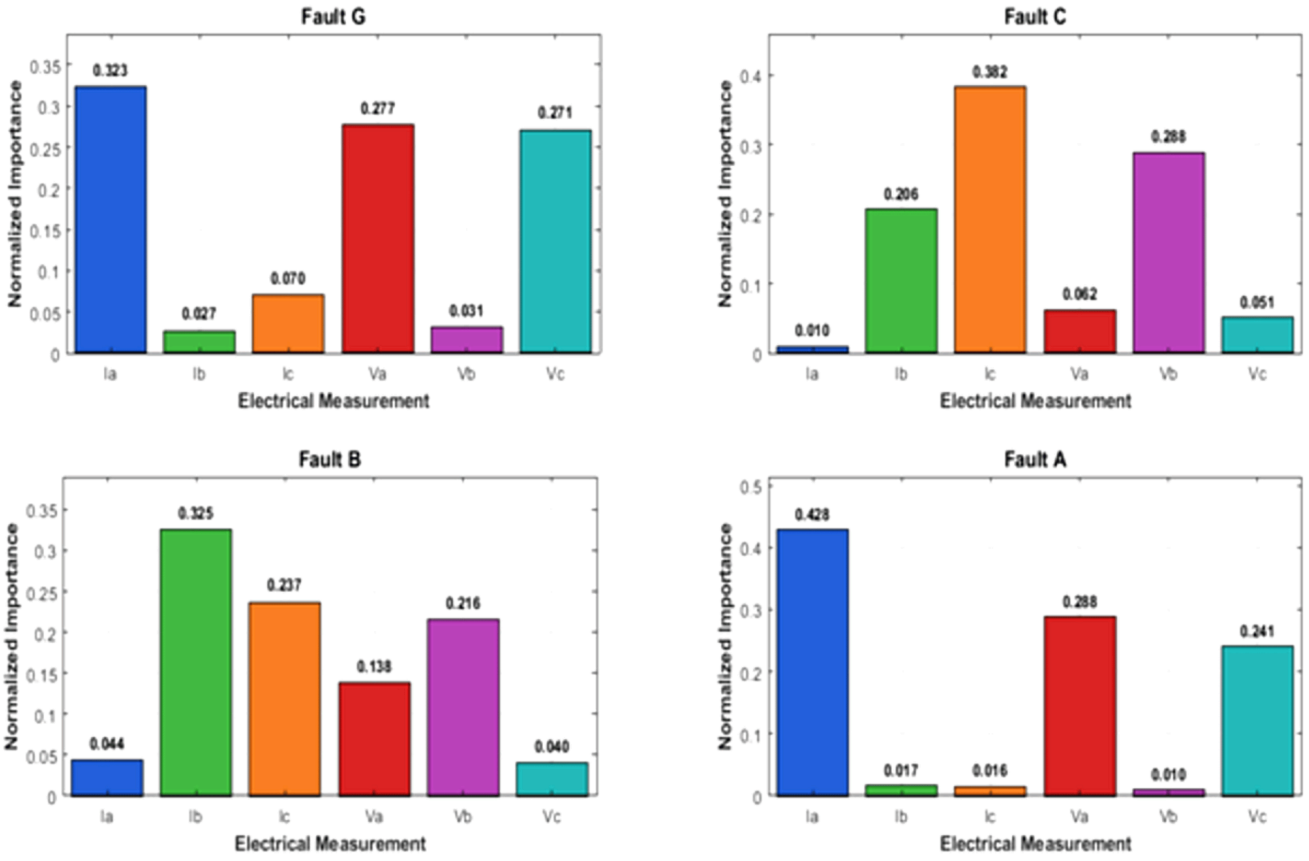


FIGURE 9: Normalised mutual-information importance of the six electrical measurements for each fault target (Phase 2, 231 training samples)

Receiver Operating Characteristic (ROC) Curve Comparison

Figure *10* shows a 2×2 set of ROC curves for the four fault targets under Phase 3 conditions. AMTEC (thicker black line) attains the highest area under the curve (AUC) for all four targets: 0.9723 for G, 0.9996 for C, 1.0000 for B, and 0.9996 for A. Its mean AUC across the four targets is 0.9929, compared with 0.9720 for the next best classifier, RF. The margin is largest for Fault G, where the baseline AUCs range from 0.8428 (DT) to 0.9276 (AdaBoost) and AMTEC exceeds the best baseline by 0.0447. For Faults C and B, AMTEC exceeds the best baseline (RF in both cases) by 0.0110 and 0.0116, respectively. For Fault A, the differences are small: AMTEC (0.9996) is followed closely by RF (0.9955) and GNB (0.9940), consistent with the near-perfect per-target Fault A results. These AUC values are consistent with the benefit of weighted soft voting, probability fusion, and meta-learner stacking, although the present experiments do not isolate the contribution of the individual mechanisms. Because AUC summarises ranking quality over all thresholds, it does not describe performance at the fixed decision threshold used for the classification metrics. In particular, the high AUC for Fault G (0.9723), alongside a recall of 0.8729 for Fault G at 100 samples (Table *9*), suggests that part of the ground-fault recall shortfall may reflect the operating threshold rather than limited class separability.

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

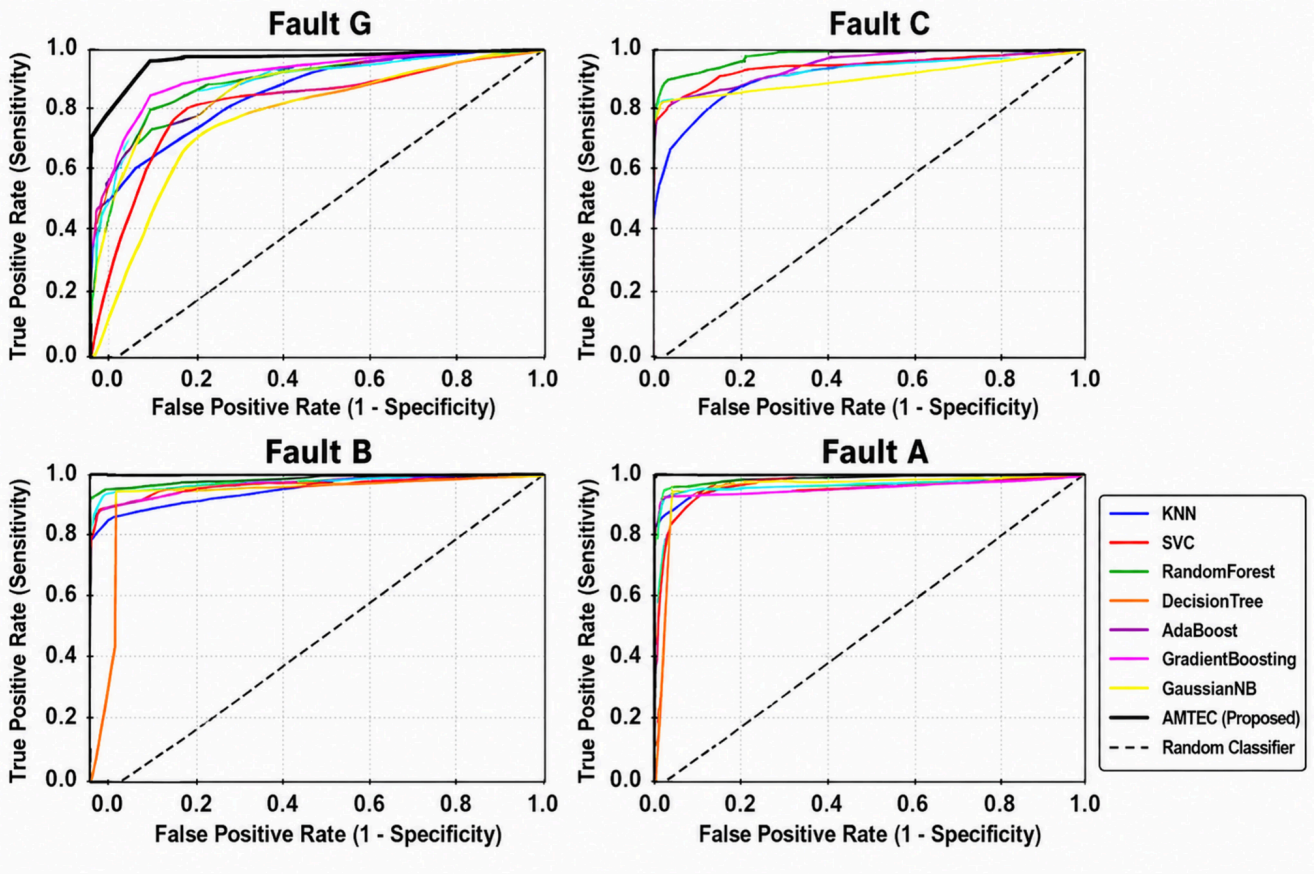


FIGURE 10: ROC curves comparison - AMTEC vs seven baseline classifiers

KNN, K-Nearest Neighbour; SVC, Support Vector Classification; NB, Naive Bayes; AMTEC, Adaptive Multi-Target Ensemble Classifier; ROC, Receiver Operating Characteristic

Sensitivity Analysis of AMTEC

To evaluate the robustness of the proposed AMTEC framework, a sensitivity analysis was conducted as displayed in Figure *11*. The dynamic top-*k*, the mRMR redundancy weight (λ), and meta-learner tree-count sweeps were run on the 231-sample Phase 2 training partition.

A fourth, brief sweep (Figure *11a*) varied the training-set size from 50 to 500 samples, each subset drawn from the full 7,861-sample pool; it is included only to confirm the general trend that reducing training size reduces accuracy, not to characterise the Phase 2 condition. Accuracy rises from 84.2% at 50 samples to 88.6% at 150 samples, then more slowly to 91.0% at 500 samples, consistent with the data-scarce behaviour observed in Phases 2 and 3. Because all four sweeps used a random seed independent of the headline Phase 2 experiment, the default-configuration point in each panel approximates but does not exactly reproduce the 0.8902 reported in Table *5*; the sweeps characterise relative sensitivity rather than the headline result. A comprehensive, multi-seed learning-curve analysis with confidence intervals, extended to larger training sizes, is left to future work.

The top-*k* sensitivity analysis (Figure *11b*) shows a substantial improvement from 0.7776 at $k = 1$ to 0.8877 at $k = 2$, then a flat low *k* range: 0.8900 at $k = 3$, 0.8883 at $k = 4$, 0.8882 at $k = 5$. Accuracy then rises sharply at $k = 6$ (0.9980) and $k = 7$ (0.9997). These values are far above every other setting in the sweep and above the strict accuracy reported in Table *5*. The meta-learner configuration is unchanged at every *k*, and the meta-feature vector remains four-dimensional, so the rise is not accompanied by any change in representational capacity. The only quantity that varies with *k* is the number of top-ranked base classifiers whose probabilities are fused, by *F*1-weighted averaging, into the meta-features. Because the

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3.
DOI https://doi.org/10.7759/s44389-026-00386-3

cause of this jump has not been isolated, the $k = 6$ and $k = 7$ values are treated as unreliable, most likely reflecting information leakage somewhere in the selection, fusion, and meta-learning pipeline, and are excluded from the interpretation of the sweep. $k = 3$ is retained as the selected configuration: accuracy is stable across $k = 2$-5 (0.8877-0.8900), and $k = 3$ gives the highest value in that range.

The mRMR lambda sensitivity analysis (Figure *11c*) shows accuracy between 0.8849 and 0.8916 across $\lambda = 0$-1.0. Accuracy is highest at $\lambda = 0.3$ (0.8916) and $\lambda = 0.4$ (0.8912), remains between 0.8886 and 0.8904 across $\lambda = 0.5$-0.9, and is lowest at $\lambda = 0.0$ (0.8849). The selected value of $\lambda = 0.3$ lies at the peak of this range. The $\lambda = 1.0$ value (0.8890) is not an extreme degradation point but lies within the plateau.

The meta-learner tree-count sensitivity analysis (Figure *11d*) shows accuracy between 0.8869 and 0.8917 across 10-200 trees. Values are 0.8869 at 10 trees, 0.8889 at 25 trees, 0.8900 at 50 trees, 0.8912 at 75 trees, 0.8878 at 100 trees, 0.8917 at 150 trees, and 0.8883 at 200 trees. The selected configuration of 75 trees (0.8912) lies within the upper part of this range and close to the highest observed value (150 trees, 0.8917); no single tree count is clearly optimal, and the differences among 50, 75, 150, and 200 trees are small.

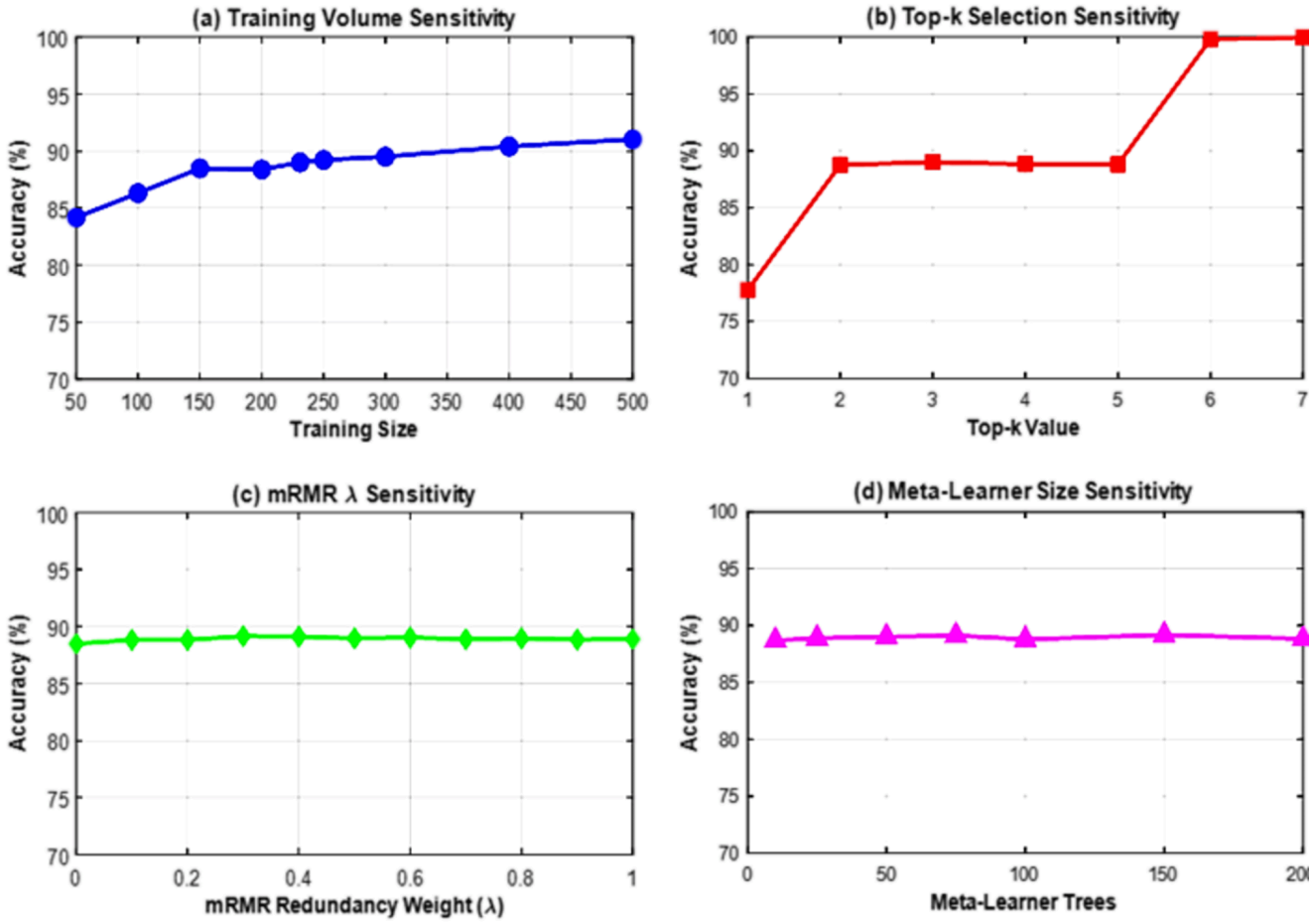


FIGURE 11: Sensitivity analysis of AMTEC model

AMTEC, Adaptive Multi-Target Ensemble Classifier; mRMR, Minimum Redundancy Maximum Relevance

AMTEC vs MTEC baseline comparison

Table *10* presents the direct comparison between AMTEC and MTEC under the common 231-sample evaluation condition. The MTEC overall accuracy of 0.8292 was adopted directly from [8] as the primary performance reference, since [8] reports results on the same Kaggle dataset [46], with the same seven-classifier pool and the same 231-sample training

constraint. This establishes comparability of setting rather than a controlled reproduction: the MTEC partition, seed, and implementation are not under the present authors' control, so the comparison is a benchmark against a published result.

AMTEC outperformed MTEC across all four targets. The largest gain was in overall accuracy, which rose by 6.10 pp, indicating improvement across all four targets simultaneously. The largest per-target gain was for Class B (0.9487-0.9983, +4.96 pp). MTEC's weakest target was Class G (0.8813), where AMTEC gained 1.15 pp (to 0.8928), while the smallest gains were for Class C (+0.25 pp, 0.9970-0.9995) and Class A (+0.03 pp, 0.9992-0.9995), where MTEC already performed strongly. The distribution of gains is therefore not a simple function of MTEC's per-target performance: the largest improvement occurs for Class B, an intermediate target, rather than for Class G, the weakest. This pattern is consistent with a target-specific effect of the architectural differences between the frameworks, in which the per-target mRMR feature subsets and re-ranked classifier composition benefit Classes B and G differently, rather than with a uniform tendency for the gains to be largest wherever MTEC performed worst.

Metric	MTEC	AMTEC	Improvement
Overall accuracy	0.8292	0.8902	+6.10 pp
G Accuracy	0.8813	0.8928	+1.15 pp
C Accuracy	0.9970	0.9995	+0.25 pp
B Accuracy	0.9487	0.9983	+4.96 pp
A Accuracy	0.9992	0.9995	+0.03 pp

TABLE 10: Direct performance comparison: AMTEC vs MTEC baseline (231 samples)

AMTEC, Adaptive Multi-Target Ensemble Classifier; MTEC, Multi-Target Ensemble Classifier

Summary of findings

The proposed AMTEC framework delivered the highest overall accuracy in both evaluation phases (231 and 100 samples), improving on both the published MTEC result and the best individual baseline at each threshold. The four architectural components are expected to contribute as follows:

The per-target mRMR feature selection is expected to reduce estimation variance before any model is trained, by retaining a compact, target-specific subset of maximally relevant and minimally redundant features rather than applying all six voltage and current features uniformly; the selected subsets should reduce the influence of noisy or redundant dimensions that would otherwise distort decision boundaries under the severe sample-to-dimension ratio imposed by 231 training samples.

The dynamic top-three selection mechanism is designed to address MTEC's fixed-composition limitation by adapting the ensemble composition per target at each data volume. Because the classifiers are re-ranked according to their validation F1-scores at each data volume, a classifier whose relative standing declines under increased data scarcity becomes proportionately less likely to be retained in the top-three composition. By preserving ensemble diversity across data regimes, the mechanism is intended to draw on complementary inductive biases from instance-based, margin-based, tree-based, and probabilistic learners, rather than relying on a single classifier whose dominance may shift as training data decrease.

Xu et al. [27] provide corroborating evidence that heterogeneous ensemble learning with weighted voting improves limited-data fault classification, while noting that unlike the weighted average soft voting mechanism (HWM) model, AMTEC eliminates manual weight tuning, adds per-target feature selection, and introduces meta-learner stacking, addressing limitations that HWM shares with MTEC. The F1-weighted soft probability voting amplified the contributions of classifiers that demonstrated stronger per-target discriminative capability while suppressing the influence of weaker learners, replacing MTEC's equal-weight mode aggregation with a performance-proportional fusion that is sensitive to per-target classifier quality differences.

The RF meta-learner stacking captured nonlinear interaction patterns among the three-classifier probability estimates and the ensemble probability Pensemble(x) that no fixed aggregation rule could represent, enabling the meta-learner to exploit agreement and disagreement signals among the base classifiers as informative features in their own right. The combined effect of these four mechanisms is reflected in AMTEC's strict multi-target overall accuracy of 89.02% at 231 training samples, a 6.10 pp improvement over MTEC's 82.92% and an 8.39 pp improvement over the best individual baseline classifier (RF: 80.63%). This result surpasses the 82-85% range typically reported for comparable multi-target and single-dataset limited-data studies reviewed in this study and, to the best of the authors' knowledge, represents the highest strict multi-target overall accuracy reported on this dataset at the 231-sample training condition. The near-perfect per-target F1-scores for Fault C (0.9994), Fault B (0.9985), and Fault A (0.9995), together with a perfect Fault A recall of 1.0000 across all Phase A-positive test samples, indicate that AMTEC improves on all four targets, with most of the residual error concentrated on the ground target (accuracy 0.8928, recall 0.8675). Because almost all strict-accuracy failures involve a ground-target error, this target accounts for the difference between the near-perfect per-target F1-scores and the strict multi-target accuracy of 89.02%.

A contribution of this study is the Phase 3 evaluation at 100 training samples. To our knowledge, this extreme-scarcity condition has not previously been reported for this dataset and formulation: within the limited-data studies reviewed in this study, no MTEC for transmission line fault classification has been systematically assessed at this threshold, and no prior study, including [8], has evaluated multi-target ensemble performance below 231 training samples on this dataset. The behaviour of these classifiers under critical data scarcity is therefore largely uncharacterised in the reviewed literature, and the Phase 3 results provide a first point of reference for this dataset and formulation. At 100 training samples (~1.3% of the full dataset), AMTEC achieved an overall accuracy of 86.87%, exceeding the best individual baseline (RF: 74.29%) by 12.58 pp. More significantly, AMTEC at 100 training samples (86.87%) exceeds the best individual classifier accuracy recorded at 231 training samples (RF: 80.63%). This suggests that AMTEC's design may mitigate the effect of data scarcity, although it should be noted that the 100-sample and 231-sample results derive from different partitions and test sets, so the comparison is indicative rather than a direct demonstration. The dynamic top-three selection further reinforces this robustness by re-ranking classifiers at each data regime independently, ensuring that performance shifts occurring as training size drops from 231 to 100 samples are accommodated rather than ignored. The result is that AMTEC's performance degraded by only 2.15 pp from Phase 2 to Phase 3, compared to 6.34 pp for RF over the same transition, indicating that AMTEC degrades at approximately one-third the rate of the nearest competitor as data become critically scarce. This degradation rate differential provides an empirical indication of AMTEC's data efficiency relative to conventional classifiers in multi-target ensemble fault classification and highlights its potential relevance for future deployment in power networks, subject to field-data validation.

The sensitivity analysis provides a more comprehensive view of the AMTEC architecture, moving the evaluation beyond a single operating point to explore the framework's hyperparameter stability. AMTEC's performance is relatively robust to hyperparameter variations: the training-volume sweep (Figure *11a*) traces accuracy across 50-500 samples on a single seed, the mRMR λ sweep reveals a broad plateau implying that the chosen redundancy weight sits within a stable region, and the meta-learner tree sweep shows that accuracy is relatively insensitive to moderate changes in tree count. The top-k sweep (Figure *11b*) exhibits a nearly flat plateau across intermediate values before rising sharply at higher values. That rise is attributed to leakage in the selection stage rather than to representational capacity, since the meta-feature vector and the meta-learner configuration are unchanged at every k; a larger candidate pool allows the ranking step to fit partition-specific regularities under the single-holdout protocol. The $k = 6$ and $k = 7$ values are therefore treated as an

artefact, and the top-three configuration is retained as a pragmatic choice that balances the diversity needed for generalisation against computational efficiency, while avoiding the regime affected by this selection leakage. These results provide tentative empirical support for the framework's design under data-scarce conditions. The mRMR λ sweep analysis (Figure *11c*) results suggest that a weak-to-moderate redundancy penalty is preferable to either extreme, although $\lambda = 1.0$ is not a clear degradation point and the differences across the intermediate range are small. The tree-count sweep (Figure *11d*) shows accuracy between 0.8869 and 0.8917 across 10-200 trees, a total spread of 0.48 pp; the selected 75-tree configuration (0.8912) lies within this range, close to the highest observed value (150 trees, 0.8917), and no single tree count is clearly optimal. These results provide tentative empirical support for the selected hyperparameter configuration under data-scarce conditions.

A notable distinction between the two frameworks is their implementation environment: MTEC was developed in Python using scikit-learn, whereas AMTEC was implemented entirely in MATLAB R2023b using the Statistics and Machine Learning Toolbox, with all classifier training, mRMR feature selection, dynamic ensemble composition, F1-weighted soft voting, and meta-learner stacking executed natively within MATLAB. Both frameworks use the same publicly available Kaggle dataset, an identical seven-classifier pool, and the same 231-sample training constraint, with the MTEC baseline accuracy of 0.8292 adopted directly from [8] as originally reported. The 6.10 pp margin is therefore a benchmark against a published result rather than a matched-codebase reproduction and may partly reflect implementation and partition differences between the two studies in addition to the architectural differences between the frameworks. Within-phase comparisons against the seven baselines, which share a common partition and seed, are unaffected by this caveat.

To contextualise AMTEC's performance within the broader fault classification literature, it is essential to recognise the fundamental architectural distinctions between AMTEC and other learning methodologies that predominate at the high accuracy end of the performance spectrum. AMTEC's 89.02% at 231 samples exceeds the comparable multi-target and single-dataset limited-data studies on transmission lines: Fahim et al. [28] reported 84.40% with 208 training samples, Shakiba et al. [29] reported 82.42% under domain shift, and MTEC [8] achieved 82.92% on the same dataset. AMTEC's 86.87% at 100 samples exceeds all three despite using fewer than half the training samples of the nearest competitor. Table *11* presents a structured comparison of AMTEC with selected studies from the limited-data stream.

Study/Ref.	Year	Application Domain	Method	Training Volume	Best Accuracy (%)
Fahim et al. [28]	2019	Transmission line	Artificial neural network (ANN)	208 samples	84.40
Shakiba et al. [29]	2022	Transmission line	Transfer learning with LeNet-5 CNN	Limited target domain	82.42
El Ghaly [8]	2025	Transmission line	Multi-target ensemble classifier	231 samples	82.92
AMTEC (proposed model)	2026	Transmission line	Adaptive multi-target ensemble classifier	231 samples/100 samples	89.02/86.87

TABLE 11: Comparison of AMTEC with state-of-the-art limited-data fault classification models

AMTEC, Adaptive Multi-Target Ensemble Classifier; MTEC, Multi-Target Ensemble Classifier; CNN, Convolutional Neural Network

Practical implications for power system protection

The operational significance of AMTEC's results can be discerned through the physical implications of each metric in the context of power system protection. However, it is crucial to recognise that the Fault G recall of 0.8675 at 231 training samples signifies a safety limitation that warrants open discussion. A Fault G recall of 0.8675 at 231 training samples indicates that approximately 13 out of every 100 actual ground-fault events in the test set were not detected by AMTEC under the limited-data regime. In practical power system protection, undetected ground faults can lead to sustained fault currents, insulation degradation, personnel safety hazards, and potential cascading failures. This shortfall underscores the recommendations of this study: specifically, the necessity to validate AMTEC on field-recorded datasets and to incorporate complementary detection mechanisms, such as differential protection relays, in any real-deployment scenario. It also indicates that AMTEC should be positioned as a decision-support layer alongside, rather than a replacement for conventional protection functions until field-data validation is complete. AMTEC's ground-fault recall at this data volume is below that of AdaBoost (0.8818) and gradient boosting (0.8693), although AMTEC compensates with higher overall accuracy and superior specificity (0.9120 vs 0.8184 and 0.8308, respectively). The Phase 3 Class G recall of 0.8729 at 100 training samples confirms that this limitation persists under more extreme data scarcity, and addressing it through field-data validation and potential augmentation remains the most critical avenue for future research.

The Phase 2 Class A recall of 1.0000 indicates that, on this partition, no Phase A-positive sample was missed. This may be relevant to transmission line relay coordination, wherein undetected Phase A faults can escalate into broader system instability, although relay behaviour itself was not evaluated in this study. The high specificity values achieved by AMTEC across all targets (G: 0.9120, C: 0.9998, B: 1.0000, A: 0.9988 in Phase 2) indicate that these recall values are obtained without a large false-alarm rate on this partition, which in a protection setting would be relevant to avoiding unnecessary circuit breaker operations.

Limitations and recommendations for future work

Three limitations of the present study merit acknowledgement. First, the study utilised a publicly available simulation-based transmission line fault dataset representing a 30-km transmission line under ideal operating conditions. Consequently, practical factors such as high-impedance faults, varying fault resistances, measurement noise, and diverse transmission network configurations were not explicitly considered. Future work should evaluate AMTEC using field-recorded fault data and more realistic simulation environments to further assess its robustness and practical applicability.

Second, the comparison against the MTEC baseline is based on a previously published result rather than a re-implemented, matched-codebase reproduction. Because MTEC was implemented in Python/scikit-learn and AMTEC in MATLAB, the reported 6.10 pp improvement should be interpreted as a benchmarked comparison against a previously reported result. Future work should formally reproduce the MTEC baseline within the same MATLAB pipeline used for AMTEC to strengthen the comparative claims.

Lastly, although the sensitivity analysis in Figure *11a* reports a single-seed sweep across 50-500 training samples, the two principal evaluation conditions (231 and 100 training samples) remain discrete operating points, and the sweep was not repeated over multiple seeds or extended to the full-data regime. Future work should conduct a multi-seed learning-curve analysis with confidence intervals across this range, extended to larger training sizes to identify the minimum data requirements for stable performance and to characterise how AMTEC's relative advantages evolve with additional data. Repeating each phase over multiple seeds with variance estimates or significance tests against MTEC and the best baseline, and re-running Phase 3 as a nested subset of Phase 2 on a commonly held-out partition, would substantiate the reported margins and isolate the effect of training size.

Conclusions

This paper proposed and evaluated AMTEC, a hybrid multi-target ensemble framework for transmission line fault classification under limited and extreme data scarcity conditions. By incorporating per-target mRMR feature selection, dynamic top-three classifier selection, F1-score-weighted soft voting, and RF meta-learner stacking, AMTEC addresses the

feature-selection, ensemble-diversity, and voting limitations identified in the MTEC baseline.

At 231 training samples, AMTEC achieved a strict multi-target overall accuracy of 89.02%, improving upon the MTEC baseline by 6.10 pp, with near-perfect per-target performance and a perfect Fault A recall across all Phase-A-positive test samples on the evaluated partition, a result that may be relevant to relay coordination. At 100 training samples, a condition not previously assessed in the reviewed literature to the authors' knowledge, AMTEC attained 86.87% overall accuracy and degraded by only 2.15 pp relative to the 231-sample condition, compared with 6.34 pp for RF over the same reduction. The sensitivity analysis indicated that the framework's performance is stable across a broad range of hyperparameter settings: for the number of selected classifiers, accuracy is nearly flat between $k = 2$ and $k = 5$, at approximately 88.8-89.0%, and it is similarly stable across the mRMR redundancy weight and the meta-learner tree count. Single-classifier selection ($k = 1$) is markedly lower, at 77.8%, and the high k values ($k = 6$-7) are treated as an artefact of selection leakage rather than as a genuine performance level. The mutual information analysis further indicated that the relative importance of the six electrical measurements varies markedly across the four fault targets, supporting the case for target-specific rather than uniform feature treatment.

Collectively, these findings indicate that AMTEC's combination of target-specific feature selection, adaptive classifier selection, performance-weighted soft voting, and stacked meta-learning constitutes a methodological contribution to multi-target ensemble learning for fault classification. AMTEC achieved the highest overall accuracy in both data-scarce phases, and its component-level robustness across progressively constrained training scenarios, together with its stability across hyperparameter settings, highlights its potential relevance for resource-constrained power network environments, subject to further validation on field-recorded datasets. The advantage is not uniform across all metrics, however: on ground-fault recall, AdaBoost (0.8818) and gradient boosting (0.8693) exceed AMTEC (0.8675), so any claim of superiority should be understood as applying to overall accuracy and specificity rather than to every per-target metric.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Daniel Kwegyir, Michael W. Apprey, Emmanuel A. Frimpong, Frank Kulor, Kafui T. Agbevanu, Evans W. Akpari, Peter Asigri

Acquisition, analysis, or interpretation of data: Daniel Kwegyir, Michael W. Apprey

Drafting of the manuscript: Daniel Kwegyir, Michael W. Apprey, Emmanuel A. Frimpong, Frank Kulor, Kafui T. Agbevanu, Evans W. Akpari, Peter Asigri

Critical review of the manuscript for important intellectual content: Daniel Kwegyir, Michael W. Apprey, Emmanuel A. Frimpong, Frank Kulor, Kafui T. Agbevanu, Evans W. Akpari

Supervision: Daniel Kwegyir, Emmanuel A. Frimpong, Frank Kulor, Kafui T. Agbevanu, Evans W. Akpari, Peter Asigri

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue.

Animal subjects: All authors have confirmed that this study did not involve animal subjects or tissue.

Conflicts of interest: In compliance with the ICMJE uniform disclosure form, all authors declare the following:

- **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work.
- **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted

work.

- **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

The dataset is publicly available on Kaggle at https://kaggle.com/code/yaarvnpatr/detection-of-the-electrical-faults

Acknowledgements

The authors acknowledge the Department of Electrical and Electronic Engineering, KNUST, for institutional support and thank Abdallah El Ghaly for reporting the MTEC benchmark results and for the Kaggle dataset reference that together enabled the comparative evaluation presented in this study.

References

1. Rahman T, Hasan T, Ahammad A, Ahmed I, Rakhaine N: MLPNN and ensemble learning algorithm for transmission line fault classification. International Transactions on Electrical Energy Systems. 2025, 2025:6114718. 10.1155/etep/6114718
2. Muzzammel R, Arshad R, Raza A, Sobahi N, Alqasemi U: Two terminal instantaneous power-based fault classification and location techniques for transmission lines. Sustainability. 2023, 15:809. 10.3390/su15010809
3. Brito PL: Hybrid approach for detection and diagnosis of short-circuit faults in power transmission lines. Energies. 2024, 17:2169. 10.3390/en17092169
4. Akter SB, Pias TS, Deeba SR, Hossain J, Rahman HA: Ensemble learning based transmission line fault classification using phasor measurement unit (PMU) data with explainable AI (XAI). PLoS ONE. 2024, 19:e0295144. 10.1371/journal.pone.0295144
5. Anwar T, Mu C, Yousaf MZ, Khan W, Khalid S, Hourani AO, Zaitsev I: Robust fault detection and classification in power transmission lines via ensemble machine learning models. Scientific Reports. 2025, 15:2549. 10.1038/s41598-025-86554-2
6. Turanlı O, Benteşen Yakut Y: Classification of faults in power system transmission lines using deep learning methods with real, synthetic, and public datasets. Applied Sciences. 2024, 14:9590. 10.3390/app14209590
7. Tunio NA, Hashmani AA, Khokhar S, Tunio MA, Faheem M: Fault detection and classification in overhead transmission lines through comprehensive feature extraction using temporal convolution neural network. Engineering Reports. 2024, 6:e12950. 10.1002/eng2.12950
8. El Ghaly A: Hybrid ML algorithm for fault classification in transmission lines using multi-target ensemble classifier with limited data. Eng. 2025, 6:4. 10.3390/eng6010004
9. Moradi-Sepahvand M, Amraee T, Aminifar F, Akbari A: Coordinated expansion planning of transmission and distribution systems integrated with smart grid technologies. International Journal of Electrical Power & Energy Systems. 2023, 147:108859. 10.1016/j.ijepes.2022.108859
10. Muntasir F, Al Wazed Wasi W, Nafis M, Hossain MM: A fast response fault detection technique for HVAC transmission line using machine learning. 2023 10th IEEE International Conference on Power Systems (ICPS). IEEE, Cox's Bazar, Bangladesh; 2023. 1-6. 10.1109/ICPS60393.2023.10428730
11. Teimourzadeh H, Moradzadeh A, Shoaran M, Mohammadi-Ivatloo B, Razzaghi R: High impedance single-phase faults diagnosis in transmission lines via deep reinforcement learning of transfer functions. IEEE Access. 2021, 9:15796-15809. 10.1109/ACCESS.2021.3051411
12. Goni MOF, Nahiduzzaman M, Anower MS, et al.: Fast and accurate fault detection and classification in transmission lines using extreme learning machine. e-Prime - Advances in Electrical Engineering, Electronics and Energy. 2023, 3:100107. 10.1016/j.prime.2023.100107
13. Al Kharusi K, El Haffar A, Mesbah M: Fault detection and classification in transmission lines connected to inverter-based generators using machine learning. Energies. 2022, 15:5475. 10.3390/en15155475
14. Etukuri S, Siva M, Varma BRK: Enhanced fault classification in inverter-fed transmission lines using deep learning. Engineering Research Express. 2024, 6:045302. 10.1088/2631-8695/ad7d6a

15. Ebrahimi H, Golshannavaz S, Yazdaninejadi A, Pouresmaeil E: Improving protection reliability of series-compensated transmission lines by a fault detection method through an ML-based model. IET Generation, Transmission & Distribution. 2024, 18:3452-3461. 10.1049/gtd2.13294

16. Michau G, Hsu CC, Fink O: Interpretable detection of partial discharge in power lines with deep learning. Sensors. 2021, 21:2154. 10.3390/s21062154

17. Sarmiento JLP, De Villa Delfino JC, Arboleda ER: Machine learning advances in transmission line fault detection: A literature review. International Journal of Science and Research Archive. 2024, 12:2880-2887. 10.30574/ijsra.2024.12.1.1150

18. Maduako I, Igwe CF, Abah JE, Onwuasaanya OE, Chukwu GA, Ezeji F, Okeke FI: Deep learning for component fault detection in electricity transmission lines. Journal of Big Data. 2022, 9:81. 10.1186/s40537-022-00630-2

19. Harish A, Prince A, Jayan MV: Fault detection and classification for wide area backup protection of power transmission lines using weighted extreme learning machine. IEEE Access. 2022, 10:82407-82417. 10.1109/ACCESS.2022.3196769

20. Saidina Omar AM, Osman MK, Ibrahim MN, Hussain Z, Abidin AF: Fault classification on transmission line using LSTM network. Indonesian Journal of Electrical Engineering and Computer Science. 2020, 20:231-238. 10.11591/ijeecs.v20.i1.pp231-238

21. Hariharan VK, Geetha A, Granelli F, Nair MG: Machine learning techniques for fault detection in smart distribution grids. Energies. 2025, 18:5179. 10.3390/en18195179

22. Jyothula VR, Purohit Y, Paraye M: Fault detection in power transmission line using machine learning techniques. 2023 14th International Conference on Computing, Communication and Networking Technologies (ICCCNT). IEEE, Delhi, India; 2023. 1-5. 10.1109/ICCCNT56998.2023.10306740

23. Alhanaf AS, Farsadi M, Balik HH: Fault detection and classification in ring power system with DG penetration using hybrid CNN-LSTM. IEEE Access. 2024, 12:59953-59975. 10.1109/ACCESS.2024.3394166

24. Yang NC, Yang JM: Fault classification in distribution systems using deep learning with data preprocessing methods based on fast dynamic time warping and short-time Fourier transform. IEEE Access. 2023, 11:63612-63622. 10.1109/ACCESS.2023.3288852

25. Nayak P, Das SR, Mallick RK, Mishra S, Althobaiti A, Mohammad A, Aymen F: 2D-convolutional neural network based fault detection and classification of transmission lines using scalogram images. Heliyon. 2024, 10:e38947. 10.1016/j.heliyon.2024.e38947

26. Da Costa CH, Moritz GL, Lazzaretti AE, et al.: A comparison of machine learning-based methods for fault classification in photovoltaic systems. 2019 IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America). IEEE, Gramado, Brazil; 2019. 1-6. 10.1109/ISGT-LA.2019.8895279

27. Xu P, Jia Z, Wang X, Ma B, Liu Z: A heterogeneous ensemble learning diagnosis scheme for limited fault sample scenarios in transmission lines. Engineering Research Express. 2025, 7:025418. 10.1088/2631-8695/add2ce

28. Fahim SR, Sarker Y, Islam OK, Sarker SK, Ishraque MF, Das SK: An intelligent approach of fault classification and localization of a power transmission line. 2019 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON). IEEE, Dhaka, Bangladesh; 2019. 53-6. 10.1109/PEEIACON48840.2019.9071925

29. Shakiba FM, Shojaee M, Azizi SM, Zhou M: Transfer learning for fault diagnosis of transmission lines [PREPRINT]. arXiv. 2022, 10.48550/arXiv.2201.08018

30. Ijaz N, Hasan MN, Koo I: Few-shot transfer learning-based fault classification in wireless sensor networks. IEEE Access. 2025, 13:55017-55033. 10.1109/ACCESS.2025.3554748

31. Thanapol P, Lavangnananda K, Bouvry P, Pinel F, Leprévost F: Reducing overfitting and improving generalization in training convolutional neural network (CNN) under limited sample sizes in image recognition. 5th International Conference on Information Technology (InCIT). IEEE, Chonburi, Thailand; 2020. 300-305. 10.1109/InCIT50588.2020.9310787

32. Peng H, Long F, Ding C: Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2005, 27:1226-1238. 10.1109/TPAMI.2005.159

33. Hassan JU, Nizami IF: Machine learning algorithm analysis for detecting and classification faults in power transmission system. 2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2). Rawalpindi, Pakistan; 2022. 1-5. 10.1109/ICoDT255437.2022.9787450

34. Ehsani R, Drabløs F: Robust distance measures for kNN classification of cancer data. Cancer Informatics. 2020, 19:1176935120965542. 10.1177/1176935120965542

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3. DOI https://doi.org/10.7759/s44389-026-00386-3

35. Goswami T, Roy UB: Predictive model for classification of power system faults using machine learning. TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON). IEEE, Kochi, India; 2019. 1881-1885. 10.1109/TENCON.2019.8929264

36. Hou K, Shao G, Wang H, Zheng L, Zhang Q, Wu S, Hu W: Research on practical power system stability analysis algorithm based on modified SVM. Protection and Control of Modern Power Systems. 2018, 3:11. 10.1186/s41601-018-0086-0

37. Cortes C, Vapnik V: Support-vector networks. Machine Learning. 1995, 20:273-297. 10.1007/BF00994018

38. Daho MEH, Settouti N, Lazouni MEA, Chikh MEA: Weighted vote for trees aggregation in random forest. 2014 International Conference on Multimedia Computing and Systems (ICMCS). IEEE, Marrakech, Morocco; 2014. 438-443. 10.1109/ICMCS.2014.6911187

39. Zhongmiao K, Zhan S: Research on fault detection and location method of medium and low voltage distribution network based on optical fiber sensing. 2024 IEEE 7th International Conference on Automation, Electronics and Electrical Engineering (AUTEEE). IEEE, Shenyang, China; 2024. 529-536. 10.1109/AUTEEE62881.2024.10869806

40. Aslan E, Özüpak Y: Fault detection in electricity transmission lines: A comparative analysis of ensemble machine learning and convolutional neural network methods. Firat University Journal of Experimental and Computational Engineering. 2026, 5:150-168. 10.62520/fujece.1581543

41. Ding Y, Zhu H, Chen R, Li R: An efficient AdaBoost algorithm with the multiple thresholds classification. Applied Sciences. 2022, 12:5872. 10.3390/app12125872

42. Friedman JH: Greedy function approximation: A gradient boosting machine. The Annals of Statistics. 2001, 29:1189-1232.

43. Pradhan R, Nayak M, Sunder GVS, Hembram B, Biswal M: Fault detection in AC microgrid using Gaussian naive Bayes: A probabilistic approach. 2024 IEEE 4th International Conference on Sustainable Energy and Future Electric Transportation (SEFET). IEEE, Hyderabad, India; 2024. 1-6. 10.1109/SEFET61574.2024.10717930

44. Sabzevari M, Martínez-Muñoz G, Suárez A: Building heterogeneous ensembles by pooling homogeneous ensembles. International Journal of Machine Learning and Cybernetics. 2022, 13:551-558. 10.1007/s13042-021-01442-1

45. Ko AHR, Sabourin R, Britto AS: From dynamic classifier selection to dynamic ensemble selection. Pattern Recognition. 2008, 41:1718-1731. 10.1016/j.patcog.2007.10.015

46. Kaggle: Detection of the electrical faults. (2025). Accessed: May 6, 2025: https://kaggle.com/code/yaarvnpatr/detection-of-the-electrical-faults.

47. Sun KW, Lee CH, Wang J: Multilabel classification via co-evolutionary multilabel hypernetwork. IEEE Transactions on Knowledge and Data Engineering. 2016, 28:2438-2451. 10.1109/TKDE.2016.2566621

48. Sujon KM, Hassan R, Choi K, Samad MA: Accuracy, precision, recall, f1-score, or MCC? Empirical evidence from advanced statistics, ML, and XAI for evaluating business predictive models. Journal of Big Data. 2025, 12:268. 10.1186/s40537-025-01313-4

How to cite this article:
Apprey M W, Kwegyir D, Frimpong E A, et al. (November 23, 2026) Multi-Target Fault Classification in Transmission Lines Using an Adaptive Ensemble Learning Approach with Limited Datasets. Cureus J Comput Sci 3 : es44389-026-00386-3. DOI https://doi.org/10.7759/s44389-026-00386-3