An Automatic Multi-scale Feature Learner and a Transformer Algorithm for Fault Detection

Chunfeng Zhang, Yu Gong, Yongjun Zhang, *Senior Member*, *IEEE*, and Siliang Liu

***Abstract*—Partial discharge (PD) in covered conductors (CCs) indicates the risks of latent faults and significant insulation degradation. Precisely identifying PD patterns is vital for maintaining electrical systems. A framework for recognizing PD patterns in overhead CCs based on an automatic multi-scale feature learning network and a Transformer is introduced in this paper. First, the method effectively removes background noise via the periodic settings of a multi-seasonal time series decomposition algorithm. An automatic feature multi-scale learning network is then constructed to learn signal features, aiming to minimize the degree of manual intervention. It enhances time series data on the basis of three-phase signal features to address class imbalance problems. An innovative global multichannel pattern recognition framework utilizing a Transformer is designed, featuring positional encoders to identify intra-phase and inter-phase feature correlations and a dynamic gating mechanism for capturing complex data patterns. In experimental validations, the proposed algorithm achieves a detection accuracy of 98.6% and a specificity of 99.2%, representing an superior performance in this field. This research provides an accurate and highly generalizable solution for PD detection, offering solid theoretical support for the digital operations and maintenance of power transmission and distribution equipment.**

***Index Terms*—Partial discharge, pattern recognition, time series decomposition, Transformer.**

I. Introduction

In power systems, the utilization of covered conductors (CCs), particularly in medium-voltage (MV) overhead lines in forested or topographically varied areas, has emerged as a significant research area [1], [2]. These conductors are preferred because of their high operational reliability and decreased land use rates. In contrast with uninsulated overhead lines, CCs do not instantaneously cause a short circuit upon contact with each other or with tree branches. However, prolonged contact may degrade their insulation performance, thereby impacting the normal operation of the distribution system [3]. Partial discharge (PD), which has been identified as a key indicator of electrical insulation deterioration, is a phenomenon that occurs in specific regions between conductors and is characterized by dielectric breakdown. Its pattern is characterized as pulse components of the current or voltage signal resulting from PD activity due to insulation deterioration [4]. Accurately assessing PD patterns is essential for early detection of emerging faults, and can contribute to enhancing the safety and reliability of the overall system [5].

With the advancement of artificial intelligence technology, particularly the growing applications of machine learning and deep learning in fault detection, there has been a notable rise in automated solutions for processing and analyzing large-scale, complex datasets [6]. The standardized process begins with data preprocessing, followed by feature extraction and classification [7], [8]. Data preprocessing commonly involves the management of background noise in the input detection signal. The task of detecting PD in large-scale data poses additional challenges, including noise corruption and interference from irrelevant information, particularly in real-world settings. Consequently, denoising and data refinement become crucial for such PD detection tasks [9], [10]. Various denoising methods, such as discrete wavelet transform (DWT), noise level estimation, and time series decomposition algorithms, are employed in different PD detection tasks [11]–[14]. However, given the complex and diverse background noise contained in real environments, the existing denoising algorithms still have significant room for improvement in accuracy. Notably, effective noise removal is critical for the final identification of PD patterns [15]. Once denoising is completed, the next step addresses the feature extraction and classification [16]. Machine learning algorithms depend on feature engineering processes informed by expert knowledge but face performance bottlenecks when confronted with more complex signal features. Owing to their powerful nonlinear learning capabilities, deep learning methods

Received: September 17, 2024
Accepted: June 20, 2025
Published Online: September 1, 2025
Chunfeng Zhang, Yu Gong, Yongjun Zhang (corresponding author), and Siliang Liu are with the School of Electric Power, South China University of Technology, Guangzhou 510640, China (e-mail: 492082682@qq.com; 202011002681@mail.scut.edu.cn; zhangjun@scut.edu.cn; 463695442@qq.com).
DOI: 10.23919/PCMP.2024.000213

exhibit greater adaptability to complex signal features, but they require customization based on the length of the input data [17]. In the context of microsecond-level data, which typically encompass fewer than a few thousand data points, such lengths are well-suited for deep learning algorithms to demonstrate their powerful automatic feature learning capabilities, particularly through the use of long short-term memory (LSTM) neural networks and convolutional neural networks (CNNs) for feature extraction. The multilevel and automatic feature extraction algorithm based on LSTM and CNNs inspires feature engineering [18]. A similar architecture is also used in this paper to study the feature extraction process. However, these methods often heavily rely on automatic feature extraction, which can lead to the loss of important information, especially in complex and changing environments [19], [20]. Hyperparameter optimization algorithms can sometimes help mitigate such issues. When managing millisecond-level and higher-sampling-rate measurement data, the number of measurements within a signal may approach a million. Due to computational limitations, the direct analysis of such large-scale data is impractical, necessitating a feature engineering process informed by expert knowledge. Specifically, techniques that combine waveform, statistical, entropy, and fractal features have yielded significant results in terms of training deep learning models [21], [22]. However, these approaches entail high computational costs, and their effectiveness is largely contingent upon the available expert knowledge. Although optimized feature extraction schemes are available, they often lack the robustness and adaptability necessary for effective transfer across different applications [23].

Presently, in research on recognizing PD patterns for CCs in MV overhead lines, the majority of studies focus only on single-phase signals and neglect the correlations among three-phase signals, potentially leading to overfitting problems [13]. Additionally, given that PD is a low-probability event, a significant imbalance exists between the numbers of fault and non-fault samples. The direct application of deep learning may result in model bias toward non-fault signals, thereby impacting the resulting classification accuracy [24].

In summary, within real-world settings, PD pattern recognition algorithms encounter challenges that arise from the complexity and diversity of background noise. Currently, the majority of the existing algorithms depend on manual expertise for performing feature extraction, which is a process that is laborious and has limited generalizability. Furthermore, because of the excessive lengths of PD signals, they surpass the direct processing capabilities of deep learning. Existing studies on PD pattern recognition primarily focus on single-phase signals, often overlooking the potential voltage amplitude correlations present among three-phase signals. This approach tends to oversimplify the problem and concentrate solely on the signal characteristics of a single phase, which may neglect the key fault information disclosed by the global features of three-phase signals. Moreover, the class imbalance issue further exacerbates the complexity of pattern recognition. These challenges require comprehensive resolution and optimization strategies in future research endeavors.

This paper aims to overcome the limitations of the existing research by proposing an innovative and intelligent method for PD detection. The main contributions of this work include efficiently removing background noise by combining noise frequency characteristics with multi-seasonal time series decomposition algorithms. Employing a multiscale 1D-CNN facilitates automated signal feature learning and enables the integration of three-phase signals to address class imbalance problems involving time series data. Additionally, constructing a dynamic gated dual-tower Transformer enhanced with positional encoding (PE) technology allows for the precise processing of PD signals. These innovations markedly improve the resulting detection accuracy and diminish the need for manual interventions.

The organization of this paper is as follows. Section II provides the detailed steps of the proposed method, encompassing signal preprocessing, automatic multiscale feature learning, and the design of a dynamic gated dual-tower Transformer classifier. Section III introduces and discusses the detailed numerical results. Finally, Section IV presents the conclusions drawn herein.

II. Methodology

In this section, an overview of the utilized dataset and a schematic diagram of the proposed method are provided. The signal processing, automatic feature learning, class imbalance correction techniques and the utilized evaluation metrics are subsequently delineated. Finally, the implementation of the Transformer classifier developed for fault detection is detailed.

A. Problem Description and Overview of the Approach

The data utilized in this study are sourced from the ENET Center at the Technical University of Ostrava [10]. The dataset comprises 8712 signal samples collected from 2904 distinct measurement points, with each point recording voltage signals in three phases. These measurement points are situated in remote terrains, including forests and mountains. Each signal sample represents a 50 Hz single-cycle voltage waveform recorded at a sampling frequency of 40 MHz and contains 800 000 data points. Experienced professionals at the ENET Center have meticulously analyzed all the signals and labeled them on the basis of their expertise

in PD patterns, whereas the Center employed multiple algorithms to review all the samples for evaluation purposes. The samples have been pre-classified as PD (525 samples) or non-PD samples (8186 samples). Examples of these signals are illustrated in Fig. 1.

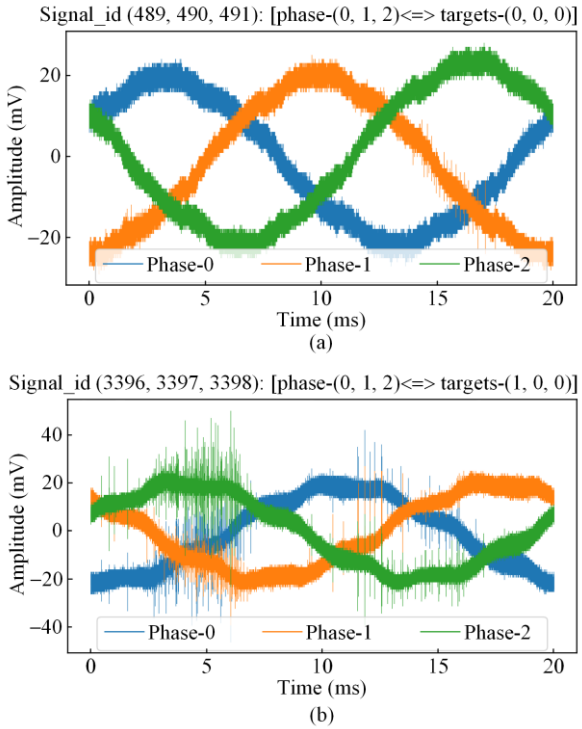


Fig. 1. Sample signal recording from the ENET dataset (target is 0 for non-PD; target is 1 for PD).

The objective of the fault detection task is to ascertain the presence of a PD signal in the input data, and a model that is adept at accurately classifying PD and non-PD signals can aid in preventing the complete combustion of CCs, potentially averting serious outcomes such as wildfires. One significant challenge concerning this task is that certain non-PD signals may present fault-like patterns. Importantly, in contrast with laboratory data, the utilized dataset lacks prior knowledge on fault-related PD patterns or background noise interference [12].

The architecture of the proposed method is illustrated in Fig. 2, detailing the following specific implementation steps.

1) The initial three-phase signal is decomposed by decomposing multiple seasonal trends via the LOESS (MSTL) algorithm to eliminate the trend and seasonal components. The residual components obtained from this decomposition process are subsequently employed to remove random impulse interference, thus completing the denoising of the background noise.

2) A multiscale 1D-CNN is employed to automatically extract features from the noise-reduced signals. Subsequently, utilizing an integrated form of the three-phase signal features, a multivariate time series data enhancement algorithm is applied for performing PD data enhancement to mitigate the effects of imbalanced classes.

3) A dynamically gated dual-tower Transformer, featuring a step encoder and a channel encoder, is constructed. This setup facilitates the learning of in-phase multiscale relationships and three-phase interdependence, culminating in training classifiers to distinguish between PD and non-PD signals.

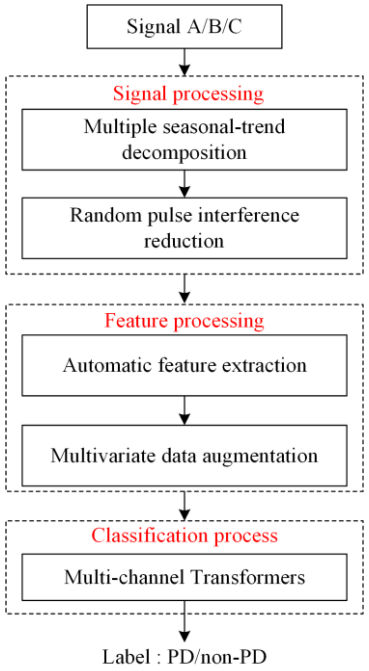


Fig. 2. Diagram of the proposed PD detection approach.

B. Denoising Based on the Decomposition of Multiple Seasonal Trends

References [10] and [12] highlight that raw signals derived from real MV overhead lines exhibit significant uncertainty. This is primarily attributed to their unique background noise, which is nearly impossible to replicate under laboratory conditions. The ENET Center has classified all the signal pulse components contained in the raw signals, except for the PD patterns, as background noise. Background noise originates from various sources, including discrete spectrum interference (DSI) from radio transmissions, repetitive pulse interference from power electronics, random pulse interference (RPI) from events such as lightning, switching operations, corona discharges, and ambient noise, including amplifier noise [10].

Overhead lines function as long-line antennas, where DSI predominantly originates from longwave transmitters, and high levels of DSI can occasionally obscure PD patterns in the original signals. Another significant source of interference is RPI, which typically manifests as corona discharges. RPI frequently results in misclassified peaks within the temporal domain of the

original signal, which may be erroneously identified as PD modes (refer to Fig. 3).

Traditional denoising techniques, such as threshold setting and signal flattening, are now deemed insufficient, whereas methods involving digital trap filters and wavelet transforms typically depend on human intervention and expert insights [7]. Consequently, a multivariate time series decomposition algorithm is introduce here as a noise reduction approach. This method aims to streamline the noise reduction process and offers a novel perspective with respect to signal noise reduction.

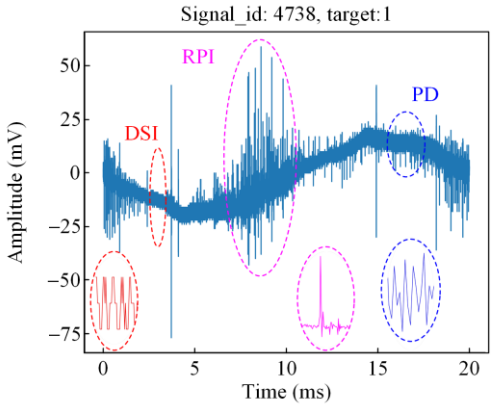


Fig. 3. PD pattern with noise during a CC fault.

The purpose of time series decomposition is to reveal fundamental patterns in time series data, enabling each pattern to exhibit specific characteristics or types of behavior within the data. As depicted in Fig. 1, the original signal exhibits noise superimposed upon a sinusoidal cycle. Given that PD constitutes a type of irregular noise stemming from unpredictable external events, such as branches hitting conductors, time series decomposition algorithms are initially employed in this paper to extract noise information from the input signals and subsequently conduct PD pattern recognition on the basis of this extracted noise information.

A seasonal-trend decomposition via the LOESS (STL) algorithm is implemented for time series decomposition in [14]. STL is widely employed in numerous applications. However, its limitations include reduced capacity to handle trend changes, long periods, high-noise data, and the ability to decompose only a single type of seasonal feature. The nature of single-season decomposition, which focuses solely on the seasonal changes observed over a fixed period, potentially fails to adequately capture seasonal fluctuations across multiple time scales in the given signal. Therefore, this approach may result in significant unexplained fluctuations remaining in the residual components.

In comparison, multi-seasonal decomposition accounts for multiple scales of seasonal changes, including annual seasonality and shorter cycles such as monthly or weekly seasonality [25]. Consequently, multi-seasonal decomposition is more affective at capturing the seasonal characteristics of time series data, resulting in smaller residual components due to the explicit modeling of seasonal fluctuations. This section details the utilization of the MSTL algorithm for decomposition, which targets a refined multi-seasonal decomposition, illustrated as:

$$\begin{cases} \boldsymbol{s}_t = \sum_{i=1}^{m} \boldsymbol{s}_{i,t} \\ \boldsymbol{y}_t = \boldsymbol{\tau}_t + \boldsymbol{s}_t + \boldsymbol{r}_t, \ t = 0,1,\cdots,N-1 \end{cases} \tag{1}$$

where $\boldsymbol{y}_t$ denotes the time series observed at time t; $\boldsymbol{\tau}_t$ signifies the trend component; m indicates the number of distinct seasonal components; $\boldsymbol{s}_{i,t}$ represents the ith seasonal component at time t; $\boldsymbol{s}_t$ constitutes the aggregate of multiple seasonal components; and $\boldsymbol{r}_t = \boldsymbol{a}_t + \boldsymbol{n}_t$ encapsulates the residual part, encompassing noise $\boldsymbol{n}_t$ and potential anomalies $\boldsymbol{a}_t$.

$\boldsymbol{\tau}_t$ encapsulates the global characteristics of the original signal. A trend manifests in the data as a continuous increase or decrease, and $\boldsymbol{\tau}_t$ can depict the gradually varying 50 Hz sinusoidal cycle. $\boldsymbol{s}_t$ illustrates the seasonality within the original signal, signifying seasonal patterns as repetitive cycles embedded within the signal. In the context of the signal under discussion, $\boldsymbol{s}_t$ can denote elements such as repetitive pulse interference and radio interference, encompassing periodic repetitive contents. $\boldsymbol{r}_t$ mirrors disturbances or noise within the original signal and represents the residual component remaining after the exclusion of $\boldsymbol{\tau}_t$ and $\boldsymbol{s}_t$.

As previously noted, PD in overhead line covering conductors manifests randomly and irregularly, and is typically distributed within the residual component $\boldsymbol{r}_t$ of the voltage signal. Additionally, RPI, which is epitomized by corona discharges, constitutes another significant source of interference, frequently yielding misclassified peaks within the time domain of the signal, which are readily mistaken for PD patterns [10]. Consequently, while the denoising process based on temporal decomposition mitigates noise, the processed signal retains corona discharge peaks alongside the actual PD peaks. Corona discharge peaks typically manifest as high-amplitude false hit peaks, and are often accompanied by peaks with the opposite polarity, thereby forming symmetric peak pairs. Parameter setting in [10] is integrated regarding false hit peaks for effectively removing corona discharges, while to more precisely filter out background noise, the relationship between the lower threshold and the achieved recognition accuracy is investigated via the light gradient boosting machine (LightGBM), aiming to establish the optimal lower threshold.

C. Automatic Multi-scale Feature Learning

Owing to the transient nature of PD signals, high-frequency sampling is needed. The sampling process yields signal lengths near one million, surpassing the processing capabilities of deep learning algorithms such as the generative pretrained Transformer (GPT) (limited to a length of 4096). Consequently, extracting features from voltage signals becomes an essential step [14].

Inspired by visual Transformers, which employ image patches as sequential data, and cross-modal Transformers for sleep stage classification [18], a multi-scale 1D-CNN is used as depicted in Fig. 4, enabling the learning of feature representations directly from the original signals. Unlike methods based on standard architectures such as ResNet50-1D-CNN, this approach enables the model to discern optimal feature representations in parallel by integrating both local and global aspects, thereby incorporating extensive voltage amplitude and phase variations as well as their correlations into the PD detection analysis. This methodology strives to surpass the performance of manual features, simultaneously diminishing the amount of manual intervention required in the proposed method.

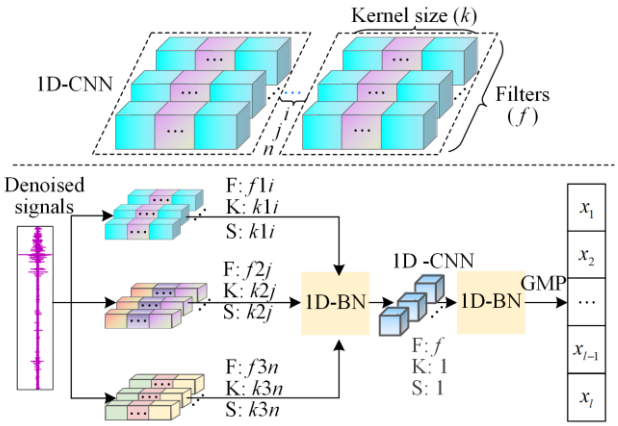


Fig. 4. Multi-scale 1D-CNN blocks with filters (F), kernel size (K), and stride (S).

The noise-reduced signal, measuring 800 000 points in length, is fed into a multiscale 1D-CNN for automatic feature extraction. The feature extraction process uses a nonoverlapping window method, with the window size set equal to the convolutional kernel size (K). To prevent overlapping between windows, the convolutional stride size (S) is configured to match K, ensuring the extraction of features with uniform sizes across all windows. Additionally, this setting enables the features acquired from each window to encapsulate the information of their respective periods, thereby preserving the continuity of the time series and enhancing the interpretability of the features.

Raw signals undergo processing via three parallel paths, each featuring a 1D convolutional layer succeeded by a leaky rectified linear unit (LeakyReLU) activation layer. The features at each scale are subsequently normalized via 1D batch normalization (1D-BN). These features are concatenated along the embedding dimension and subsequently processed by a 1D-CNN with a kernel size of 1, followed by LeakyReLU activation, batch normalization, and ultimately aggregation by a global maximum-average pooling (GMP) layer. The global maximum pooling and global average pooling steps each generate a value per channel, and these values are combined to form the final output via the fully connected layer. This network architecture minimizes the dimensionality of the features while retains key statistical information, offering a rich representation for conducting an in-depth analysis. Moreover, larger convolutional kernels capture broader temporal features (global features), whereas smaller kernels concentrate on local details (local features).

Upon completing the multiscale feature learning process, addressing the class imbalance problem in the data becomes imperative. In the current dataset, the number of PD samples is 525, representing approximately 6% of the total, whereas the number of non-PD samples is 8187, constituting approximately 93% of the total. This significant class imbalance issue, if unaddressed, could result in training bias within the modeling algorithm. Despite achieving high accuracy, the model may neglect the more crucial PD samples. While oversampling is a common method among researchers for addressing class imbalances, this strategy can induce overfitting and impede the model generalization procedure.

Neural network-based generative models, including generative adversarial networks (GANs) and variational autoencoders (VAEs), are well-suited for modeling complex relationships. Data augmentation algorithms, exemplified by GANs such as TimeGAN, are capable of capturing temporal dynamics across time series data [26]. A framework based on an auxiliary classifier GAN is developed, which uses stacked one-dimensional convolutional layers to extract local features from raw input data [27]. Reference [28] introduces a deep adversarial data augmentation technique based on the least-squares GAN (LSGAN), making the bidirectional gated recurrent unit (BiGRU)-Attention-based model applicable to small training datasets. Furthermore, a conditional Wasserstein GAN with a gradient penalty (CWGAN-GP) is employed to generate synthetic data, facilitating the creation of balanced and representative training sets for classifiers. This approach not only mitigates the overfitting tendency associated with the synthetic minority oversampling technique but also enhances the learning capacities of neural networks by ensuring a uniform weight distribution [29]. GANs have achieved significant success and continue to evolve.

Limited research has been conducted on time series data generators, represented by VAEs [30], in which a conditional VAE data augmentation algorithm is developed based on LSTM networks. Reference [31] proposes TimeVAE, which uses a CNN in both its encoder and decoder, with additional parallel blocks in the decoder while each block is responsible for specific temporal attributes. in [32], a data augmentation method is introduced using the decoding weights of an autoencoder based on a single-hidden-layer feedforward neural network. This approach involves implementing grouping and training according to labels, obtaining the decoding weights of the autoencoder to generate weighted samples, and linearly combining these weighted samples with the original samples to produce augmented samples.

The VAE generation algorithm allows for greater control over the variability of the generated data by directly influencing the standard deviation of the latent distribution of the original dataset. Given the complexity of the PD data discussed in this paper, which may necessitate some degree of manual intervention, the VAE generation algorithm is utilized to address class imbalance issues.

For PD data, which involve three phases, the interpulse correlations between different phases are intricately linked to PD occurrence. The phase experiencing a PD fault can induce changes in the signals of the other two phases. Therefore, considering the three-phase data holistically for learning purposes while accounting for the temporal relationships between signals is essential.

Consequently, the proposed study involves a time series data augmentation algorithm, which treats the three-phase signals collectively. In multivariate time series modeling cases, three aspects are pivotal: 1) the correlations among multiple variables at a given time step; 2) short-term correlations; and 3) long-term correlations.

A time series data augmentation algorithm developed on the basis of metric learning and a VAE is presented in this paper, tailored to the above three requirements. This algorithm employs metric learning to characterize the correlations among variables at specific time steps and uses GRU neural networks to address both the short-term and long-term correlation requirements of time series data.

The loss function of the VAE is also referred to as the evidence lower bound (ELBO) loss function [33], depicted as:

$$L_{\theta,\phi} = -E_{q_\phi(z|x)}\left[\log p_\theta(\boldsymbol{x}|\ z)\right] + D_{\mathrm{KL}}\left(q_\phi(z|\ \boldsymbol{x}) \,||\, p_\theta(z)\right) \tag{2}$$

where the first term denotes the negative log-likelihood of the data given $\boldsymbol{z}$, which is sampled from $q_\phi(z|\ \boldsymbol{x})$; the second term corresponds to the Kullback-Leibler (KL) divergence loss between the encoded latent spatial distribution and the prior distribution.

In a traditional VAE, the encoder maps the input data into a low-dimensional hidden variable distribution within a latent space via a purely unsupervised approach. However, this approach for representing the hidden variable distribution may not be optimal, as it can potentially overlook crucial data structures and relationships [34]. Thus, guided learning based on labels is utilized to construct a discriminative latent VAE space that is more suitable for data augmentation. A triplet loss function is introduced to encourage the encoder to yield more discriminative feature representations, as:

$$\mathcal{L}_{\mathrm{triple}}(\cdot) = \max\left\{0, \left\|\boldsymbol{z}^{(\mathrm{a})} - \boldsymbol{z}^{(\mathrm{p})}\right\| + \rho - \left\|\boldsymbol{z}^{(\mathrm{a})} - \boldsymbol{z}^{(\mathrm{n})}\right\|\right\} \tag{3}$$

where 'a' represents the anchor sample; 'p' represents the positive sample; 'n' represents the negative sample; and ρ serves as a margin.

Metric learning is often integrated by incorporating an additional metric loss term into the loss function of a VAE, establishing a discriminative latent VAE space through metric learning. The modified loss function is delineated as:

$$L_{\mathrm{VAE-triplet}} = L_{\mathrm{recon}} + \alpha L_{\mathrm{KL}} + \beta L_{\mathrm{triplet}} \tag{4}$$

where α and β serve as the weighting coefficients for the KL loss term and the triplet loss term, respectively, facilitating the balancing of the three loss components; L_{recon} represents the cross-entropy between the input sequence vector and its reconstruction produced by the VAE decoder; L_{KL} denotes the KL divergence between the approximate posterior distribution of the latent vector derived from the VAE encoder and the prior distribution. Ordinary VAE often struggle to continuously embed the temporal attributes or other domain attributes of targets into the latent VAE space without constraints, and the L_{triplet} triplet loss function imposes constraints that facilitate the continuous embedding of pertinent temporal or domain attributes in the latent VAE space.

Figure 5 presents the overall schematic of the VAE model, which is based on metric learning. For the metric loss calculation, the anchor, positive, and negative labeled samples within the time series dataset are utilized. $\boldsymbol{Y}_{\mathrm{a}}$, $\boldsymbol{Y}_{\mathrm{b}}$, and $\boldsymbol{Y}_{n}$ denote the labels for the anchor, positive, and negative samples, respectively, encompassing the sample similarity variables computed via dynamic time warping (DTW) and the original sample labels. $f(\boldsymbol{x}_{\mathrm{a}})$, $f(\boldsymbol{x}_{\mathrm{p}})$ and $f(\boldsymbol{x}_{n})$ are latent vectors corresponding to the input time series data. Optimizing the positions of the anchor, positive, and negative latent variables allows the latent VAE space to more closely approximate the actual time series data space.

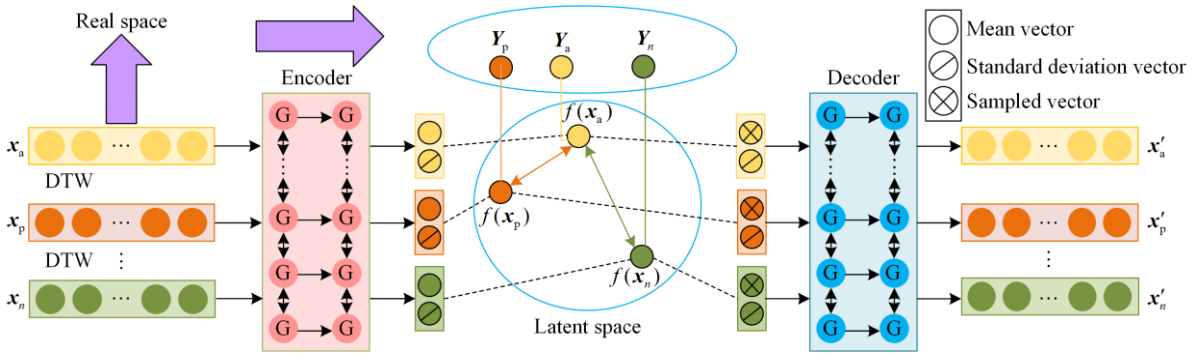


Fig. 5. Block diagram of the multivariate time series data augmentation algorithm.

D. Pattern Recognition Algorithm Based on Transformers

As depicted in Figs. 1 and 6, the occurrence of PD in one phase may induce changes in the voltage signals of the remaining two phases, illustrating the strong correlation between the PD signals across the three phases. According to a statistical analysis of the faulty samples, 156 exhibit issues in all three phases, constituting 80.4% of the total; 21 present problems in two phases, representing 10.8% of the total; and 17 present problems in just one phase, accounting for 8.8% of the total. This statistical analysis suggests that simultaneous three-phase PDs predominate, indicating that an occurrence in one phase not only impacts the voltage signals of the other two phases but also implies a high likelihood of concurrent discharges in these phases, similar to how a falling tree impacts all three-phase signals measured at a single point. These analyses underscore the importance of investigating the correlations among the three-phase signals.

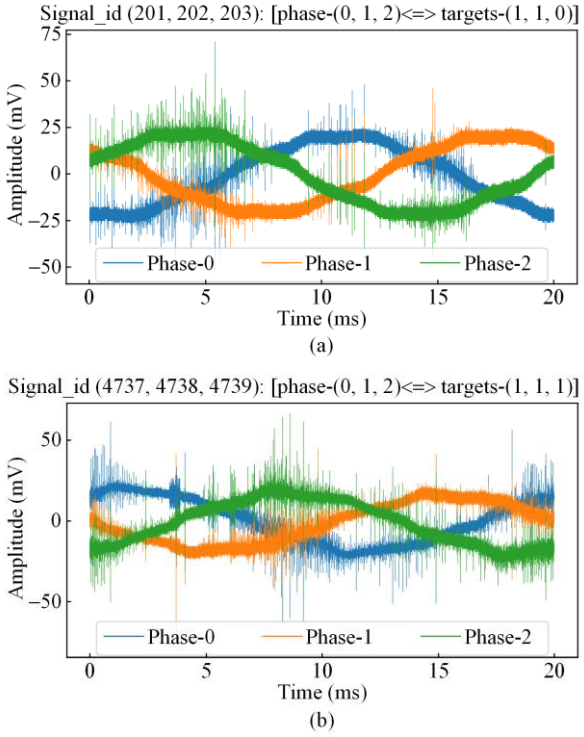


Fig. 6. Three-phase PDs. (a) Two of the three phase signals have PD signals. (b) The PD signal in all three phase signals.

To more effectively explore the inter-phase and intra-phase relationships of PD signals, the Transformer architecture is utilized as a classifier. Since its introduction in 2017, the Transformer architecture has emerged as a principal model for processing sequential data. Its core strength lies in its self-attention mechanism, which helps the model produce a wide variety of multi-stream variants, enabling it to directly learn dependencies between different parts of the input sequence and excel in time series analysis tasks [35].

A Transformer employs its self-attention mechanism to directly determine the relationships between the elements in a sequence, which is a feature that is particularly advantageous for addressing the abrupt and multi-scale characteristics of PD signals. In contrast with traditional recurrent neural networks (RNNs) and LSTM, a Transformer eliminates the need for sequential data processing, thus facilitating the parallel processing of the entire sequence and the capture of long-range dependencies. The self-attention mechanism is delineated as:

$$\text{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^{\text{T}}}{\sqrt{d_{\text{k}}}}\right)\boldsymbol{V} \tag{5}$$

where $\boldsymbol{Q}$, $\boldsymbol{K}$, and $\boldsymbol{V}$ represent the query, key, and value matrices, respectively; and d_{k} denoting the dimensionality of the keys. The multi-head attention mechanism extends the self-attention concept by concurrently executing several self-attention operations in parallel, each targeting different segments of the input data. This approach enables a more thorough information capture process. This mechanism permits the model to concentrate on various segments of the input across diverse representational subspaces. This capability is advantageous for apprehending the diverse temporal and frequency features that are inherent in PD patterns. The multi-head attention mechanism is depicted as:

$$\begin{cases} \text{MultiHead}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \text{Concat}(e_{\text{head1}},\cdots,e_{\text{head}i})\boldsymbol{W}^{\text{O}} \\ e_{\text{head}i} = \text{Attention}(\boldsymbol{Q}\boldsymbol{W}_i^{\text{Q}},\boldsymbol{K}\boldsymbol{W}_i^{\text{K}},\boldsymbol{V}\boldsymbol{W}_i^{\text{V}}) \end{cases} \tag{6}$$

where $e_{\text{head}i}$ denotes the output of the ith attention head; $\boldsymbol{W}_i^{\text{Q}}$, $\boldsymbol{W}_i^{\text{K}}$, $\boldsymbol{W}_i^{\text{V}}$ and $\boldsymbol{W}^{\text{O}}$ are the learned weight matrixes.

The architecture of the Transformer is inherently flexible, allowing it to effectively adapt to multiscale time series data. Through the tailored design of the attention mechanism and network layers, the model can be attuned to the various time scales that are inherent in PD signals [35]. This paper regards PD classification as a multivariate time series data classification problem, and the proposed method designs classifiers through two strategies: a twin-tower encoder and a dynamic gating mechanism. Figure 7 shows the overall architecture of the dynamic gated Transformer network presented in this paper.

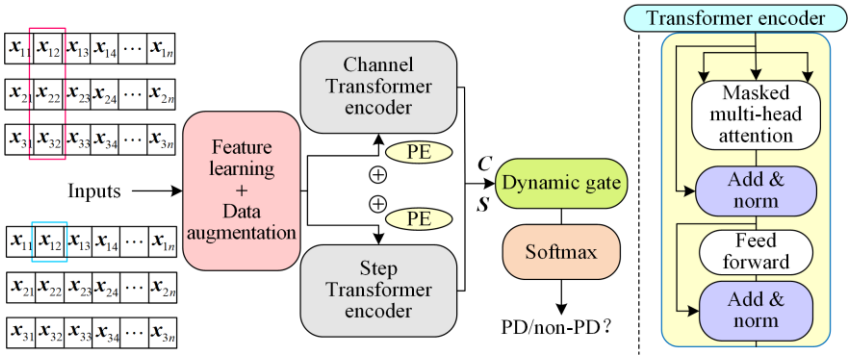


Fig. 7. Diagram of the architecture of the dynamic gated Transformer network.

The dual-tower Transformer architecture is an extension of the traditional Transformer that processes three-phase signals in parallel through two encoder towers: a step encoder and a channel encoder.

The step encoder is designed to capture correlations in the signal across different time steps. This encoder focuses on the relationships among all channels (e.g., three-phase signal data) at a specific time step. The highlighted blue area on the left side of Fig. 7 illustrates how the step encoder pays attention to data from different channels at a specific time step (e.g., the *i*th time step). This mechanism enables the model to consider all channel data simultaneously at the same time point, thereby enhancing its understanding of the overall state at that time step.

The channel encoder is intended to capture correlations between different channels (e.g., three-phase signals) across the entire time series. It should be noted that the designed channel encoder combines the PE technology with the masked multi-head attention mechanism to effectively learn the sequential dependencies between the three-phase monitoring data. The highlighted red area on the left side of Fig. 7 shows how the channel encoder attends to all time steps within a specific channel. This mechanism allows the model to comprehensively consider data from all time points within a channel, thereby improving its understanding of the overall dynamic changes within the corresponding channel.

The combination of these two encoders provides a comprehensive PD signal analysis framework that is capable of simultaneously encoding and capturing the temporal dynamics and potential phase relationships in PD signals. This bidirectional encoding method offers an effective approach for understanding the complex structures of multivariate time series.

To merge the features of the two towers encoding stepwise and channel correlations, simply concatenating all features from both towers could diminish their individual performance. This paper introduces a novel dynamic gating mechanism to effectively learn the outputs from each tower. Initially, upon obtaining the outputs $\boldsymbol{C}$ and $\boldsymbol{S}$ from the two towers, they are processed through a fully connected layer with nonlinear activation, yielding the primary gated features $\boldsymbol{C}'$ and $\boldsymbol{S}'$, as:

$$\begin{cases} \boldsymbol{C}' = f(\boldsymbol{W}_{\mathrm{c}}\boldsymbol{C} + b_{\mathrm{c}}) \\ \boldsymbol{S}' = f(\boldsymbol{W}_{\mathrm{s}}\boldsymbol{S} + b_{\mathrm{s}}) \end{cases} \tag{7}$$

where $\boldsymbol{W}_{\mathrm{c}}$ and $\boldsymbol{W}_{\mathrm{s}}$ denote the weight matrices; b_{c} and b_{s} denote the bias terms; and $f(\cdot)$ represents the nonlinear rectified linear unit (ReLU) activation function.

Subsequently, an extra network layer is introduced, which accepts $\boldsymbol{C}'$ and $\boldsymbol{S}'$ as inputs and outputs parameters for adjusting the gating weights. This layer is implemented as a fully connected layer that is designed to dynamically adjust the gating weights on the basis of the input features.

$$\boldsymbol{D} = g(\boldsymbol{W}_{\mathrm{d}} \times \mathrm{Concat}(\boldsymbol{C}', \boldsymbol{S}') + b_{\mathrm{d}}) \tag{8}$$

where $\boldsymbol{W}_{\mathrm{d}}$ represents the weight matrix; b_{d} represents the bias term; and $g(\cdot)$ represents the nonlinear sigmoid activation function. The gating weights are computed via the softmax function and subsequently combined with the dynamically adjusted parameters $\boldsymbol{D}$.

$$\begin{cases} \boldsymbol{h} = \boldsymbol{W} \times \mathrm{Concat}(\boldsymbol{C}, \boldsymbol{S}) + b \\ \boldsymbol{g}_1, \boldsymbol{g}_2 = \mathrm{Softmax}(\boldsymbol{h} \odot \boldsymbol{D}) \end{cases} \tag{9}$$

where $\boldsymbol{W}$ denotes the weight matrix and b denotes the bias term, both serving to transform the concatenated raw outputs of the two encoders; $\boldsymbol{h}$ represents the intermediate feature vector derived from this transformation; and $\odot$ symbolizes the elementwise multiplication operation.

Finally, the computed gating weights $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$ are applied to weight $\boldsymbol{C}$ and $\boldsymbol{S}$, respectively, with the resultant weighted features then concatenated to construct the final feature vector $\boldsymbol{y}$:

$$\boldsymbol{y} = \text{Concat}(\boldsymbol{C} \times \boldsymbol{g}_1,\ \boldsymbol{S} \times \boldsymbol{g}_2) \tag{10}$$

The key aspect of the dynamic gating mechanism is its ability to dynamically modify the gating weight computation according to the characteristics of the input data, as opposed to simply relying on the input features alone. This method allows the model to more flexibly adapt to a range of input scenarios, consequently increasing its ability to discern complex data patterns.

E. Evaluation Parameters

The effectiveness of PD recognition is manifested in the classification outcomes. This study uses true positive (T_{P}) and true negative (T_{N}), which are the numbers of correctly predicted positive and negative samples, respectively; and false positive (F_{P}) and false negative (F_{N}), which are the numbers of incorrectly predicted positive and negative samples, to characterize the obtained classification results [14].

Precision represents the ratio of correctly classified positive-class samples:

$$P = \frac{T_{\text{P}}}{(T_{\text{P}} + F_{\text{P}})} \tag{11}$$

Recall denotes the ratio of positive-class samples that are accurately identified as positive:

$$R = \frac{T_{\text{P}}}{(T_{\text{P}} + F_{\text{N}})} \tag{12}$$

The F1-score is defined as the weighted harmonic mean between precision and recall:

$$F_1 = \frac{2PR}{(P + R)} \tag{13}$$

Specificity measures the ratio of negative-class samples that are accurately classified as negative:

$$S = \frac{T_{\text{N}}}{(T_{\text{N}} + F_{\text{P}})} \tag{14}$$

The false positive rate (F_{PR}) reflects the ratio of negative class samples incorrectly identified as positive:

$$F_{\text{PR}} = \frac{F_{\text{P}}}{(T_{\text{N}} + F_{\text{P}})} \tag{15}$$

The false-discovery rate (F_{DR}) indicates the ratio of samples predicted as positive that are in fact negative:

$$F_{\text{DR}} = \frac{F_{\text{P}}}{(T_{\text{P}} + F_{\text{P}})} \tag{16}$$

The negative predictive value (N_{PV}) represents the ratio of negative-class samples that are precisely predicted as negative:

$$N_{\text{PV}} = \frac{T_{\text{N}}}{(T_{\text{N}} + F_{\text{N}})} \tag{17}$$

Additionally, accuracy-related metrics are introduced to offer a more comprehensive assessment of the achieved performance. The predictive accuracy represents the proportion of all measurement points that are correctly classified. The average accuracy, which is an accuracy metric for multiclass problems, is calculated as the average accuracy across each class, reflecting the average performance of the classifier across all categories. Unlike predictive accuracy P_{a}, average accuracy A_{a} provides a more comprehensive performance evaluation in multi-class classification scenarios since it accounts for the imbalance among different categories.

$$P_{\text{a}} = \frac{T_{\text{P}} + T_{\text{N}}}{T_{\text{P}} + T_{\text{N}} + F_{\text{P}} + F_{\text{N}}} \tag{18}$$

$$A_{\text{a}} = \frac{1}{N}\sum_{i=1}^{N} \frac{T_{\text{P}i} + T_{\text{N}i}}{T_{\text{P}i} + T_{\text{N}i} + F_{\text{P}i} + F_{\text{N}i}} \tag{19}$$

where $T_{\text{P}i}$ denotes the true positives for the *i*th class; $T_{\text{N}i}$ denotes the true negatives; $F_{\text{P}i}$ denotes the false positives; and $F_{\text{N}i}$ denotes the false negatives for the *i*th class.

III. Experiments and Results

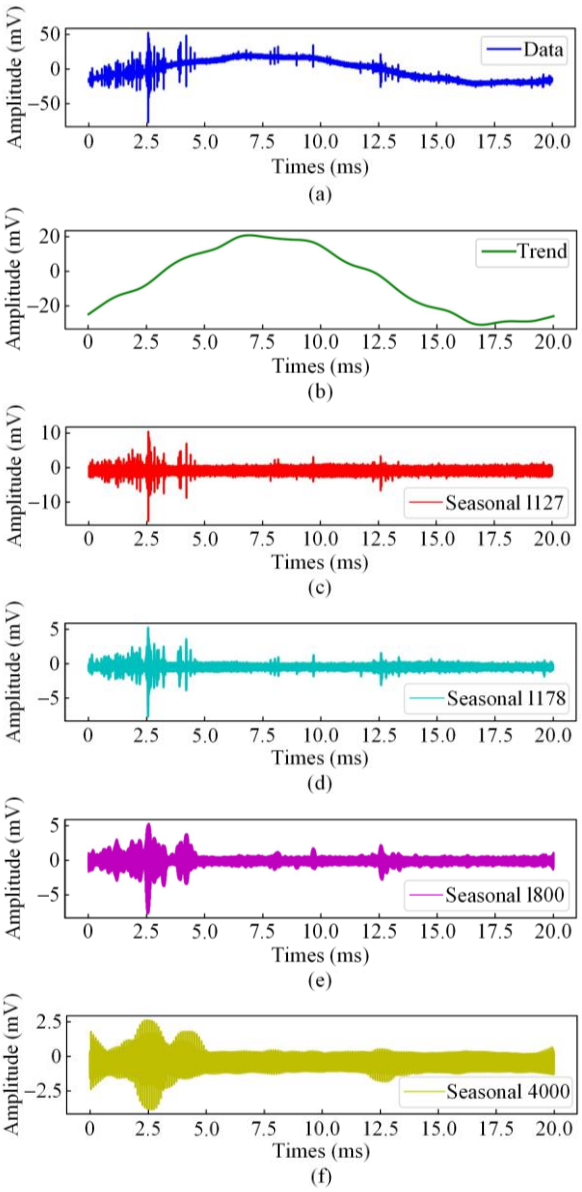
In this section, the implementation details of the proposed method are outlined, along with the results of the PD classification task.

A. Analysis of the Denoising Algorithm

An in-depth analysis is conducted on the raw signals derived from MV overhead lines, with particular emphases on the diversity and complexity of the background noise. Notably, the ENET Center has classified all the signal pulse components, except for the PD patterns, as background noise. This encompasses DSI, repetitive pulse interference, RPI, and ambient and amplifier noise [10].

Drawing on the relevant literature and the characteristics of noise signals, the frequencies of the principal noise sources are determined. The frequency range for DSI noise generally lies between 225 kHz and 1500 kHz. For repetitive pulse interference, 50 kHz is chosen as the representative frequency. The environmental and amplifier noises exhibit broad frequency ranges. At a sampling rate of 40 MHz and a duration of 20 ms, a total of 800 000 data points are obtained. With these parameters, the periods for various noise sources are calculated and converted into corresponding numbers of sampling points. Specifically, the period for 225 kHz DSI noise equates to 178 sampling points, that for 1500 kHz DSI noise corresponds to 27 sampling points, that for 50 kHz repetitive pulse interference corresponds to 800 sampling points, that for 1 kHz environmental noise corresponds to 40 000 sampling points, and that for 10 kHz environmental noise corresponds to 4000 sampling points [7], [10].

Building on the aforementioned setup, the MSTL algorithm is used for signal decomposition purposes, effectively separating various noise components, as depicted in Fig. 8. Figure 8(a) displays the input raw signal, while Fig. 8(b) illustrates a sinusoidal curve representing the trend component derived from the signal decomposition process. This component isolates the interference noise of the power line, specifically the 50 Hz sine wave, akin to the function of a trap filter. Figures 8(c)–(g) display the seasonal components isolated by the MSTL algorithm, each depicting different background noise components: 1500 kHz DSI noise, 225 kHz DSI noise, 50 kHz repetitive impulse interference, 1 kHz ambient noise, and 10 kHz ambient noise. The effectiveness of the proposed method is underscored by its capacity to accurately identify and analyze various noise sources, thereby providing a solid foundation for subsequent signal processing and noise suppression steps.



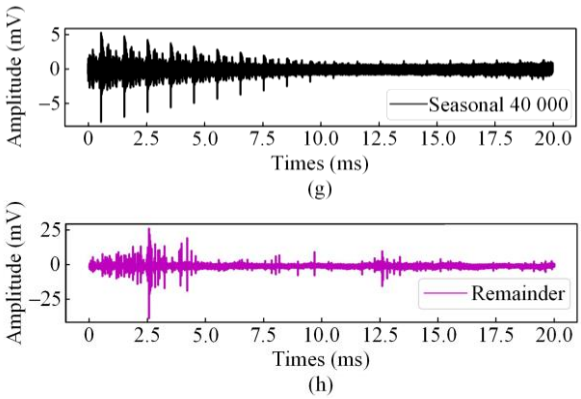


Fig. 8. Example of MSTL signal decomposition. (a) Original signal. (b) Trend component. (c) Seasonal component with period 1127. (d) Seasonal component with period 1178. (e) Seasonal component with period 1800. (f) Seasonal component with period 4000. (g) Seasonal component with period 40 000. (h) Residual component.

Ultimately, only the residual component r_t is retained as shown in Fig. 8(h). As described in Section II.B, the residual component r_t still contains RPI as well as genuine PD peaks. As one of the main sources of interference in overhead lines, RPI signals share characteristics that are similar to those of PD signals, often manifesting as corona discharges. The pulse waveform and frequency of RPI are strongly influenced by field conditions, and its signals may have a larger amplitude than PD signals [10].

Following the identification criteria for corona discharge peaks, which are outlined in [10], these disturbances are identified and eliminated. This procedure is illustrated in Fig. 9, where each peak is compared with its subsequent peaks. Peaks are classified as symmetric peak pairs if the distance between them falls below the preset maximum distance threshold and if the amplitude ratio exceeds the predetermined height ratio. Oscillations commonly follow symmetric peak pairs resulting from corona discharges, and these oscillations can be misclassified as PD modes.

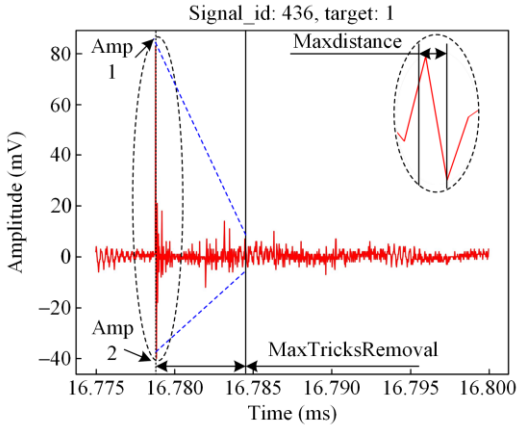


Fig. 9. Details of corona discharge peaks.

To prevent such misclassifications, all peaks within the defined maximum distance are eliminated. Additionally, to further minimize the misdiagnosis risk, all peaks exceeding a predetermined amplitude threshold are removed from the peak set. These measures aim to optimize the signal processing strategy and accurately identify and eliminate interference sources, thereby enhancing the accuracy of PD pattern detection.

However, literatures do not specify the lower limit for peak detection, and establishing an accurate lower threshold is crucial for eliminating unrelated background noise, thereby increasing the accuracy of PD pattern recognition. Initially, peak value features are defined for all peaks within the threshold range [1]–[10]. The LightGBM model is subsequently fitted to the peak features selected for each threshold. The final results, as depicted in Fig. 10, reveal that the classifier achieves the highest accuracy with a lower limit of 5. Consequently, in this study, the lower signal limit is set at 5 to effectively filter out background noise. Figure 11 shows the final noise reduction effect achieved.

To assess the performance of the denoising algorithm proposed in this paper, it is compared with several existing denoising methods. The "NF+DWT" method employs a digital notch filter to remove power line interference (typically 50 Hz) and a DWT to denoise the original signal by decomposing it, filtering out the noise, and reconstructing the useful signal. The "STL" method uses the STL time series decomposition algorithm to decompose the original signal and extract the components that are closely related to the PD activity. The "noise estimation" method achieves denoising by flattening the processed signal, estimating its noise level, and empirically identifying and addressing background noise pulses.

Table I presents a comparative evaluation of the four distinct denoising algorithms using LightGBM. This evaluation involves constructing all the previously mentioned features for determining the lower limit threshold based on the signal data processed through these denoising algorithms, where these features serve as inputs for LightGBM. The MSTL-based denoising algorithm outperforms the other algorithms in terms of accuracy, recall, and specificity, underscoring the superiority of the denoising algorithm introduced in this study.

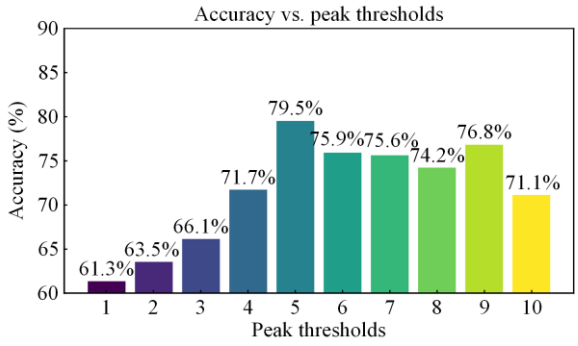


Fig. 10. Threshold selection results obtained based on the LightGBM.

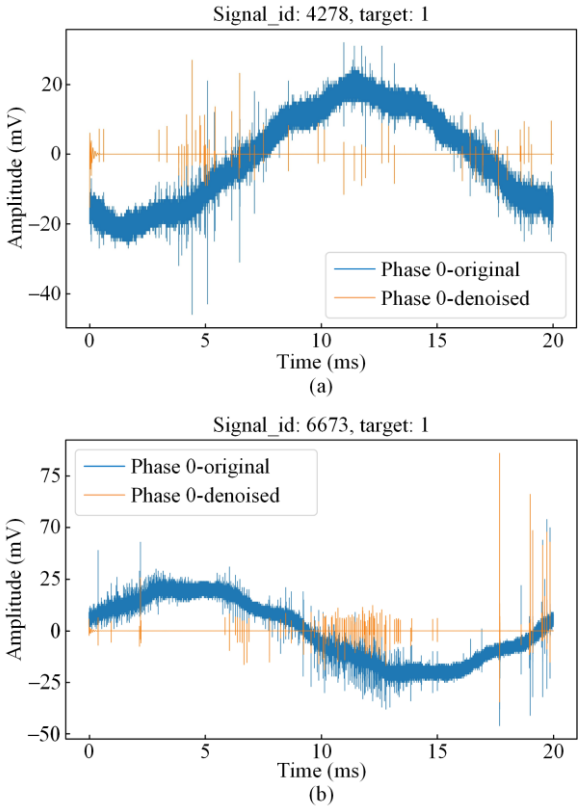


Fig. 11. Example of a voltage signal before and after performing noise reduction. (a) Labeled non-PD. (b) Labeled PD.

TABLE I
COMPARISON AMONG THE EFFECTS OF DIFFERENT DENOISING ALGORITHMS

Model	Accuracy	Recall	Specificity
NF+DWT	0.753	0.633	0.843
STL	0.746	0.624	0.838
Noise estimation	0.767	0.651	0.853
Proposed algorithm	0.795	0.687	0.873

B. Multiscale Feature learning and Data Augmentation Analysis

1) Multiscale Feature Learning

A single PD signal is considered as an example, where the signal length is 800 000. After processing the signal via the noise reduction method described in Section III.A, an equal-length noise-reduced signal is obtained, which is subsequently fed into the designed multi-scale 1D-CNN feature learning network for feature extraction purposes. The extracted features are utilized for the subsequent PD pattern recognition modeling process, thereby underscoring the significance of the performance attained by the multi-scale feature learning network.

A multi-scale feature learning network is trained using the PD dataset introduced in Section II.A. The samples are divided into a training set and a testing set at an ratio of 80%: 20%. Importantly, the numbers of input samples for different categories should be balanced during each training session. Multiple sets of repeated experiments are conducted to improve the reliability of the results.

The specific training strategy is as follows. A Bayesian optimization method based on Gaussian processes is employed to automatically adjust the hyperparameters of the multi-scale 1D-CNN feature learning network. Additionally, it constructs a simple two-layer LSTM neural network as a feature effectiveness evaluator. The objective of hyperparameter optimization is to improve the accuracy of the simple two-layer LSTM in terms of classifying PD samples.

The architecture of the multi-scale 1D-CNN feature learning network comprises three fixed parallel paths, each containing one, two, or three convolutional layers. The kernel size of the convolutional layers in each path corresponds to the step size to ensure that the windows do not overlap and that the temporal integrity of the features is preserved. The optimization process concentrates on adjusting the number of filters and the kernel sizes within the convolutional layers of the parallel structure to ensure that the final feature length adheres to the standard length requirements of deep learning models. The number of filters contained in the last convolutional layer defaults to 512, with both the step size and kernel size established as 1. The expected improvement is defined as an acquisition function and these hyperparameters are optimized across 100 iterations until the achieved performance improvement ceases to be significant. Following each iteration, based on the performance achieved when using the 20% partitioned from the training set as the validation set, the hyperparameters are refined and optimized to closely approximate the optimal parameter combination. The kernel sizes are set in a decremental fashion to effectively capture both global and local features. Table II displays the hyperparameter range settings for the convolutional layers of each parallel path.

TABLE II
RANGES OF HYPERPARAMETER SETTINGS

Path	Convolutional layer	Filter number range	Nuclear size range
1	1	16–128	500–5000
2	1	16–128	100–1000
2	2	16–128	10–100
3	1	16–128	10–50
3	2	16–128	10–50
3	3	16–128	10–50

Table III lists the final hyperparameter optimization results: 1) a 1D-CNN with a kernel size of 4000 and 64 filters; 2) two 1D-CNNs with kernel sizes of 400 and 10, with the first convolutional layer having 32 filters and the second having 64 filters; and 3) three 1D-CNNs, each with kernel sizes of 20, 20, and 10, where the first convolutional layer has 64 filters, the second has 96, and the third has 128 filters. For Bayesian optimization purposes, various numbers of filters and kernel sizes are chosen for the different convolutional layers in each path. A detailed analysis of these results is presented below.

TABLE III
RESULTS OF HYPERPARAMETER OPTIMIZATION

Path	Convolutional layer	Filter number	Nuclear size
1	1	64	4000
2	1	32	400
2	2	64	10
3	1	64	20
3	2	96	20
3	3	128	10

In paths 2 and 3, the later convolutional layers favor a larger number of filters. This suggests that as the network depth increases, additional feature mappings are necessary to capture more abstract temporal information. Deeper network layers require additional filters to augment the capacity of the model to learn complex features. The optimization process generally opts for larger kernel sizes in the initial layer, followed by smaller sizes in subsequent layers, indicating that larger kernels capture a broader range of temporal dependencies at the outset, whereas smaller kernels are better for detecting local or fine-grained features. On the basis of this framework, the LSTM model attains its highest accuracy when equipped with 256 filters. Consequently, the filter count of the final layer is established as 256, as shown in Fig. 12.

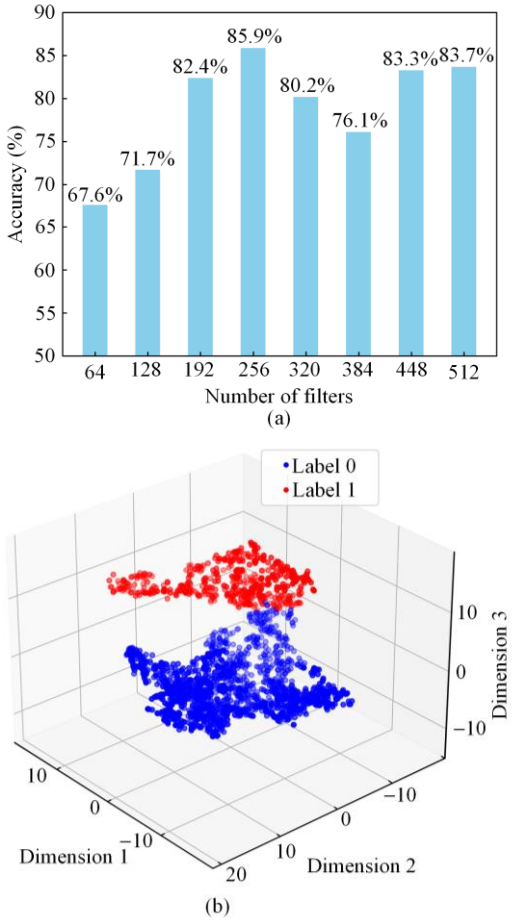


Fig. 12. Results of hyperparameter optimization. (a) The number of filters is determined. (b) 3D graphs of PD signals and non-PD signals.

Figure 12(b) presents a 3D representation of the PD and non-PD signals. This diagram is based on dimensionally reducing the features derived from multi-scale feature learning. A hypothetical dividing hyperplane (diagonal line) exists between the majority of the blue and red points, suggesting that most blue and red points are distinguishable when the proposed 1D-CNN is utilized for feature learning. This classifier-based data categorization method by indicates the statistical significance of the separation between the two classes.

2) Multivariate Data Augmentation Analysis

A single measurement point can simultaneously measure three-phase voltage signals of MV overhead-covered conductors. Given that a PD in one phase can induce variations in the voltage signals of the other two phases, this indicates a strong correlation among the PD signals of the three phases. Therefore, the three-phase signals derived from a single measurement point are analyzed as a cohesive unit to leverage their global features for the final recognition of PD patterns. By grouping and encoding the labels of the three-phase data acquired from the same measurement point, eight distinct data categories are obtained, as illustrated in Table IV. For the 2nd–4th columns in the Table IV, "0" indicates "non-PD" and "1" indicates "PD".

TABLE IV
SCHEMATIC OF THREE-PHASE SIGNAL COMBINATION CODING

Measurement point/phase	A	B	C	Label_code
0	0	0	0	'0'
1	1	1	1	'1'
67	1	1	0	'2'
96	0	0	1	'3'
271	1	0	0	'4'
601	1	0	1	'5'
944	0	1	1	'6'
1994	0	1	0	'7'

A statistical analysis reveals that among the fault samples, 156 exhibit issues across all three phases, accounting for 80.4% of the cases. In contrast, 21 samples have faults in two of the three phases (10.8%), and 17 samples present issues in only one phase (8.8%). The analysis also suggests that the majority of the PD cases occur simultaneously in all three phases, where a PD in one phase affects not only the voltage signals of the other two phases but also implies a high PD probability in them. This highlights the importance of concurrently analyzing the three-phase signals, as exemplified by scenarios such as a tree fall impacting all three phases at a measurement point.

With respect to the overall proportions of samples with and without PD signals, 6.7% of the three-phase data groups contain fault signals, and 93.3% do not, highlighting a class imbalance problem. To address this, data augmentation is performed to achieve an equal number of samples across the different categories, resulting in Counter, i.e., {0: 2710, 1: 700, 2: 700, 3: 700, 4: 700, 5: 700, 6: 700, 7: 700}.

As illustrated in Fig. 13(a), "Sample 1" depicts the original two-dimensional distribution of the data, whereas "Sample 2" illustrates the two-dimensional distribution of the newly generated data produced after applying the data augmentation algorithm. The newly generated data adequately capture the distribution of the original data, suggesting that the new data have successfully retained the relevant properties of the original data. Additionally, the evaluation criterion is employed for time series data augmentation tasks, as outlined in [26]. Fidelity (discriminative score): The generated data must be indistinguishable from the real data. This quantitative evaluation method is applied to further substantiate the effectiveness of the generated data. The discriminative score is 0.128±0.015, demonstrating that the generated data are difficult to distinguish from the original data, confirming their validity. Figure 13(b) shows the relative proportions of various types of data.

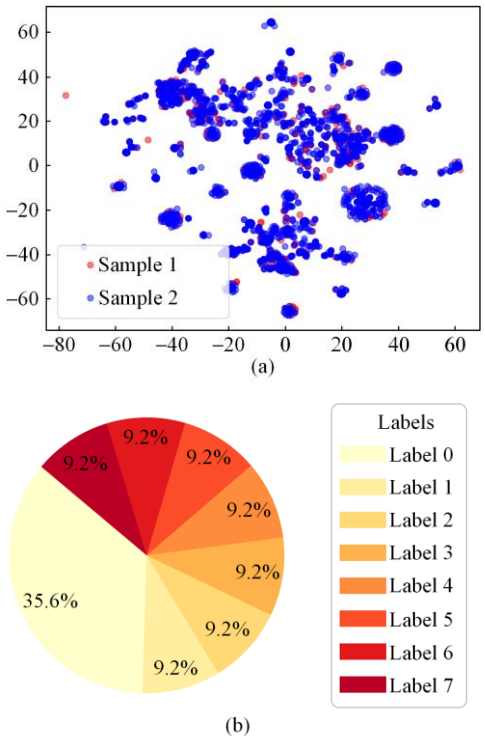


Fig. 13. Results Analysis. (a) T-distributed stochastic neighbor embedding (TSNE) diagram of the enhanced data and the original data. (b) Enhanced data sharing charts.

The final dataset, comprising 7610 samples, is divided into training and test sets at an 80% : 20% ratio. Ensuring a balanced number of samples from each category during each training session is crucial. To bolster the credibility of the experiment, multiple repeated experiments are conducted.

C. Pattern Recognition Results Analysis

1) Algorithmic Ablation

To demonstrate the utility of each component within the overall framework of the algorithm proposed in this paper, ablation experiments are conducted. In these experiments, the impacts of excluding the following components on the resulting algorithmic performance are investigated: the "automatic multi-scale feature learner", the "triple signal utilization scheme," and the "position encoder". Specifically, in the experiment related to the "automatic multi-scale feature learner," it is substituted with the traditional feature construction method from "Kaggle 1st" to verify the feature extraction effectiveness differences between the multi-scale feature learner and traditional feature extraction methods.

As shown in Table V, the utilization of the triple signal has the most substantial influence on the algorithm proposed in this paper, followed by the use of the automatic multi-scale feature extractor and, finally, the position encoder. Based on this ablation study, it can conclude that all three components enhance the performance of the final algorithm. Furthermore, to validate the effectiveness of the dynamic gating mechanism outlined herein, a comparative study is conducted on the single-tower Transformer, the standard linear gated dual-tower Transformer [35], and the dynamic gated dual-tower Transformer proposed in this paper. And in the Table, "w/o" is the abbreviation of "without".

TABLE V
ABLATION RESULTS CONCERNING EACH PART OF THE ALGORITHM

	Accuracy	Recall	Specificity	MCC
w/o 1D-CNN	0.913	0.856	0.949	0.815
w/o three phase	0.867	0.787	0.920	0.721
w/o position encoder	0.921	0.869	0.953	0.832
Proposed algorithm	0.986	0.975	0.992	0.970

As shown in Table VI, the traditional Transformer has the lowest accuracy, while performance improves with the use of the linear gated dual-tower Transformer.

TABLE VI
INVESTIGATING TRANSFORMERS WITH DIFFERENT ARCHITECTURES

	Accuracy	Recall	Specificity	MCC
Single tower Transformer	0.871	0.793	0.923	0.729
Linear gated two-tower Transformer	0.902	0.839	0.942	0.793
Proposed algorithm	0.986	0.975	0.992	0.970

However, because the linear combination operation does not fully leverage the outputs of the two encoders, its effectiveness is inferior to that of the dynamic gated dual-tower Transformer proposed in this paper. In summary, the comparison suggests that the dynamic gated dual-tower Transformer enables the model to adapt more flexibly to different input conditions, thereby enhancing its ability to capture complex data patterns.

2) Analysis of the Algorithmic Results

Training and predictive analyses involving the Transformer-based pattern recognition algorithm are the focus of this section. The achieved classification accuracy is assessed via the cross-entropy loss between the model predictions and the actual labels. In the training strategy, a batch size of 2 is utilized, with tests conducted every five epochs to monitor the resulting performance, and the model with the highest test set accuracy is saved for generalizability and usability purposes.

As shown in Figs. 14 and 15, the model exhibits high accuracy across all labels. All labels yield accuracy rates exceeding 98%, indicating the ability of the model to correctly classify most cases, whether they are positive or negative. From a precision standpoint, the model also demonstrates efficient performance, with labels 3, 4, 6, and 7 achieving ideal precision, indicating that no false positives are contained in these categories. The precision values remain above 92.4% for labels 0, 1, 2, and 5, with labels 2 and 5 achieving 99.3% and 98.6%, respectively, reflecting low rates of false positives. The recall rate, which is indicative of the ability of the model to capture positive instances, exceeds 95% across all labels. The F1 score serves as a comprehensive evaluation metric, and exceptionally high scores are obtained; notably, values of 99.7% and 99.3% are achieved for labels 2 and 5, respectively.

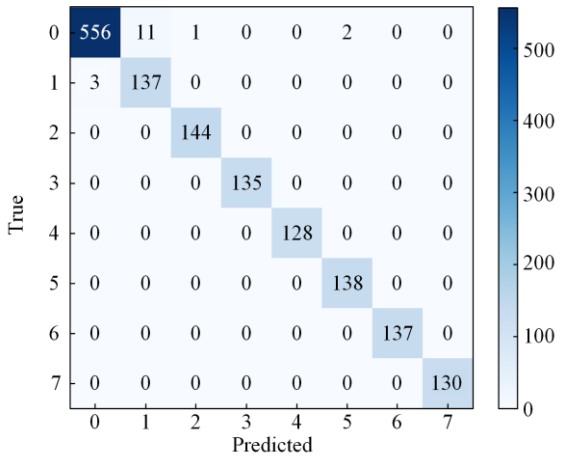


Fig. 14. Confusion matrix of the classification results.

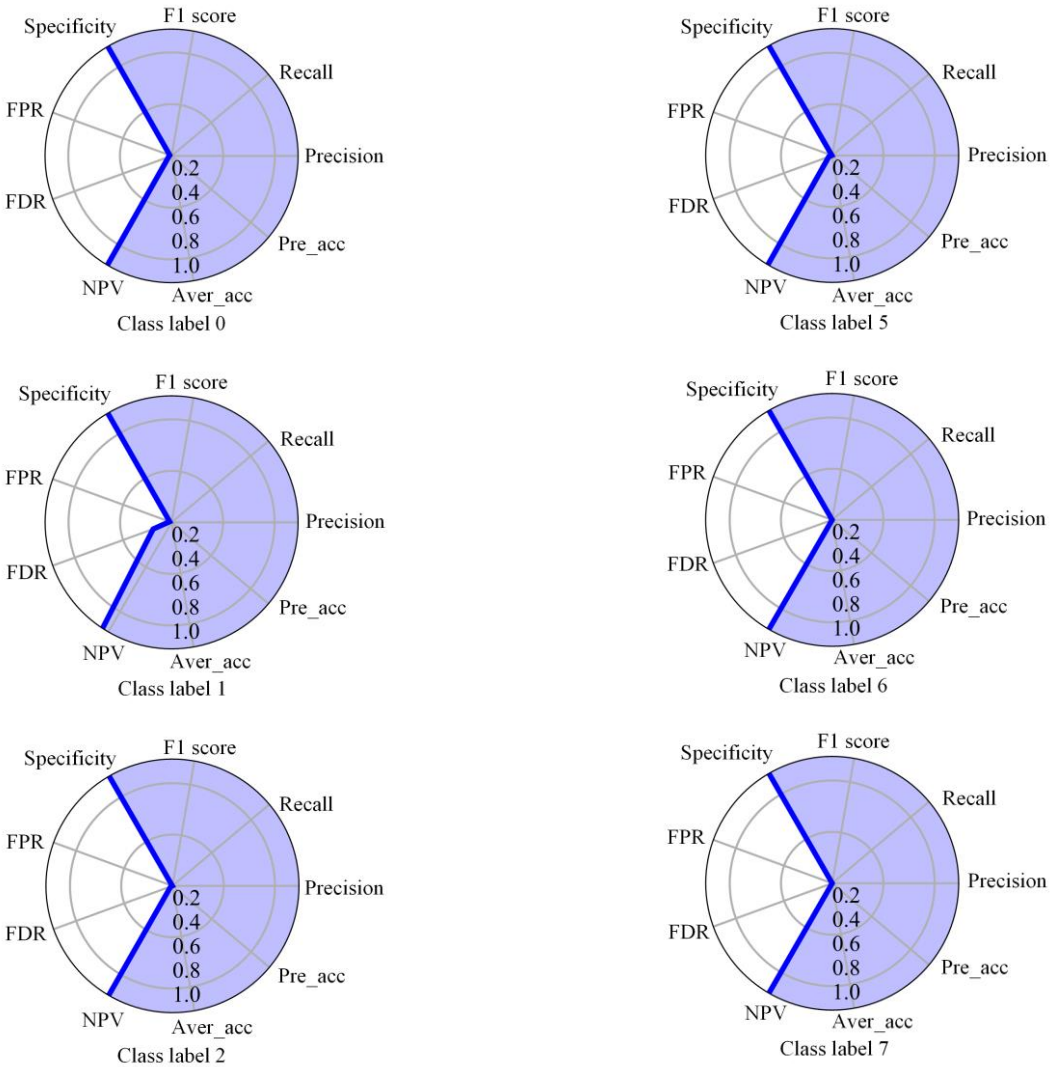


Fig. 15. Evaluation indices of the proposed algorithm.

The model displays exceptional classification capabilities across various categories, particularly in terms of its accuracy and F1 scores. While some false positives are present for certain labels, such as label 1, these areas may be the focus of future optimization work. Moreover, the model excels at capturing positive instances, achieving ideal recall rates for some labels. Overall, the model performs exceptionally well in multiclass classification tasks.

Additional metrics, including specificity, the FPR, the FDR, and the NPV, are calculated for a comprehensive evaluation of the performance achieved by the model, as illustrated in Fig. 15. Specificity reflects the proficiency of the model in correctly identifying negative classes, with values exceeding 99.2% produced across all labels. The FPR are notably low, with the maximum values being 0.7% for label 0 and 0.8% for label 1, indicating a minimal misclassification rate. The FDR is maintained at a low level, with the highest rate at 7.6% for label 1, which is within an acceptable range. The NPV is consistently high, underscoring the

accuracy of the model in terms of predicting negative classes.

The algorithm attains an accuracy value of 98.62%. When the assumption from [12] stating that a fault is present if PD signals appear in any phase is adopted, the accuracy of the algorithm reaches 99.08%. The classification model demonstrates high sensitivity and accuracy for positive samples while effectively handling negative samples. This balance is essential in multi-class classification tasks, particularly in contexts that are sensitive to false positives and negatives. Therefore, the model maintains high accuracy while effectively balancing its predictions for positive and negative samples.

D. Comparison with Other Technologies

The proposed methodology offers a comprehensive, integrated, and intelligent solution for detecting PDs in MV overhead power lines. In the model evaluation, a comparative analysis is conducted with the denoising methods, feature extraction techniques, and classifiers detailed in the literature, as shown in Table VII. The "Kaggle 1st" model employs features derived from pulse construction, with most features being statistics related to pulse counts and heights, and classification is performed via LightGBM [12]. The "DWT+SVM" approach constructs features on the basis of voltage signal energy and employs a support vector machine (SVM) as the classifier [23]. The "STL+LSTM" method uses local weighted regression for time series decomposition, decomposes voltage signals via STL and builds statistical features from the observed residuals for classification with an LSTM neural network [14]. The "DWT+LSTM" method denoises voltage signals via the DWT and uses an LSTM neural network to conduct feature-based classification [11]. The "LMFE" approach accounts for the intra-phase and inter-phase relationships of three-phase signals, learns representative waveforms through cluster analyses, constructs multi-scale features, and employs an RNN model for classification [13].

TABLE VII
COMPARISON BETWEEN THE PROPOSED METHOD AND OTHER METHODS

Method	Accuracy	Recall	Specificity	MCC
DWT+SVM	0.702	0.571	0.806	0.391
Kaggle 1st	0.863	0.781	0.918	0.712
STL-LSTM	0.801	0.695	0.877	0.586
DWT+LSTM	0.872	0.794	0.923	0.731
LMFE	0.881	0.809	0.930	0.751
Proposed	0.986	0.975	0.992	0.970

An analysis based on Table VII yields the following insights. DWT combined with an SVM for classification achieves an accuracy rate of only 70.2%. Conversely, DWT combined with LSTM performs similarly to the noise estimation-based LMFE algorithm. Compared with other methodologies, the MSTL-based algorithm proposed in this study is more accurate. Notably, the proposed method achieves an accuracy value of 98.6%, surpassing those of the other algorithms. Except for LMFE, which considers multichannel feature correlations, the other algorithms do not account for the correlations between three-phase signals. The proposed method, which uses a Transformer, effectively leverages inter-phase signal correlations, achieving a 10.5% increase in accuracy over that of LMFE and outperforming the other models. The proposed method effectively balances the classifications of positive and negative samples, with a 97.5% recognition rate for PD samples, substantially mitigating the associated risks. A comparative study is also conducted on the performance achieved by the proposed algorithm under varying training sample sizes. As shown in Fig. 16(b), with just 60% of the training samples, the algorithm outperforms the traditional "DWT+SVM" and "STL-LSTM" methods across all the metrics. When the training sample proportion reaches 70%, the performance indicators of the proposed method clearly demonstrate its superiority. This finding indicates that the proposed algorithm can maintain high accuracy even in cases with limited training data. The experimental results demonstrate the superior performance of the proposed method.

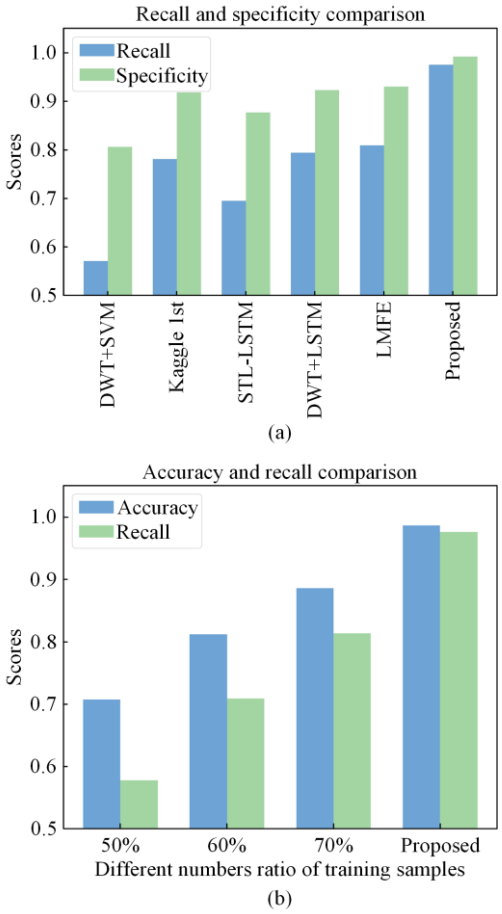


Fig. 16. Experimental results. (a) Comparison among different algorithms. (b) Performance achieved by the proposed algorithm under different numbers of training samples.

IV. Conclusion

In this study, an innovative automatic multiscale feature learner coupled with a Transformer as a classification framework for detecting PD in overhead covered conductors is introduced. The primary features of the method include the following:

1) A multi-seasonal time series decomposition algorithm is developed on the basis of noise frequency analysis, which efficiently removes background noise and retains the critical PD signals;

2) Through an automatic feature learning network, the proposed method intelligently extracts local and global features, minimizes the required manual intervention, and integrates three-phase signals for data augmentation purposes, thereby ensuring the correlation of the three-phase signals in the input global time series;

3) Utilizing a Transformer-based global multichannel pattern recognition framework, this approach captures the temporal dynamics and spatial correlations among the features contained in three-phase signals, offering an effective approach for deeply analyzing the complex structures of multivariate time series. An experimental validation demonstrates that the method in this study achieves 98.6% detection accuracy and 99.2% specificity, significantly surpassing the results of other methods such as STL-LSTM and LMFE. The method achieves a high recognition rate of 97.5% for PD samples, effectively mitigating the risk posed by PD. This significant performance improvement is attributed to its in-depth exploration of multiscale features and signal correlations.

The proposed framework demonstrates exceptional performance in terms of accurately identifying PDs, offering robust theoretical support for the digital operations and maintenance of power distribution equipment. Excelling at handling complex multivariate time series data, the proposed approach holds significant promise for power systems and broader industrial applications.

For future work, the PD detection model will be optimized via the following approaches: 1) integrating signal denoising, feature extraction, and pattern recognition into a cohesive end-to-end model for joint optimization; 2) introducing semi-supervised learning strategies to improve the performance of the model by leveraging unlabeled data; 3) addressing the class imbalance issue at the edge by implementing cost-sensitive mechanisms and incremental learning algorithms; and 4) optimizing the edge computing process via model compression and knowledge distillation techniques to achieve efficient real-time detection. These improvements are expected to enhance the performance and practical applicability of the developed model.

Acknowledgment

Not applicable.

Authors' Contributions

Chunfeng Zhang: conceptualization, methodology, formal analysis, writing original draft preparation, and writing review & editing. Yu Gong: validation and investigation. Yongjun Zhang: conceptualization, methodology, and formal analysis. Siliang Liu: formal analysis and writing original draft preparation. All authors read and approved the final manuscript.

Funding

This work is supported by the Ministry of Education's Industry-University Cooperation and Collaborative Talent Cultivation Project (No. 231101950290232).

Availability of Data and Materials

Not applicable.

Declarations

Competing interests: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

Authors' Information

Chunfeng Zhang received a master's degree in control science and engineering from China Jiliang University in Hangzhou, China, in 2021; he is currently pursuing a Ph.D. degree in electrical engineering at South China University of Technology, Guangzhou, China. His research interests include big data technologies, partial discharge pattern recognition, Bayesian optimization algorithms, and finite-element grid dissection algorithms.

Yu Gong graduated from the Department of Electrical Engineering and Automation, South China University of Technology in July 2004, obtaining a Bachelor's degree. He graduated from the Department of Automatic Control, University of Sheffield, UK in October 2006, with a Master's degree. Currently, he is pursuing a doctoral degree in electronic information engineering at the School of Electric Power, South China University of Technology in Guangzhou, China. His research interests lie in the application of artificial intelligence in power systems and hydro-power management technology.

Yongjun Zhang received the Ph. D. degree in electrical engineering from South China University of Technol-

ogy, Guangzhou, China, in 2004. Currently, he is a professor with the School of Electric Power, South China University of Technology. His main research interests include reactive power optimization, smart energy, and high-voltage direct current transmission.

Siliang Liu received his master's degree in 2019 from the School of Electric Power at South China University of Technology in Guangzhou, China, and is currently pursuing a Ph.D. degree in electrical engineering at South China University of Technology. His research interests include automatic operation and maintenance technology of power systems, power big data, and artificial intelligence algorithms.

REFERENCES

[1] X. Wang, H. Du, and J. Gao *et al*., "Grounding fault location method of overhead line based on dual-axis magnetic field trajectory," *Protection and Control of Modern Power Systems*, vol. 8, no. 1, pp. 1-14, Jan. 2023.

[2] E. Ogliari, M. Sakwa, and M. Palo *et al*., "General machine learning-based approach to pulse classification for separation of partial discharges and interference," *IEEE Sensors Journal*, vol. 23, no. 21, pp. 26839-26849, Nov. 2023.

[3] L.Klein, J.Fulneček, and D.Seidl *et al*., "A data set of signals from an antenna for detection of partial discharges in overhead insulated power line," *Scientific Data*, vol. 10, no. 1, Aug. 2023.

[4] B. Lu, W. Huang, and Q. Dong *et al*., "The study on a new method for detecting corona discharge in gas insulated switchgear," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-8, Nov. 2022.

[5] T. Martinovič and J. Fulneček. "Fast algorithm for contactless partial discharge detection on remote gateway device," *IEEE Transactions on Power Delivery*, vol. 37, no. 3, pp. 2122-2130, Jun. 2022.

[6] F.Ahsan, N. H. Dana, and S. K. Sarker *et al*., "Data-driven next-generation smart grid towards sustainable energy evolution: techniques and technology review," *Protection and Control of Modern Power Systems*, vol. 8, no. 1, pp. 1-42, Jan. 2023.

[7] N. Rosle, N. A. Muhamad, and M. N. K. H. Rohani *et al*., "Partial discharges classification methods in XLPE cable: a review," *IEEE Access*, vol. 9, pp. 133258-133273, Sep. 2021.

[8] J. Long, X. Wang, and W. Zhou *et al*., "A comprehensive review of signal processing and machine learning technologies for UHF PD detection and diagnosis (I): preprocessing and localization approaches," *IEEE Access*, vol. 9, pp. 69876-69904, May 2021.

[9] S. Lu, H. Chai, and A. Sahoo *et al*., "Condition monitoring based on partial discharge diagnostics using machine learning methods: a comprehensive state-of-the-art review," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 27, no. 6, pp. 1861-1888, Dec. 2020.

[10] S. Misák, J. Fulnecek,and T. Vantuch *et al*., "A complex classification approach of partial discharges from covered conductors in real environment," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 24, no. 2, pp. 1097-1104, Apr. 2017.

[11] N. Qu, Z. Li, and J. Zuo *et al*., "Fault detection on Insulated overhead conductors based on DWT-LSTM and partial discharge," *IEEE Access*, vol. 8, pp. 87060-87070, May 2020.

[12] K. Chen, T. Vantuch, and Y. Zhang *et al*., "Fault detection for covered conductors with high-frequency voltage signals: from local patterns to global features," *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 1602-1614, Mar. 2021.

[13] C. Huang, S. Ding, and S. Li *et al*., "LMFE: learning-based multiscale feature engineering in partial discharge detection," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 5, pp. 1-9, May 2022.

[14] M. Dong and J. Sun, "Partial discharge detection on aerial covered conductors using time-series decomposition and long short-term memory network," *Electric Power Systems Research*, vol. 184, Jul. 2020.

[15] Z. Lei, F. Wang, and C. Li, "A denoising method of partial discharge signal based on improved SVD-VMD," *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 30, no. 5, pp. 2107-2116, Oct. 2023.

[16] B. Chen, Y. Hu, and L. Wu *et al*., "Partial discharge pulse extraction and interference suppression under repetitive pulse excitation using time-reassigned multisynchrosqueezing transform," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1-9, Oct. 2023.

[17] N. Xu, H. B. Gooi, and L. Wang *et al*., "Loop optimization noise-reduced LSTM based classifier for PD detection," *IEEE Transactions on Industry Applications*, vol. 59, no. 1, pp. 392-402, Jan. 2023.

[18] J. Pradeepkumar, M Anandakumar, and V. Kugathasan *et al*., "Toward interpretable sleep stage classification using cross-modal transformers," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 32, pp. 2893-2904, Aug. 2024.

[19] Y. Xi, X. Tang, and Z. Li *et al*., "Fault detection and classification on insulated overhead conductors based on MCNN-LSTM," *IET Renewable Power Generation*, vol. 16, no. 7, pp. 1425-1433, Jan. 2022.

[20] M. H. Wang, S. D. Lu, and R.-M. Liao, "Fault diagnosis for power cables based on convolutional neural network with chaotic system and discrete wavelet transform," *IEEE Transactions on Power Delivery*, vol. 37, no. 1, pp. 582-590, Feb. 2022.

[21] M. Florkowski, "Anomaly detection, trend evolution, and feature extraction in partial discharge patterns," *Energies*, vol. 14, no. 13, Jan. 2021.

[22] H. Shang, F. Li, and Y. Wu, "Partial discharge fault diagnosis based on multi-scale dispersion entropy and a hypersphere multiclass support vector machine," *Entropy*, vol. 21, no. 1, Jan. 2019.

[23] X. Peng, J. Li, and G. Wang *et al*., "Random forest based optimal feature selection for partial discharge pattern recognition in HV cables," *IEEE Transactions on Power Delivery*, vol. 34, no. 4, pp. 1715-1724, Aug. 2019.

[24] S. Zeraatkar and F. Afsari, "Interval-valued fuzzy and intuitionistic fuzzy-KNN for imbalanced data classification," *Expert Systems with Applications*, vol. 184, Dec. 2021.

[25] O. Trull, J. C. García-Díaz, and A. Peiró-Signes, "Multiple seasonal STL decomposition with discrete-interval moving seasonalities," *Applied Mathematics and Computation*, vol. 433, Nov. 2022.

[26] P. Srinivasan and W. J. Knottenbelt, "Time-series transformer generative adversarial networks," *arXiv*: 2205.11164, May 2022.

[27] S. Shao, P. Wang, and R. Yan, "Generative adversarial networks for data augmentation in machine fault diagnosis," *Computers in Industry*, vol. 106, pp. 85-93, Apr. 2019.

[28] Y. Li, M. Zhang, and C. Chen, "A deep-learning intelligent system incorporating data augmentation for short-term voltage stability assessment of power systems," *Applied Energy*, vol. 308, Feb. 2022.

[29] Y. Li, J. Cao, and Y. Xu *et al*., "Deep learning based on transformer architecture for power system short-term voltage stability assessment with class imbalance," *Renewable and Sustainable Energy Reviews*, vol. 189, Jan. 2024.

[30] C. Liu, R. Antypenko, and I. Sushko I *et al*., "Intrusion detection system after data augmentation schemes based on the VAE and CVAE," *IEEE Transactions on Reliability*, vol. 71, no. 2, pp.1000-1010, Apr. 2022.

[31] A. Desai, C. Freeman, and Z. Wang *et al*., "TimeVAE: A variational auto-encoder for multivariate time series generation," *arXiv*: 2111.08095, Dec.2021.

[32] X. Feng, Q. M. Jonathan Wu, and Y. Yang *et al*., "An autuencoder-based data augmentation strategy for generalization improvement of DCNNs," *Neurocomputing*, vol. 402, pp. 283-297, Aug. 2020.

[33] J. Dai, Q. Guo, and G. Wang *et al*., "An optimized method for variational autoencoders based on Gaussian cloud model," *Information Sciences*, vol. 645, Oct. 2023.

[34] A. Biswas, R. Vasudevan, and M. Ziatdinov *et al.*, "Optimizing training trajectories in variational autoencoders via latent Bayesian optimization approach," *Machine Learning: Science and Technology*, vol. 4, no. 1, Feb. 2023.

[35] S. Xiao, S. Wang, and Z. Huang *et al*., "Two-stream transformer network for sensor-based human activity recognition," *Neurocomputing*, vol. 512, pp. 253-268, Nov. 2022.