Energy Engineering

Tech Science Press

Doi:10.32604/ee.2025.073947

ARTICLE

Power Grid Monitoring Alarm Events Identification Based on Large Language Model

Qiang Xu[1,*], Leyao Cong[1], Jianing Wang[1], Xingyu Zhu[1], Shaojun Cui[1], Guoqiang Sun[2] and Xueheng Shi[2]

[1]State Grid Wuxi Power Supply Company, State Grid Jiangsu Electric Power Co., Ltd., Wuxi, 214061, China
[2]School of Electrical and Power Engineering, Hohai University, Nanjing, 211100, China

*Corresponding Author: Qiang Xu. Email: wer0041@163.com

Received: 29 September 2025; Accepted: 12 November 2025; Published: 12 July 2026

ABSTRACT: Power system faults can trigger a massive influx of complex alarm signals to the operation and maintenance center, posing significant challenges for dispatchers in accurately identifying the underlying faults. To address the issues of sample imbalance and low accuracy in traditional power grid monitoring alarm event identification methods, a power grid monitoring alarm event identification method based on BERT large language model is proposed. Firstly, information entropy is employed to filter effective monitoring alarm signals, and the k-means clustering algorithm is used to group all alarm signals into different event types, forming the initial power grid monitoring alarm event samples. Then, to mitigate the issue of sample imbalance in power grid monitoring alarm events, a pre-trained SimBERT model is proposed to augment minority class samples, thereby reducing the imbalance ratio. Finally, the augmented samples of power grid monitoring alarm events are used to fine-tune the bidirectional encoder representation from transformer (BERT) model. A mix-training optimization strategy is adopted during fine-tuning to ultimately obtain the power grid monitoring alarm event identification model. Case study results demonstrate that the proposed model in this paper achieves superior identification precision of power grid monitoring alarm events compared to traditional deep learning methods.

KEYWORDS: Power grid monitoring alarm event; BERT; information entropy; SimBERT; mix-training

1 Introduction

When operational abnormalities or faults occur in a power grid, the operation and maintenance center typically receives a large number of text-based monitoring alarm signals within a short period [1]. These alarm signals provide critical evidence for operators to determine the nature of events. Upon receiving these alarms, operators should respond promptly to assess the power grid monitoring alarm events (PGMAE) and take appropriate action to restore normal grid operation. Traditional PGMAE assessment primarily relies on operators manually applying alarm identification rules and personal expertise, an approach that is prone to misjudgments and omissions [2]. Consequently, there is an urgent need to develop methods capable of accurately analyzing and identifying of PGMAE.

Conventional methods for PGMAE identification include support vector machine (SVM) [3], random forest (RF) [4] and so on. With the rapid development of artificial intelligence (AI), particularly breakthroughs in deep learning methods for natural language processing (NLP), accurate and rapid identification of PGMAE has become feasible [5]. Refs. [6,7] utilize hierarchical attention networks (HAN) for document

classification, demonstrating performance superior to traditional classification models. Ref. [8] integrates convolutional neural networks (CNN) with long short-term memory (LSTM) networks to form a LSTM-CNN alarm diagnosis model, thereby enhancing diagnostic accuracy. Ref. [9] combines grid knowledge bases with deep learning methods to improve the robustness of alarm events identification. However, these methods fail to effectively learn the extensive domain-specific terminology prevalent in the power industry, resulting in relatively low identification precision.

Furthermore, sample imbalance also constrains the application of deep learning methods in power grids [10]. Deep learning typically requires large sample sizes for training, which are often scarce in power grid contexts, consequently reducing model generalization capability. Moreover, the occurrence probabilities of different PGMAE in power grids vary, leading to significant disparities in sample quantities across events [11]. Easy data augmentation (EDA) [12] offers a straightforward technique to mitigate the impact of sample imbalance on deep learning algorithms. It constructs semantically similar samples by means of random swapping, random deletion, random insertion, and random substitution. However, the high randomness inherent in its construction process may introduce excessive noise, potentially affecting the training outcomes [13]. Ref. [14] applies noise injection and GAN-based methods to address data scarcity challenges in smart grid anomaly detection and load forecasting. Ref. [15] established a three-stage sparse data augmentation framework using a sequence-to-sequence enhanced super-resolution generative adversarial network to enhance system observability. Ref. [16] employed EDA method to augment malware detection data, reaching better training results compared to data trained by a variational autoencoder. Ref. [17] proposed an improved generative adversarial network integrated with transfer learning to improve data generation performance even with limited data. However, the aforementioned methods might inadvertently amplify noise existing in the samples, causing interference with the training results.

In recent years, advancements in large language model (LLM) have garnered increasing attention, achieving higher accuracy than traditional AI algorithms across multiple NLP tasks [18,19]. The emergence of the BERT model [20], which is based on the transformer [21] architecture, has significantly improved identification accuracy on numerous datasets. Ref. [22] developed PowerBERT, a power domain-specific model, by incorporating a large corpus of power industry texts during training. Ref. [23] enhanced the identification results of unstructured texts in centralized dispatching systems by integrating the RoBERTa model, trained with global masking, with bidirectional long short-term memory (Bi-LSTM) and bidirectional gated recurrent unit (Bi-GRU) networks. Ref. [24] achieved precise sentiment text classification using a BERT-BiLSTM model. However, references mentioned primarily leverage LLMs to generate semantic text vectors, which are then fed into neural networks for training, rather than directly fine-tuning the LLMs on given texts to obtain models readily applicable to downstream tasks. Furthermore, research on applying LLM specifically to the analysis and identification of PGMAE remains relatively limited.

However, directly applying off-the-shelf LLMs to PGMAE identification faces several domain-specific hurdles: (1) Information Redundancy and Noise: Raw alarm streams contain massive redundant and auxiliary signals that can obscure the core fault pattern. (2) Severe Class Imbalance: The natural occurrence frequency of different fault types leads to extreme sample size disparities. (3) Semantic Sensitivity: Minor perturbations in alarm wording should not alter the event type, requiring robust semantic understanding beyond lexical matching. Existing studies often address these challenges in isolation. To bridge this gap, this paper proposes a novel, integrated framework that systematically tackles these challenges. Our main contributions are threefold:

We introduce a task-specific data preprocessing pipeline that combines information entropy (IE) and k-means++ clustering algorithm [25]. This pipeline is specifically designed to distill critical fault information

from the noisy, high-volume alarm text streams characteristic of power system, a challenge not adequately addressed by generic text preprocessing methods.

We present a holistic strategy to combat sample imbalance, integrating SimBERT-based semantic augmentation with a novel mix-training protocol. This strategy is crucial for the power grid domain where minority fault types are critical yet underrepresented, going beyond simple oversampling or cost-sensitive learning. Then the samples are standardized according to LLM input requirements and fed into the pre-trained BERT model for training. During the training process, multiple rounds of mix-training are conducted using randomly split training sets to improve the model's effectiveness.

We demonstrate through extensive experiments that our end-to-end framework significantly outperforms not only general-purpose machine learning and deep learning models but also existing state-of-the-art methods specifically designed for power systems, achieving superior identification accuracy for both majority and minority classes.

2 PGMAE Samples Augmentation

2.1 Formation of PGMAE Samples

Power grid alarm information is typically trigger-based, issued when an abnormality is detected. To rapidly identify fault information from numerous signals, we collect monitoring information based on a time scale to form initial PGMAE samples and apply IE combined with k-means++ clustering to derive the final PGMAE samples.

When an alarm event occurs in the power system, all relevant alarm signals will be sent to monitoring center within 10 s. To ensure comprehensive signal capture, we collected all alarm signals from 15 s before to 15 s after the event occurrence. All alarm signals within this 30-s window are then divided into 10 triplets at 3-s intervals, as denoted by Eq. (1), forming a corresponding alarm signal document S. The IE of document S according is then calculated according to Eq. (2).

$$S = \left[\left(t_1, m_1, c_1 \right), \left(t_2, m_2, c_2 \right), \cdots, \left(t_{10}, m_{10}, c_{10} \right) \right] \tag{1}$$

$$H\left(S \right) = -\sum_{r=1}^{m_{\max}} \mathrm{P}\left(r \right) \ln \mathrm{P}\left(r \right) \tag{2}$$

where S is the monitoring information document; t is a 3-s time period; c_i contains all alarm signals within time period t_i; m is the number of alarm signals within time period t_i; $H(S)$ is the IE of document S; $m_{\max}$ is the maximum number of divided states, equal to the maximum value of m_i; r is the number of alarm signal counts across10 triplets; P(r) is the probability of r occurring in S.

Triplet sets in document S are deleted sequentially, and the IE is recalculated after each deletion. The triplet set whose removal causes the largest change in IE is identified as the centroid. Subsequently, triplet sets are iteratively removed from the periphery until the IE of the remaining document, denoted as $H'(S)$, falls below the original IE $H(S)$. The resultant document S' is considered to contain the condensed, effective monitoring alarm information. Given that alarm signals consist of standardized information with relatively fixed description for the same event, a statistical segmentation method is adopted for text segmentation and analysis. The co-occurrence information coefficient $M(\delta_i, \delta_j)$ for two adjacent Chinese characters δ_i and δ_j is defined as their pointwise mutual information (PMI), calculated in Eq. (3).

$$M\left(\delta_i, \delta_j \right) = \log \frac{\mathrm{P}\left(\delta_i, \delta_j \right)}{\mathrm{P}\left(\delta_i \right) \mathrm{P}\left(\delta_j \right)} \tag{3}$$

where $P(\delta_i, \delta_j)$ is the probability of δ_i and δ_j appearing adjacent to each other in the text; $P(\delta_i)$ and $P(\delta_j)$ represent the individual probabilities of Chinese characters δ_i and δ_j appearing in the text, respectively. We include character pairs with $M(\delta_i, \delta_j) > 0$ into the feature set, indicating that the two characters co-occur more frequently than by chance. This threshold was empirically validated on a held-out validation set of 1000 alarm messages, where it achieved the highest F1-score for event type clustering compared to thresholds of 0.1 and 0.2.

Given the structured and domain-specific nature of Chinese alarm texts, we adopt character-level tokenization to capture fine-grained semantic units without relying on external dictionaries. Each alarm message is treated as a sequence of Chinese characters.

We employ a modified TF-IDF [26] scheme that incorporates smoothing to avoid zero IDF values for characters appearing in all documents. The weight for a character δ_i in document S' is computed as Eq. (4).

$$W_i = \frac{f(\delta_i, S') \times \log\left(\frac{N_d}{n_i+1}\right)}{\sqrt{\sum_{j=1}^{n_f} f(\delta_j, S') \times \log\left(\frac{N_d}{n_j+1}\right)}} \tag{4}$$

where $f(\delta_i, S')$ is the frequency of character δ_i in document S'; N_d is the total number of documents; n_i is the number of documents containing character δ_i; n_f is the total number of characters in document S. The weights are L_2-normalized to form the document vectors.

Once documents are converted into vectors, we apply the k-means++ clustering method to group all documents. All documents within a single cluster constitute one PGMAE type. By manually analyzing the documents in different clusters, the practical meaning of monitoring events in each cluster is determined, thereby defining the names for the different PGMAE type.

2.2 SimBERT

The SimBERT model is an optimized variant of the BERT large language model. To address BERT's limitation in text generation, SimBERT incorporates strengths from the unified language model (UniLM) [27] in natural language generation (NLG). Specifically, it replaces the bidirectional masking mechanism used in BERT pre-training with a unidirectional masking mechanism for the input and target texts. This enhances the model's focus on the target text, strengthening its ability to generate semantically similar texts.

During pre-training, the SimBERT model treats the input text and target text as a pair of similar texts. In the same batch, it simultaneously trains the sequences [CLS]<input text>[SEP]<target text>[SEP] and [CLS]<target text>[SEP]<input text>[SEP]. The vectors of all [CLS] tokens across batches are assembled into a matrix. Each row of this matrix undergoes L_2 normalization, resulting in a normalized matrix $\mathbf{V}$. The text similarity matrix $\mathbf{S}$ is then computed using Eq. (5). After masking the diagonal elements of matrix $\mathbf{S}$, the softmax probability for each row of $\mathbf{S}$ is calculated according to Eq. (6). This enabled model to accurately identify the similar texts corresponding to an input text from a set of candidates.

$$\mathbf{S} = a\mathbf{V}\mathbf{V}^{\mathrm{T}} \tag{5}$$

$$\hat{y}_{jk} = \frac{e^{y_{jk}}}{\sum_{l=1, l\neq j}^{b} e^{y_{jl}}} \tag{6}$$

where a is a proportionality constant(set to 30); b is the batch size; y_{jk} is the element in the j-th row and k-th column of matrix $\mathbf{S}$; $\hat{y}_{jk}$ is the element in the j-th row and k-th column of the matrix after softmax normalization; $j, k = 1, 2, \ldots, b$, and $j \neq k$; e is the base of the natural logarithm.

This formulation effectively converts the learning task for similar texts into a classification task. Non-similar samples within each batch serve as negative samples, which enhances the similarity between genuinely similar samples and improves the model's capability to generate high-quality similar samples.

2.3 PGMAE Samples Augmentation Based on SimBERT

This section details the sample augmentation process based on the SimBERT model, following a strict protocol to prevent data leakage and overfitting.

First, the entire dataset of original PGMAE samples is randomly split into a temporary training set (80%), a validation set (10%), and a test set (10%). The SimBERT augmentation process is applied exclusively to the temporary training set. The validation and test sets remain completely untouched, containing only original, human-annotated samples. This ensures that the model evaluation is performed on genuine data, providing a reliable measure of its generalization capability.

For each minority event type in the temporary training set, we determine the number of samples to be generated per original sample n_a, based on a preset target sample size. The SimBERT model then generates a pool of no less than $(2n_a + 1)$ similar samples per original sample to allow for selective filtering.

To mitigate the risk of including near-duplicates that could lead to overfitting, we first remove any generated sample that is string-identical to any sample in the entire original dataset (including training, validation, and test sets). We then calculate the cosine similarity between the vector representation of each generated sample and the vector of its original source sample, according to Eq. (7). The generated samples are sorted in descending order of similarity. Instead of selecting the top n_a samples indiscriminately, we chose the top n_a samples that exhibit a cosine similarity below a carefully chosen threshold of 0.95. This ensures semantic fidelity while explicitly avoiding the inclusion of near-duplicates. The final augmented training set is formed by combining these carefully selected generated samples with the original temporary training set.

$$C = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \times \|\mathbf{B}\|} \tag{7}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the vector representations of the original sample and the generated sample vector; C is the cosine similarity between $\mathbf{A}$ and $\mathbf{B}$.

The generated power grid monitoring alarm event samples are then mixed with the original monitoring alarm event samples to form the complete set of power grid monitoring alarm event samples for large language model training.

3 Identification of PGMAE Based on Large Language Models

This section describes the fine-tuning of the pre-trained BERT model for PGMAE identification. By constructing PGMAE samples and using different event types, "sample-label" pairs are formed. The PGMAE labels served as the target for model identification. By training the model on these "sample-label" pairs, a fine-tuned PGMAE identification model is developed. To ensure the model fully learns the information within the PGMAE samples, a mix-training optimization method is employed. Based on the obtained PGMAE identification model, calculate the probability for each event label corresponding to the input sample, so as to take the event type with the highest predicted probability is taken as the predicted event type.

3.1 Identification of PGMAE Based on BERT

The BERT model, released by Google, has achieved breakthroughs in multiple NLP fields. This paper employs the BERT large language model to identify PGMAE, and optimizes the model training process through mix-training to improve analytical accuracy.

During pre-training process, BERT model is primarily trained on two tasks: Masked Language Model (MLM) and Next Sentence Prediction (NSP). MLM enables BERT to better learn vocabulary and grammatical structures, while NSP helps it learn inter-sentence relationships and language coherence [28]. Upon receiving input text, BERT performs three layers of embedding: token embedding, segment embedding, and position embedding. Token embedding mainly encodes each word in the input text and converts it into a multidimensional vector. Segment embedding, primarily used in NSP tasks, distinguishes different sentences in the input text. For the PGMAE identification task, all segment embeddings are set to 0. Position embedding encodes the position of each word within the input text.

The core architecture of the BERT model is a multi-layer transformer, as shown in Fig. 1. Each layer contains two core sub layers: a multi-head attention mechanism and a feed-forward network (FFN). The multi-head attention mechanism, composed of multiple layers of single attention layers, captures contextual relationships within the input text, allowing the model to excel without strict reliance on word order [29]. The input to the self-attention layer consists of the **Q** and **K** matrices with dimension d_k, and the **V** matrix with dimension d_v. The output attention weights are calculated based on Eq. (8). The multi-head mechanism executes the above process in parallel h times, with each head using different weight matrices to capture contextual information from different representation subspaces. The FFN is a simple two-layer fully-connected network that independently transforms each token's representation, as shown in Eq. (9).

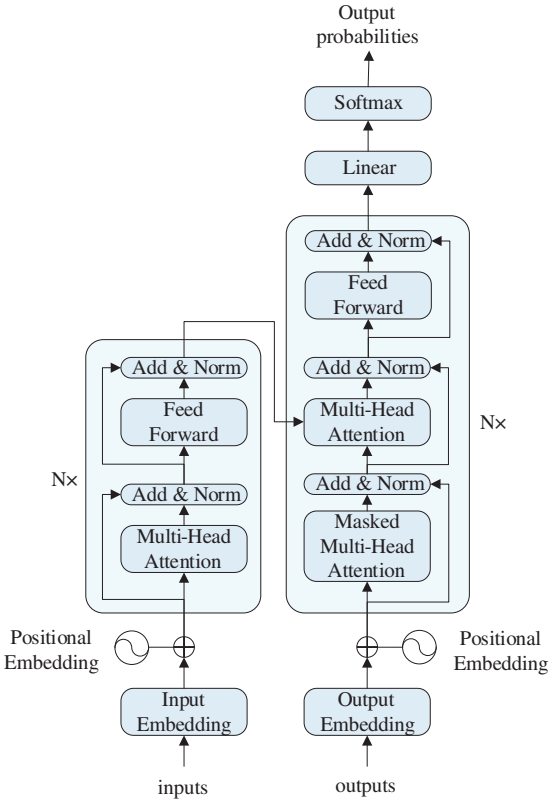

Figure 1: The transformer architecture

$$Att\left(\mathbf{Q},\mathbf{K},\mathbf{V}\right)=\text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\mathbf{V} \tag{8}$$

$$\text{FFN}\left(E\right)=\text{ReLU}\left(E\mathbf{W}_1+b_1\right)\mathbf{W}_2+b_2 \tag{9}$$

where E is a token vector; $\mathbf{W}_1$ and $\mathbf{W}_2$ are weight matrices; b_1 and b_2 are bias.

This paper fine tunes the pre-trained BERT model using the augmented PGMAE samples to train a PGMAE identification model. The structure of BERT-based PGMAE identification model is shown in Fig. 2. When a PGMAE sample is input, BERT first tokenizes the sample and prepends the [CLS] token. All tokens are then embedded into multidimensional vectors for subsequent processing. After passing through the multi-head attention and FFN layers, the embeddings are transformed into vectors whose dimension equals to the number of fault types. The vector corresponding to the initial [CLS] token is passed through a softmax function, converting its values into a probability distribution over the possible fault types. Each value in this output vector represents the probability of the input sample belonging to the corresponding fault type. The fault type with the highest probability is selected as the model's identification result. The specific version of BERT in this paper is BERT_base_chinese, which has 110M parameters. Based on the obtained PGMAE identification model, input monitoring alarm events are identified, and their corresponding event types are determined.

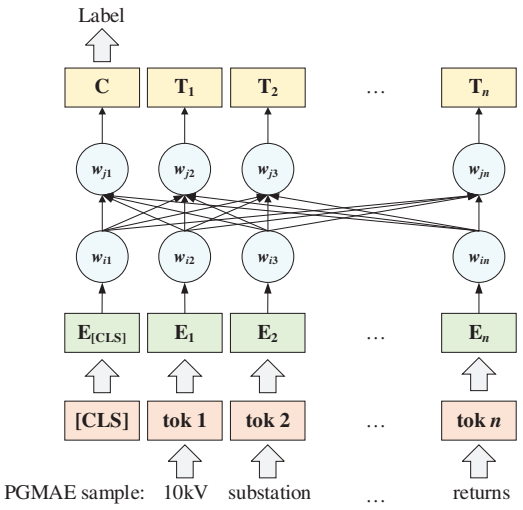

Figure 2: Structure of PGMAE identification based on BERT

3.2 Mix-Training Optimization Strategy

This section describes mix-training strategy used to optimize the BERT model training process. The mix-training strategy is designed to prevent overfitting and enhance model generalization by exposing the model to multiple, stochastically partitioned versions of the training data, acting as a form of data-level ensemble during fine-tuning. The specific mix-training process is illustrated in Fig. 3.

First, the obtained PGMAE samples are randomly divided into training, validation, and test sets in an 8:1:1 ratio, denoted as sample set R_1. This random partitioning is repeated multiple times to obtain sample sets R_2, R_3, ..., R_n. Sample set R_1 is then used to fine-tune the pre-trained BERT model for multiple iterative epochs, resulting in the fine-tuned model 1. Subsequently, sample set R_2 is used to further fine-tune model 1 for the same number of epochs, yielding fine-tuned model 2. This process continues iteratively.

After $n-1$ rounds, fine-tuned model $n-1$ is obtained. Finally, sample set R_n is used to fine-tune model $n-1$, producing the final PGMAE identification model. This sequential fine-tuning on different data subsets encourages the model to learn robust features that are invariant to specific data partitions, addressing the variance inherent in single data splits and reducing the risk of overfitting to peculiarities of any particular training set composition. Based on the final PGMAE identification model, when an alarm event occurs in the power grid, the model can effectively identify the event type, improving the work efficiency of operation and maintenance personnel and ensuring the safe and stable operation of the power grid.

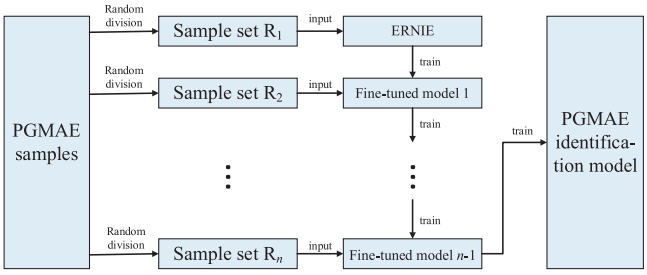

Figure 3: Process of mix-training of model

4 Results

4.1 Platform Configuration

We constructed the experimental setup using a Python-based model training environment. The hardware configuration included an Intel Xeon Silver 4310 CPU (2.10 GHz), and two NVIDIA GeForce RTX A6000 with 48 GB memory. The software environment was built on Baidu's paddlepaddle framework, using python 3.11.11, paddlehub 2.4.0 and paddlepaddle-gpu 3.0.0b1. To ensure reproducibility, the random seed was fixed to 42 for all experiments, including data splitting and model initialization. The training parameters are shown in Table 1.

Table 1: Training parameters

Training parameters	Value
Batch size	32
Maximum sequence length	512
Learning rate	5e−5
Total epochs	9
Optimizer	Adam
Loss function	Cross-entropy
Random seed	42
Learning rate scheduler	Linear with warmup (10% of steps)

We used the cross-entropy loss function for this multi-class identification task. The Adam optimizer was employed with its default parameters (beta1 = 0.9, beta2 = 0.999, epsilon = 1e−8). A linear learning rate scheduler with warmup was applied for the first 10% of the training steps to stabilize training. Early stopping was not used, as we trained for a fixed number of epochs based on validation performance. The model was

fine-tuned for 9 epochs, with each epoch involving a full pass over the training data. The validation set was used to monitor the training process and help prevent overfitting.

4.2 Establishment of Experimental Evaluation Indicators

The model's identification performance is evaluated by constructing a confusion matrix to calculate various assessment indicators, including accuracy, precision, recall and F1-score. The confusion matrix is a square matrix of order m, which equals the number of PGMAE types. The non-diagonal elements c_{ij} represent the number of PGMAE samples whose true type is i but are identified as type j, that is, the number of misidentified events. The diagonal elements c_{ii} represent the number of PGMAE whose true type is i and are correctly identified as type i, that is, the number of correctly identified events. Given that this case study involves 9 PGMAE types, a multi-class identification problem, and significant sample size disparities exist between types, precision (P_p), recall (R_p), and F1-score are adopted as evaluation indicators for each PGMAE type to provide a clearer view of model performance under class imbalance [30]. For the overall evaluation of the model's predictions, accuracy (A), macro-precision (P), macro-recall (R), and macro F1-score (F1) are employed. Macro-averaging gives equal weight to each class regardless of its sample size, making it suitable for imbalanced datasets. The relevant calculation formulas are provided in Eqs. (10)–(13).

$$\mathrm{A} = \frac{\sum_{p=1}^{m_a} c_{pp}}{N} \tag{10}$$

$$\begin{cases} \mathrm{P} = \frac{1}{m_a} \sum_{p=1}^{m_a} \mathrm{P}_p \\ \mathrm{P}_p = \frac{c_{pp}}{\sum_{q=1}^{m_a} c_{qp}} \end{cases} \tag{11}$$

$$\begin{cases} \mathrm{R} = \frac{1}{m_a} \sum_{p=1}^{m_a} \mathrm{R}_p \\ \mathrm{R}_p = \frac{c_{pp}}{\sum_{q=1}^{m_a} c_{pq}} \end{cases} \tag{12}$$

$$\mathrm{F1} = \frac{1}{m_a} \sum_{p=1}^{m_a} 2 \times \frac{\mathrm{P}_p \times \mathrm{R}_p}{\mathrm{P}_p + \mathrm{R}_p} \tag{13}$$

where N is the total number of samples; m_a is the number of alarm event types.

4.3 Experimental Results and Analysis

This case study utilizes two years of PGMAE data from regional power grid in Eastern China, collected between January 2022 and December 2023, comprising approximately 50,000 raw alarm messages. The data included timestamps, device IDs and alarm text descriptions. The samples first underwent preprocessing, including operations such as stop words removal. We applied text cleaning to remove special characters, numbers, and uniformize casing. Chinese text segmentation was performed using character-level tokenization, as described in Section 2.1. Cluster analysis was then performed on the samples. Based on the clustered results and actual PGMAE signal specifications, the sample event types were labeled. Finally, the samples were standardized and stored in form of <type label>\t<event text>. After applying IE and k-means++ clustering, 9 PGMAE types were identified. The resulting sample event types and their corresponding sample sizes are shown in Table 2. As evident from Table 2, significant class imbalance exists. Majority-class samples (e.g., single-phase faults and phase-to-phase faults) can exceed ten times the quantity of minority-class samples

(e.g., transformer body faults). Therefore, augmentation of the minority class samples was performed. After augmentation, the quantity of each minority-class samples exceeded 1000 entries, aiming to reduce the imbalance and improve the identification accuracy.

Table 2: Event types and corresponding sample sizes

Event number	Event types	Sample size
1	Bus fault	435
2	Single-phase fault (successful reclosure)	3658
3	Single-phase fault (unsuccessful reclosure)	5896
4	Phase-to-phase fault	2436
5	Transformer electrical fault	621
6	Transformer body fault	457
7	Transformer on-load voltage regulation fault	237
8	Capacitor/reactor fault	1395
9	Grounding transformer fault	468

To validate the effectiveness of the proposed model, its performance was compared with that of traditional artificial intelligence algorithms, as shown in Table 3. Among machine learning methods, the SVM model and RF were selected for comparison. For deep learning methods, the CNN model and Bi-LSTM were chosen as baseline models. The proposed model employs a mix-training approach to optimize the BERT fine-tuning process, consisting of 3 rounds with 3 iterations per round, where each iteration builds upon the previous model state. In contrast, standard BERT fine-tuning involves only 9 consecutive iterations starting directly from the base pre-trained model.

Table 3: Comparison of identification results between different models

Model	Accuracy/%	Precision/%	Recall/%	F1-Score/%
SVM	88.20	88.17	88.20	88.15
RF	91.18	91.52	91.08	91.18
CNN	92.69	92.74	92.63	92.66
Bi-LSTM	96.75	96.79	96.76	96.75
BERT	98.30	98.51	98.76	98.63
Model of this paper	99.75	99.82	99.83	99.83

As shown in Table 3, the RF model performs best among machine learning models, with accuracy and F1-score both at 91.18%. The Bi-LSTM model performs best among the traditional deep learning models, with accuracy and F1-score both at 96.75%. The BERT model, even without the mix-training optimization, outperforms all other machine learning and deep learning baselines across all four evaluation indicators. Compared to the best machine learning method (RF), the BERT-based approach improves accuracy and F1-score by over 7%. Compared to traditional deep learning methods, BERT shows varying degrees of

improvement in all indicators. For example, the accuracy of BERT model achieves 98.30%, which is 5.6% higher than the CNN model and 1.55% higher than the Bi-LSTM. After implementing the mix-training strategy, all 4 evaluation indicators show further improvement compared to standard BERT fine-tuning, with each indicator exceeding 99%. For the proposed model, the accuracy, precision, recall and F1-score are 1.45%, 1.31%, 1.07% and 1.2% higher than the standard BERT model, respectively. Compared to the Bi-LSTM model, the proposed model shows improvements of 2%, 3.03%, 3.07% and 3.08% in accuracy, precision, recall and F1-score, respectively. This indicates that the proposed model possesses excellent analytical performance. The mix-training method exposes the model to different data subsets during successive training phases, enabling it to adequately learn variations between samples and internal relationships within the data, thereby enhancing its analytical capability.

To illustrate the advantages of the proposed model, consider a sample event of the "Single-phase fault (unsuccessful reclosure)". This event sample includes multiple monitoring alarm signals related to protection activation, circuit breaker operations, recloser actions, and auxiliary signals such as "spring does not store energy". BERT's multi-head attention mechanism allows it to weigh the importance of each token within the full sequence context, irrespective of position. For example, in this sample, signals like "protection action", "reclosing action", "switch trip/close", and "spring does not store energy" appear repeatedly and interact in non-adjacent positions. BERT can effectively associate the initial "switch trip" with the later "reclosing action failure" and "protection reactivation", even when they are separated by other signals. This capability is critical for distinguishing permanent faults (where reclosing fails and the fault recurs) from temporary faults (where reclosing succeeds). Furthermore, BERT's ability to understand the semantic roles of alarm signals, such as distinguishing between "action" and "return" signals, enables it to filter out interference signals, "spring does not store energy" for example, and focus more effectively on the core identification logic. Unlike Bi-LSTM and CNN models, which often require manual design of sliding windows or time-step processing, BERT inherently models the entire sequence in parallel, leading to more efficient and accurate fault identification. In this case, BERT leverages its multi-head attention mechanism to capture global contextual relationships across the entire sequence simultaneously, whereas traditional deep learning methods such as Bi-LSTM and CNN may struggle with long-range dependencies due to window size limitations or sequential processing constraints.

To verify the impact of sample imbalance on model performance, we calculated precision, recall, and F1-score for each PGMAE type using the proposed model, the standard BERT model and the Bi-LSTM model, as shown in Tables 4–6. The corresponding confusion matrices are visualized in Fig. 4.

Table 4: Identification results of each event type on proposed model

Event number	Precision/%	Recall/%	F1-Score/%
1	97.27	97.95	97.61
2	99.54	98.69	99.12
3	100.00	99.75	99.88
4	99.41	99.68	99.55
5	98.33	97.86	98.10
6	99.06	99.66	99.36
7	99.07	100.00	99.53
8	99.84	98.10	98.86
9	99.10	97.18	98.13

Table 5: Identification results of each event type on BERT model

Event number	Precision/%	Recall/%	F1-Score/%
1	96.98	96.94	96.96
2	99.12	99.43	99.27
3	99.72	99.92	99.82
4	98.89	99.07	98.98
5	98.03	98.47	98.25
6	98.97	99.11	99.04
7	97.66	97.72	97.69
8	98.79	98.83	98.81
9	98.43	99.35	98.89

Table 6: Identification results of each event type on Bi-LSTM model

Event number	Precision/%	Recall/%	F1-Score/%
1	96.43	96.37	96.40
2	97.14	97.06	97.10
3	97.05	97.19	97.12
4	96.97	96.78	96.87
5	96.52	96.37	96.44
6	96.14	96.57	96.35
7	96.81	96.57	96.69
8	97.69	97.63	97.66
9	96.36	96.10	96.23

From Tables 4–6, the BERT model achieves higher precision, recall and F1-score across all event types compared to the Bi-LSTM model, but its performance is still lower than that of the proposed model for every event type. Although the proposed model achieves strong overall results, its identification performance varies across specific PGMAE types. Certain PGMAE types, such as bus fault (event 1) and transformer electrical fault (event 5), exhibit slightly poorer prediction outcomes, likely due to their smaller original sample sizes. For instance, the precision for bus fault identification is only 97.27%, slightly lower than for other types. The recall rate and F1-score for this event type are also relatively lower. For minority class types (event number 1, 5, 6, 7, 9), the average precision, recall and F1-score are 98.57%, 98.53%, and 98.03%, respectively. In comparison, for majority class types (event number 2, 3, 4, 8), the corresponding averages are 99.70%, 99.06%, and 99.35%, respectively. Consequently, when processing alarm samples of specific minority event types, the model remains more susceptible to misjudgments and missed detections, which could potentially impact grid operation stability.

	1	2	3	4	5	6	7	8	9
1	36	0	0	0	0	1	0	0	0
2	0	50	0	0	0	0	1	0	0
3	1	0	63	0	0	0	0	0	0
4	0	0	0	558	1	1	0	0	0
5	0	1	1	0	370	0	0	0	1
6	0	0	0	0	0	133	0	0	1
7	0	0	0	0	0	0	59	1	0
8	0	0	0	1	0	0	0	38	0
9	0	0	0	0	1	0	0	1	251

(a)

	1	2	3	4	5	6	7	8	9
1	37	0	0	0	0	0	0	0	0
2	0	50	0	0	0	1	0	0	0
3	0	0	63	0	0	0	0	1	0
4	0	0	0	558	1	0	0	0	1
5	0	0	1	0	371	0	0	0	1
6	0	0	0	0	2	132	0	0	0
7	0	1	0	0	0	0	59	0	0
8	1	0	0	0	0	0	0	38	0
9	0	0	0	0	0	0	1	0	252

(b)

Figure 4: Normalized confusion matrix: (**a**) BERT; (**b**) Proposed model

To further evaluate the model's performance across different decision thresholds and under class imbalance, we present the precision-recall curves for all 9 event types in Fig. 5.

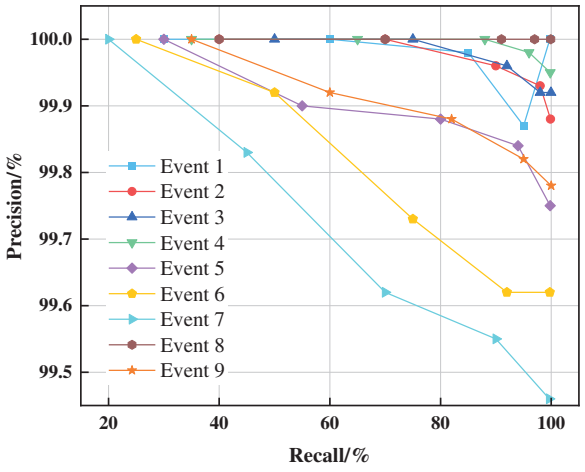

Figure 5: Precision-Recall Curves for each class

As observed, the PR curves for majority classes such as event 3 (single-phase fault, unsuccessful reclosure) and event 2 (single-phase fault, successful reclosure) maintain high precision across all recall levels, with their average precision close to 1.0, indicating near-perfect performance. In contrast, the curves for some minority classes reveal the model's remaining challenges. Specifically, Event 1 (bus fault) and event 7 (transformer on-load voltage regulation fault) show more pronounced curvature. While they can achieve high precision at lower recall levels, their precision drops more significantly as the model attempts to capture more positive samples (higher recall). This trade-off is characteristic of classes where the model's discriminatory power is less certain.

The area under the precision-recall curve, or average precision, for these minority classes, though still high (e.g., >0.98), is marginally lower than that of the majority classes. This quantifies the slight performance gap that persists even after our data augmentation efforts.

This PR analysis corroborates the findings from the confusion matrix shown in Fig. 4, confirming that while the model is highly effective overall, its performance on rare but critical fault types like bus and specific transformer faults is relatively more sensitive to the identification threshold. This insight is crucial

for operational settings where high recall for certain minority faults might be prioritized, potentially at the cost of a slight decrease in precision.

To evaluate the reliability of the model's predicted probabilities, we plotted calibration curves for all categories, as shown in Fig. 6. The dashed line represents perfect calibration. At the same time, we calculated the expected calibration error (*ECE*), which is defined in Eq. (14).

$$ECE = \sum_{b=1}^{B} \frac{n_b}{N} |acc\,(b) - conf\,(b)| \tag{14}$$

where B is the number of intervals; n_b is the number of samples in the b-th interval, N is the total number of samples; $acc(b)$ and $conf(b)$ are the accuracy and average confidence within the b-th interval, respectively. Lower *ECE* values indicate better calibration.

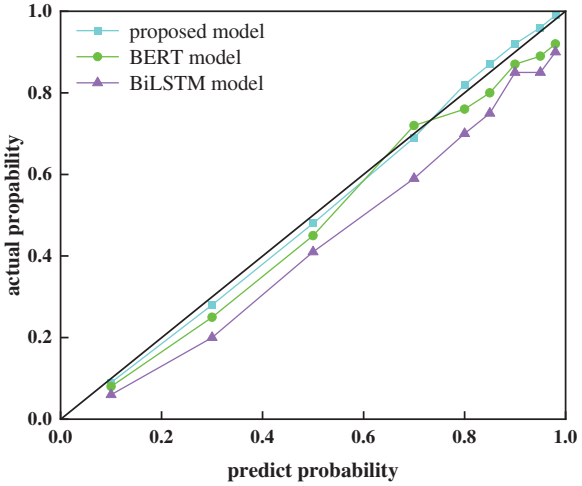

Figure 6: Calibration plots for the proposed model and the baseline BERT model

As shown in Fig. 6, the proposed model's calibration curve lies very close to the ideal diagonal, with an ECE of 0.008, demonstrating excellent calibration characteristics. In contrast, the standard BERT model without mix-training optimization shows slight overconfidence, especially in the high-confidence range (>0.9), where its predicted probabilities exceed the actual accuracy, with an ECE of 0.015. We further analyzed the calibration of minority classes. Taking "bus fault (event 1)" as an example, after introducing SimBERT data augmentation, the model's discrimination for this class improved, and its calibration curve was slightly exceeded the diagonal, indicating that the model's actual performance was even better than its predicted confidence level (see Table 7, with precision reaches 100%). This suggests that our data augmentation and mix-training strategies not only improved the model's identification performance but also significantly enhanced the reliability of its probability outputs, which is crucial for confidence-based automated decision-making in power monitoring systems.

To determine the impact of different target sample sizes on model performance, we tested target sample sizes ranging from 1000 to 3000 in increments of 500. For any minority class with fewer samples than the target size, we used SimBERT for augmentation until its count reached the target. After five rounds of SimBERT augmentation with different targets, we evaluated the performance of the BERT model, trained

with our optimized approach, on the same test set. Accuracy and F1-score were selected as evaluation indicators. The trends of these two indicators are shown in Fig. 7.

Table 7: Identification results of each event type of proposed model augmented by SimBERT

Event number	Precision/%	Recall/%	F1-Score/%
1	100.00	99.74	99.87
2	99.88	99.86	99.87
3	99.92	99.95	99.93
4	99.95	99.83	99.89
5	99.75	99.80	99.77
6	99.62	99.75	99.68
7	99.46	99.65	99.55
8	100.00	99.92	99.96
9	99.78	100.00	99.89

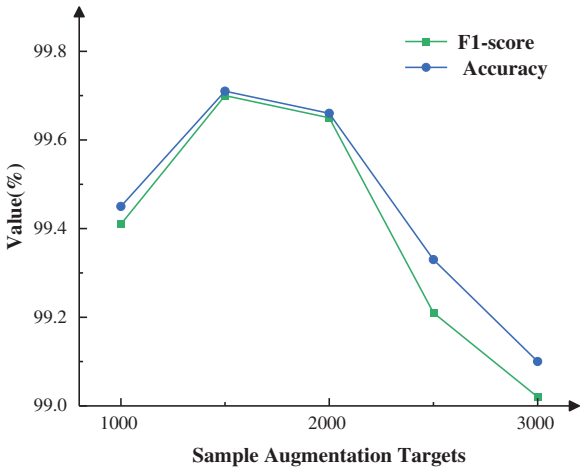

Figure 7: Impact of different sample augmentation targets on model identification results

Fig. 7 illustrates that the sample augmentation target size influences model performance. When the target number of generated target samples reaches 1500, the proposed model achieves its optimal performance, with the highest accuracy and F1-score among the five augmentation scenarios tested. When the target size is excessively high (e.g., 2000 or 3000), the SimBERT model may introduce substantial noise in the generated samples, thereby interfering with the model training process and leading to a performance decline. Consequently, with the sample augmentation target set at 1500, the accuracy and F1-score for minority class samples based on proposed model and the standard BERT model are compared in Fig. 8, and the proposed model's detailed prediction results for each PGMAE type are shown in Table 7.

From Fig. 8, after augmentation, the precision and F1-score of all minority classes show varying degrees of improvement. The bus fault event (event 1) shows the largest improvement, with precision and F1-score increasing by 2.73% and 2.26%, respectively. As shown in Table 7, after SimBERT sample augmentation, all event types exhibit improvements in precision and recall, leading to corresponding increases in F1-score compared to the non-augmented baseline (Table 4). Notably, for the bus fault event (event 1), precision increases from 97.27% (without augmentation) to 100% (with augmentation), while the recall rate rises from

97.95% (without augmentation) to 99.74% (with augmentation). This augmentation effectively alleviates the model's previously low identification rate for minority class samples and reduces the misjudgment probability for these event types within the proposed identification model.

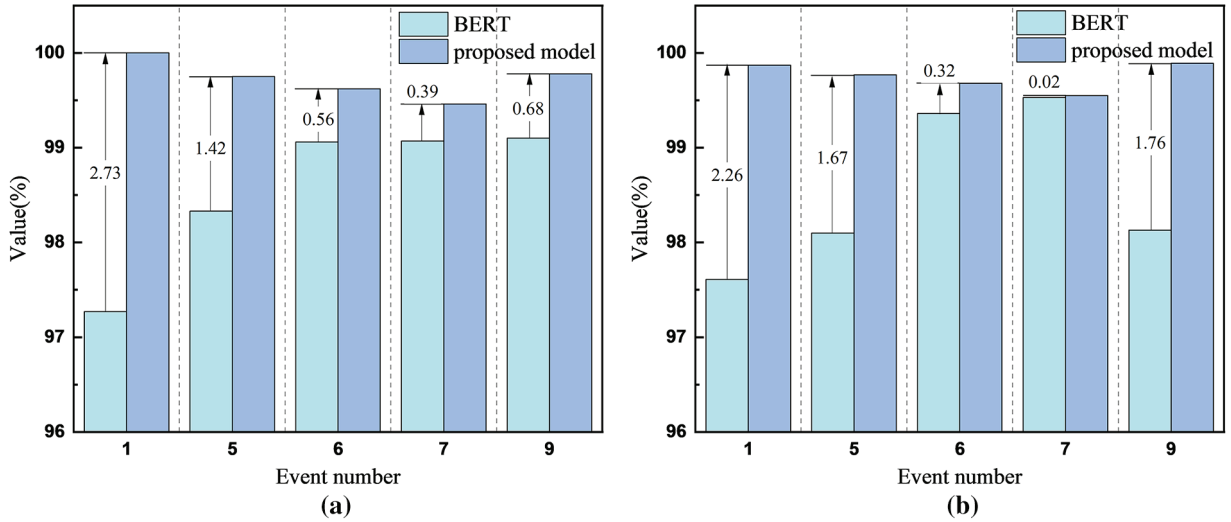

Figure 8: Identification results comparison on minority class samples: (**a**) Precision; (**b**) F1-score

To address concerns regarding data leakage and overfitting, we conducted a quantitative analysis of the similarity between original and SimBERT-generated samples and verified the model's performance on the untouched test set. We evaluated the diversity of the augmented dataset using two metrics: (1) The Jaccard similarity coefficient at the character level between each generated sample and its source original sample, which measures the proportion of shared characters. (2) The cosine similarity between the embedding vectors of original and generated samples, computed using the pre-trained SimBERT model itself as per Eq. (7). The distribution of these similarity scores across all generated samples is summarized in Table 8. The results indicate that while the generated samples are semantically relevant (high cosine similarity), they exhibit significant lexical variation (low Jaccard similarity), confirming that SimBERT produces paraphrases rather than mere replicas.

Table 8: Similarity statistics between original and augmented samples

Similarity metric	Mean	Standard deviation	Maximum
Jaccard Similarity	0.32	0.11	0.78
Cosine Similarity	0.89	0.05	0.94

The ultimate proof against overfitting and data leakage is the model's performance on the untouched test set, which contains only original, non-augmented samples. The high accuracy (99.75%) and F1-score (99.83%) reported in Table 3 for the proposed model were achieved on this original test set. This performance, significantly better than the standard BERT model, demonstrates that the gains originate from learning robust, generalized features from the augmented training set, rather than from memorizing leaked or near-duplicate data.

To evaluate the robustness of our feature extraction method, we conducted sensitivity tests on two key parameters: the co-occurrence coefficient threshold and the tokenization strategy. We tested three thresholds

for $M(\delta_i, \delta_j)$: 0, 0.1, and 0.2. The results are shown in Table 9. Table 9 shows that a threshold of 0 yields the best performance in terms of F1-score, confirming our initial empirical choice.

Table 9: Sensitivity of co-occurrence threshold on model performance

Threshold	Precision (%)	Recall (%)	F1-Score (%)
0	99.82	99.83	99.83
0.1	99.75	99.76	99.75
0.2	99.68	99.70	99.69

We compared character-level tokenization with word-level tokenization using the Jieba toolkit with a custom power grid dictionary, shown in Table 10. As shown in Table 10, character-level tokenization outperforms word-level tokenization in this domain, likely due to its ability to handle out-of-vocabulary terms and the compositional nature of Chinese alarm messages effectively.

Table 10: Performance comparison of tokenization strategies

Tokenization method	Precision (%)	Recall (%)	F1-Score (%)
Character-level	99.82	99.83	99.83
Word-level (Jieba)	98.45	98.50	98.47

4.4 Ablation Study and Component Analysis

To quantitatively evaluate the contribution of each key component in our proposed framework, we conducted a comprehensive ablation study. The results are summarized in Table 11.

Table 11: Results of the ablation study on the test set

Model variant	Accuracy/%	F1-Score/%	Δ F1-Score
Proposed full model	99.75	99.83	–
- w/o SimBERT Augmentation	97.51	97.60	−2.23
- w/o Mix-Training	98.30	98.63	−1.20
- w/o IE-Based Filtering	98.92	98.95	−0.88
- Replace BERT with Bi-LSTM	96.75	96.75	−3.08
- Replace BERT with RoBERTa	99.40	99.45	−0.38

From Table 11, removing the SimBERT-based augmentation leads to the most significant performance drop (F1-score decreases by 2.23%). This underscores the critical role of addressing class imbalance for robust identification. The impact is particularly pronounced for minority classes like “transformer on-load voltage regulation fault” (F1-score drops from 99.53% to 94.8% without augmentation). Removing mix-training results in a noticeable performance decrease (F1-score drops by 1.20%). This validates that our proposed optimization strategy effectively enhances model generalization by reducing overfitting to a single data split. The IE-based signal filtering contributes 0.88% to the overall F1-score, demonstrating its

importance in removing noisy, redundant alarm signals that could confuse the model. Replacing BERT with Bi-LSTM causes the largest performance degradation, highlighting the superior semantic encoding capability of the transformer-based architecture. The RoBERTa variant performs competitively but still falls short of our BERT_base_chinese implementation, possibly due to better domain alignment of BERT's pre-training corpus.

We also evaluated the training and inference efficiency of our approach. On our experimental setup, a single fine-tuning epoch for the BERT model takes approximately 9.75 min. The complete mix-training process (3 rounds × 3 epochs) requires about 29.25 min. For inference, the model processes approximately 120 samples per second on the NVIDIA RTX A6000 GPU, meeting real-time operational requirements for power grid monitoring centers.

5 Conclusion

This paper proposes a PGMAE identification method based on the BERT large language model to address the problem of high sample imbalance and low prediction accuracy in traditional methods.

(1) Based on IE and k-means++ clustering, all alarm signals are separated into distinct event types to form labeled PGMAE samples.
(2) The SimBERT model is used to generate augmented samples for minority classes. Generated samples with high semantic similarity to the originals are selected and added to the original training set.
(3) The augmented PGMAE samples are used to fine-tune the BERT model. The training process is optimized using a mix-training strategy to improve the model's identification accuracy.

This paper constructs an identification framework based on LLMs to achieve accurate PGMAE identification, which is beneficial for timely identifying faults in power systems and ensuring stable grid operation. Future research direction will focus on two core tasks. First, we plan to integrate this identification model with existing large-scale power grid disposal systems to construct an end-to-end automated platform for event identification and intelligent disposal. Second, and equally important, is the development of a robust operational framework to address real-world challenges. This framework will include: a streaming inference mechanism to handle alarm bursts within defined time windows; an out-of-distribution detection module, based on predictive confidence thresholds, to flag unseen event types for human review; and a continuous monitoring plan coupled with a periodic retuning strategy to mitigate concept drift, thereby ensuring the model's long-term reliability and adaptability in a dynamic grid environment.

Acknowledgement: Not applicable.

Funding Statement: This research was funded by Science and Technology Project of State Grid Jiangsu Electric Power Co. Ltd. (No. J2024172).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Qiang Xu and Guoqiang Sun; methodology, Leyao Cong; software, Leyao Cong, Jianing Wang and Xueheng Shi; validation, Qiang Xu, Leyao Cong and Xingyu Zhu; formal analysis, Shaojun Cui; investigation, Xingyu Zhu and Shaojun Cui; resources, Qiang Xu; data curation, Leyao Cong; writing—original draft preparation, Jianing Wang and Xueheng Shi; writing—review and editing, Qiang Xu and Guoqiang Sun; visualization, Qiang Xu; supervision, Guoqiang Sun; project administration, Qiang Xu; funding acquisition, Guoqiang Sun. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Oh JY, Yoon YT, Sohn JM. A comprehensive review of alarm processing in power systems: addressing overreliance on fault analysis and projecting future directions. Energies. 2024;17(13):3344. doi:10.3390/en17133344.
2. Simonson RJ, Keebler JR, Blickensderfer EL, Besuijen R. Impact of alarm management and automation on abnormal operations: a human-in-the-loop simulation study. Appl Ergon. 2022;100(1):103670. doi:10.1016/j.apergo.2021.103670.
3. Taghipourbazargani N, Dasarathy G, Sankar L, Kosut O. A machine learning framework for event identification via modal analysis of PMU data. IEEE Trans Power Syst. 2023;38(5):4165–76. doi:10.1109/TPWRS.2022.3212323.
4. Liu D, Sun K. Random forest solar power forecast based on classification optimization. Energy. 2019;187:115940. doi:10.1016/j.energy.2019.115940.
5. Zhang L, Ji W, Zhao Y, Huang D, Qian B. Review on application and challenges of large language models in power grids. IEEE Access. 2025;13:106932–41. doi:10.1109/ACCESS.2025.3580439.
6. Ginson Y, Edwin EB, Ebenezer V, Kirubakaran SS, Roshni Thanka M, Raja MM. Neural sequence modelling for spam classification via bidirectional LSTM and hierarchical neural networks: a deep learning approach. In: Proceedings of the 2025 Fourth International Conference on Smart Technologies, Communication and Robotics (STCR); 2015 May 9–10; Sathyamangalam, India. p. 1–6. doi:10.1109/STCR62650.2025.11019808.
7. Mariyam A, Althaf Hussain Basha SK, Viswanadha RS. Long document classification using hierarchical attention networks. Int J Intell Syst Appl Eng. 2023;11:2708. doi:10.18293/IJISAE.2023.2708.
8. Bai Z, Sun G, Zang H, Zhang M, Shen P, Liu Y, et al. Identification technology of grid monitoring alarm event based on natural language processing and deep learning in China. Energies. 2019;12(17):3258. doi:10.3390/en12173258.
9. Su Y, Shi L, Zhou K, Bai G, Wang Z. Knowledge-informed deep networks for robust fault diagnosis of rolling bearings. Reliab Eng Syst Saf. 2024;244:109863. doi:10.1016/j.ress.2023.109863.
10. Fasogbon SK, Shaibu SA. Energy grid optimization using deep machine learning: a review of challenges and opportunities. Preprints. 2023. doi:10.20944/preprints202306.1874.v1.
11. Yang Y, Hu Q, Liu Y, Pan X, Gao S, Hao B. Power grid monitoring event recognition method integrating knowledge graph and deep learning. Front Energy Res. 2022;10:950954. doi:10.3389/fenrg.2022.950954.
12. Henning S, Beluch W, Fraser A, Friedrich A. A survey of methods for addressing class imbalance in deep-learning based natural language processing. In: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistic; 2023 May 2–6; Dubrovnik, Croatia. p. 523–40. doi:10.18653/v1/2023.eacl-main.38.
13. Karimi A, Rossi L, Prati A. AEDA: an easier data augmentation technique for text classification. In: Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021; 2021 Nov 7–11; Punta Cana, Dominican Republic. p. 2748–54. doi:10.18653/v1/2021.findings-emnlp.234.
14. Sun X, Wu D, Boulet B. Analyzing the benefits of data augmentation for smart grid anomaly detection and forecasting. In: Proceedings of the 2023 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE); 2023 Sep 24–27; Regina, SK, Canada. p. 416–22. doi:10.1109/CCECE58730.2023.10288888.
15. Xu T, Zhang J, Meng H, Liu L, Wang K, Qiao J, et al. SeqESR-GAN-based sparse data augmentation for distribution networks. IEEE Trans Ind Inform. 2024;20(11):12913–23. doi:10.1109/TII.2024.3431009.
16. Bae J, Lee C. Easy data augmentation for improved malware detection: a comparative study. In: Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp); 2021 Jan 17–20; Jeju Island, Republic of Korea. p. 214–8. doi:10.1109/bigcomp51126.2021.00048.
17. Lan J, Zhou Y, Guo Q, Sun H. Data augmentation for data-driven methods in power system operation: a novel framework using improved GAN and transfer learning. IEEE Trans Power Syst. 2024;39(5):6399–411. doi:10.1109/TPWRS.2024.3364166.

18. Annepaka Y, Pakray P. Large language models: a survey of their development, capabilities, and applications. Knowl Inf Syst. 2025;67(3):2967–3022. doi:10.1007/s10115-024-02310-4.
19. Minaee S, Mikolov T, Nikzad N, Chenaghlu M, Socher R, Amatriain X, et al. Large language models: a survey. arXiv:2402.06196. 2024.
20. Devlin J, Chang MW, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT); 2019 Jun 2–7; Minneapolis, MN, USA. p. 4171–86.
21. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS); 2017 Dec 4–9; Long Beach, CA, USA. p. 20–6.
22. Jia J, Yang Q, Fu H, Yang JG, He YD. Research on pre-training language model construction and text semantic analysis based on power equipment big data. Proc CSEE. 2023;43(3):1027–37. (In Chinese). doi:10.13334/j.0258-8013.pcsee.212629.
23. Qin N. Research on the classification method of CTC alignment joint-test problems based on RoBERTa-wwm and deep learning integration. IEEE Access. 2023;11:139395–408. doi:10.1109/ACCESS.2023.3340527.
24. Wei Z, Wang C, Yang X, Zhao W. Imbalanced sentiment classification of online reviews based on SimBERT. J Intell Fuzzy Syst. 2023;45(5):8015–25. doi:10.3233/jifs-230278.
25. Arthur D, Vassilvitskii S. k-means++: the advantages of careful seeding. In: Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms; 2007 Jan 7–9; New Orleans, LA, USA. p. 1027–35.
26. Jing LP, Huang HK, Shi HB. Improved feature selection approach TFIDF in text mining. In: Proceedings of International Conference on Machine Learning and Cybernetics; 2002 Nov 4–5; Beijing, China. p. 944–6. doi:10.1109/ICMLC.2002.1174522.
27. Dong L, Yang N, Wang W, Wei F, Liu X, Wang Y, et al. Unified language model pre-training for natural language understanding and generation. In: Proceedings of the 33rd Annual Conference on Neural Information Processing Systems (NIPS); 2019 Dec 8–14; Vancouver, BC, Canada. p. 13042–54.
28. Aroca-Ouellette S, Rudzicz F. On losses for modern language models. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2020 Nov 16–20; Online. p. 4970–81. doi:10.18653/v1/2020.emnlp-main.403.
29. Haviv A, Ram O, Press O, Izsak P, Levy O. Transformer language models without positional encodings still learn positional information. In: Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2022; 2022 Dec 7–11; Abu Dhabi, United Arab Emirates. p. 1382–90. doi:10.18653/v1/2022.findings-emnlp.99.
30. Wu H, Shang W, He C, Li B, Song K. Design of power equipment fault detection and diagnosis system based on deep learning. In: Proceedings of the 2024 International Conference on Electrical Drives, Power Electronics & Engineering (EDPEE); 2024 Mar 12–14; Athens, Greece. p. 911–6. doi:10.1109/EDPEE61724.2024.00175.