



## OPEN ACCESS

## EDITED BY

Jun Liu,  
Xi'an Jiaotong University, China

## REVIEWED BY

Gangjun Gong,  
North China Electric Power University, China  
Zhongbao Wei,  
Beijing Institute of Technology, China

## \*CORRESPONDENCE

Zhao Ruifeng,  
✉ ruifzhao@126.com

RECEIVED 03 April 2024

ACCEPTED 10 June 2024

PUBLISHED 31 July 2024

## CITATION

Jiangang L, Ruifeng Z, Zhiwen Y, Yue D, Jiawei S  
and Ting Y (2024), Text classification for  
distribution substation inspection based on  
BERT-TextRCNN model.  
*Front. Energy Res.* 12:1411654.  
doi: 10.3389/fenrg.2024.1411654

## COPYRIGHT

© 2024 Jiangang, Ruifeng, Zhiwen, Yue, Jiawei  
and Ting. This is an open-access article  
distributed under the terms of the [Creative  
Commons Attribution License \(CC BY\)](#). The use,  
distribution or reproduction in other forums is  
permitted, provided the original author(s) and  
the copyright owner(s) are credited and that the  
original publication in this journal is cited, in  
accordance with accepted academic practice.  
No use, distribution or reproduction is  
permitted which does not comply with these  
terms.

# Text classification for distribution substation inspection based on BERT-TextRCNN model

Lu Jiangang<sup>1</sup>, Zhao Ruifeng<sup>1\*</sup>, Yu Zhiwen<sup>1</sup>, Dai Yue<sup>1</sup>, Shu Jiawei<sup>2</sup>  
and Yang Ting<sup>2</sup>

<sup>1</sup>Guangdong Power Grid Limited Liability Company, Guangdong, China, <sup>2</sup>School of Electrical Automation and Information Engineering, Tianjin University, Tianjin, China

With the advancement of source-load interaction in the new power systems, data-driven approaches have provided a foundational support for aggregating and interacting between sources and loads. However, with the widespread integration of distributed energy resources, fine-grained perception of intelligent sensing devices, and the inherent stochasticity of source-load dynamics, a massive amount of raw data is being recorded and accumulated in the data center. Valuable information is often dispersed across different paragraphs of the raw data, making it challenging to extract effectively. Distribution substation inspection plays a crucial role in ensuring the safe operation of the power system. Traditional methods for inspection report text classification typically rely on manual judgment and accumulated experience, resulting in low efficiency and a significant misjudgment rate. Therefore, this paper proposes a text classification method for inspection reports based on the pre-trained BERT-TextRCNN model. By utilizing the dense connection between the BERT embedding layer and the neural network, the proposed method improves the accuracy of matching long texts. This article collected 2,831 maintenance data for the first quarter of 2023 from the distribution room, including approximately 58 environmental testing data, 738 environmental box testing data, approximately 672 distribution room testing data, and approximately 1,363 box type substation testing data. A text corpus was constructed for experiments. Experimental results demonstrate that the proposed model automatically classifies a large volume of manually recorded inspection report data based on time, location, and faults, achieving a classification accuracy of 94.7%, precision of 92%, recall of 92%, and F1 score of 90.3%.

## KEYWORDS

source-load interaction, BERT-TextRCNN, text classification, nlp, information extraction

## 1 Introduction

With the continuous development of the energy internet, the integration of distributed power sources, distributed energy storage, and the widespread adoption of new types of loads have led to the emergence of new characteristics in the system, such as diversified power supply, interactive power consumption, and digitized management (Zeng et al., 2020). At the same time, the perception capability of various stages in the source-load interaction scenario has gradually improved, and the data acquisition system collects data generated during dynamic adjustment

processes, including source-load interactions (Cai et al., 2021). The processing of massive heterogeneous data becomes crucial after the integration of multiple energy sources.

However, the randomness and decentralization of source-load interaction result in different sources and compositions of data, leading to the multi-source heterogeneity of grid data and posing significant challenges to the application of big data and artificial intelligence technologies in the “source-load interaction” mode. For example, the complementary characteristics of different distributed renewable energy generation, coordinated planning of capacity allocation for various renewable energy generation devices within distribution areas, overall output fluctuations, and the controllable characteristics of flexible loads generate multiple sources of data. Due to the unstructured nature of textual record data, which is diverse and abundant with high redundancy, it is difficult for humans to quickly locate and retrieve useful information, thus limiting the true value of the data. Therefore, it is necessary to identify and classify heterogeneous textual data.

In order to improve the efficiency of document processing by analysts, researchers have developed various text classification-based research methods (Hong et al., 2019; Liao et al., 2020). Early studies focused on the development of machine learning-based methods. Goh and Ubeynarayana (2017) evaluated six machine learning methods for classifying accident reports, including linear regression, random forest, support vector machine, decision tree, and naive Bayes methods. The results showed that machine learning methods produced superior classification results. Machine learning methods require manual extraction of text features, which is inefficient and weakens the model’s generalization ability.

With the development of artificial intelligence, deep learning methods have shown considerable potential (Xu and Tan, 2019; Fang et al., 2020) and have become increasingly mature in computer vision and natural language processing. For example, Chang et al. (2020) built a multimodal sentiment classification model with a composite hierarchical fusion using convolutional networks (TCN) and soft attention mechanisms. They filtered and reduced the multimodal feature information through selective attention mechanisms to extract sentiment features from the text, resulting in classification results.

So far, BERT is one of the most accurate language models for text classification tasks (Choi et al., 2021) and has demonstrated excellent performance in various natural language processing tasks. However, due to the specific language style and structure of inspection case texts, the BERT model pretraining corpora, such as BooksCorpus and English Wikipedia, have different data distributions from the data in power inspection texts (Coster and Kauchak, 2011). The latter contains denser professional vocabulary, necessitating further training of BERT. Therefore, this paper employs transfer learning to incorporate language features from accident case texts into the model and designs the TextRCNN bidirectional recurrent structure to fully learn the contextual information of power data, enabling better classification of accident case texts (Liao et al., 2021).

In summary, this paper focuses on the automatic classification technology for unstructured inspection text data in operation at distribution substation in the source-load interaction scenario. To address the massive data in the source-load interaction scenario, a BERT-TextRCNN model based on large-scale pretraining language models is proposed. It utilizes the bidirectional recursive structure of

TextRCNN to capture contextual information to the maximum extent and establishes a text classification model through pretraining and fine-tuning. This model organizes and reduces redundant information in documents by classifying key information, achieving automatic text classification. By utilizing artificial intelligence algorithms for automatic identification, it improves the quality of grid data and provides data support for research on maintenance and scheduling of distribution systems under the source-load interaction mode.

The main contributions of this paper are as follows.

- (1) Considering that traditional models like Word2Vec require a large amount of raw data for training word vectors, the pretrained language model BERT is applied to the classification task of inspection records using the pretraining and fine-tuning approach, enabling feature extraction from small-sample data.
- (2) TextRCNN is applied to the classification task of power station maintenance texts, encoding the collected text records and constructing a labeled case dataset. It extracts the required key information from the power text data, providing data support for the exploration of the value of heterogeneous data in the source-load interaction scenario and related research.
- (3) Experimental scenarios are set based on actual inspection data from distribution substations. The proposed model is compared with different algorithms to validate its effectiveness in automatic text classification.

## 2 Related work

### 2.1 Characteristics of distribution substation inspection text data

#### 2.1.1 Data acquisition

A massive amount of raw data is recorded and accumulated in the central platform, leading to a significant increase in data processing pressure. In traditional distribution systems, data management relies on manual input by inspection personnel and manual information extraction by distribution network operation and maintenance staff, resulting in inefficiency and the consumption of a large amount of labor resources. By employing artificial intelligence techniques to intelligently identify and classify heterogeneous information in the data platform, it becomes possible to rapidly respond to relevant information regarding the distribution, power supply capacity, and operation and maintenance status of power supply points on the distribution network side. This optimization enhances the interactive operation of the “source-load interaction” system. Figure 1 shows the structure of the classification model based on the BERT pre-training model.

#### 2.1.2 Power station inspection text data type

Inspection record data are complex and diverse, with a large number of multi-source heterogeneous and heterogeneous data, including regular character sets, natural language text and other kinds of data of different time scales. According to the data structure can be divided into structured data, semi-structured data,

unstructured data, *etc.* This subsection mainly analyzes the unstructured text data of each power station maintenance. Such texts are classified into three categories, including time, station type, and faults. Using recurrent neural network (RNN) for text modeling and feature extraction (Williams and Zipser, 1989), but there is a problem of gradient vanishing, which cannot effectively handle long-distance text information. To solve this problem, researchers proposed a long short term memory (LSTM) network based on RNN (Hochreiter and Schmidhuber, 1997), which better extracts long sequence features by retaining important text information and forgetting unimportant information. Due to the fact that LSTM can only model the sequence of text in one direction and cannot utilize input character information, a bidirectional long short term memory (Bi LSTM) network is introduced to extract bidirectional features from text (Du et al., 2018). Reference (Devlin et al., 2019) proposed a Transformer based bidirectional encoder representations from transformers (BERT) model, which achieves bidirectional prediction by masking the input while maintaining a deep network hierarchy.

## 2.2 Model introduction

### 2.2.1 BERT model

The BERT model is a large-scale pre-trained language model based on Transformer (Jorge and Balazs, 2019; Meng et al., 2022) that can be pre-trained independently on each large dataset and then fine-tuned in a specific domain as well as in a specific task to make it suitable for the final target task. For the distribution substation text data, Bert model is used to recognize and process the Chinese text.

BERT is used for text classification tasks in two main steps.

- 1) Train the model on a large unlabeled corpus to master the expressive power of the language and to be able to support subsequent text classification tasks.
- 2) Fine-tune the model for the text classification task and perform supervised training on a labeled dataset.

The BERT model consists of a stack of multiple Transformer encoders and contains two main sub-layers: a multi-headed attention layer and a fully connected feed-forward neural network layer. These two sub-layers are connected by residuals and then layer-normalized to finally output a 768-dimensional vector. The output of each sublayer is the input vector of each layer.

### 2.2.2 TextRCNN layer

Deep learning techniques, specifically utilizing RNN and CNN, have been employed for text classification, each with its own strengths and limitations. RNN excels in handling sequential structures and capturing the contextual information of sentences. However, RNN is a biased model that assigns higher importance and larger weights to words appearing later in the sequence. In the context of inspection text records, where important words can occur at any position, this approach may lead to a decrease in the overall classification accuracy. Moreover, the large number of parameters in RNN results in lower computational efficiency and limited parallelization effectiveness. On the other hand, CNN is an unbiased model that converts text information into matrix values as input. It incorporates convolutional layers, pooling

layers, and fully connected layers to extract the most salient features. The advantages of CNN lie in its simplicity and fast training speed. However, determining the appropriate size of the sliding window in CNN poses challenges. Selecting a window size that is too small may result in the loss of important information, while choosing a size that is too large leads to a significantly larger parameter space, hindering the discovery of long-distance relationships, semantic transitions, and intensity distributions in the text.

To overcome the limitations of TextCNN and TextRNN while leveraging the advantages of both models, this study proposes the utilization of a bidirectional recurrent network (BiLSTM) to capture contextual semantic features. Additionally, a convolutional layer is employed for feature extraction, leading to the construction of the TextRCNN model for text classification. The organizational framework for text classification is depicted in Figure 2.

## 3 BERT-TextRCNN model

### 3.1 Dataset processing

The collected distribution room inspection text data contains a large number of useless punctuation marks, stop words, and special characters, *etc.* In many algorithms, especially those based on statistics, noise and unnecessary features can adversely affect the system performance. Therefore, the collected text data needs to be cleaned by three operations.

- (1) Removal of deactivated words. The text includes many words that are not important for the classification task, such as {"a," "about," "across," "after," "...," "again"}. The most common way to deal with these words is to remove them from the text and the document.
- (2) Word lowercase processing. Various capitalized words or letters often appear in the inspection text data, such as current *I/i*, voltage *U/u*, *etc.* In the process of training a classification model, the case form of the words may affect the training effect of the model.
- (3) Word stem extraction. In text data, a device can appear in different forms (e.g., in the form of a commutator/converter, *etc.*), and the semantics of each form is actually the same. Stem extraction is a method to merge different forms of a word into the same feature space.

### 3.2 Text feature extraction

Text data are unstructured forms of data and cannot be directly used as input forms for the model. Before inputting text data into a classification model, these unstructured text sequences must be converted into structured feature vectors. In this paper, we use Bag of Words (BOW) method to convert text into vector form for input into the machine algorithm classification model. The BOW method constructs a vocabulary table by calculating the frequency of each occurring word in the training set to obtain the feature matrix of the corresponding text (Jing et al., 2022), as follows.

Taking a set of unstructured text data, expressed as  $D = D(t_1, w_1, t_2, w_2, \dots, t_n, w_n)$ , the feature terms  $t_1, t_2, \dots, t_n$  are considered as

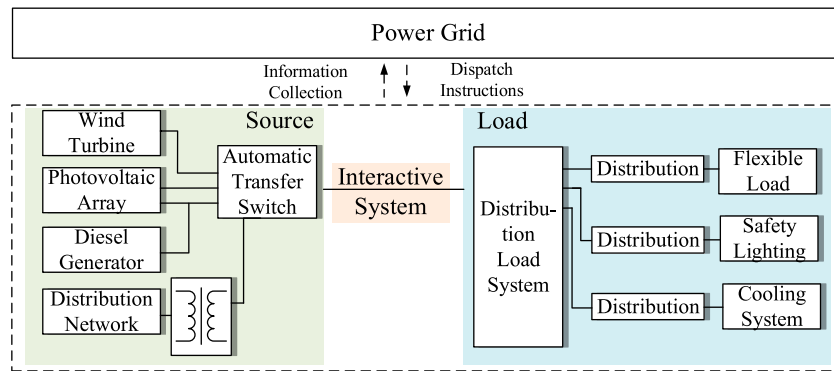


FIGURE 1 Data acquisition in source-load interaction mode

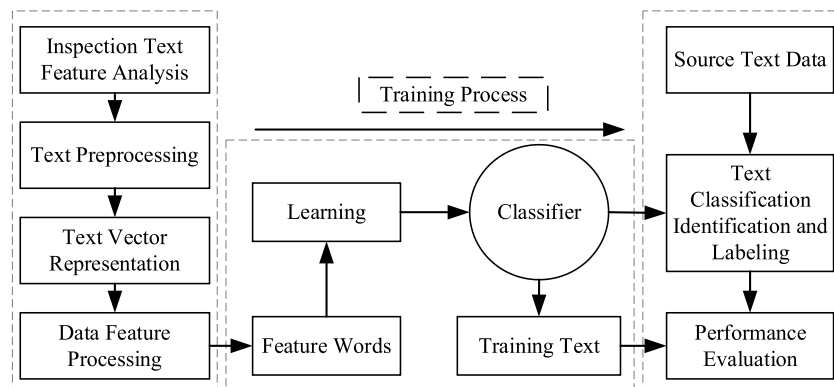


FIGURE 2 Inspection text classification framework.

an  $n$ -dimensional coordinate system, and the weights  $w_1, w_2, \dots, w_n$  are the corresponding coordinate values, so that the unstructured data is expressed as a vector in the space,  $D = D(t_1, w_1, t_2, w_2, \dots, t_n, w_n)$  is the vector space model of the unstructured data.

A plurality of tags are set in the inspection text data, and the tags are used to characterize the attributes of each data in said multi-source data such as the source and type of data; the multi-source data is classified according to the plurality of tags to obtain a plurality of sub-datasets, e.g., the multi-source data set is  $X$ , which is divided to obtain a plurality of sub-datasets  $X_1 \sim X_n$ . The expression of the divided data is:

$$X = (X_1, X_2, \dots, X_n)$$

The sub-data set is normalized according to the following equation:

$$x_i = \frac{x'_i - x_{\min}}{x_{\max} - x_{\min}}$$

where  $x'_i$  is any element of the sub-dataset,  $x_{\min}$  is the smallest element of the sub-dataset,  $x_{\max}$  is the largest element of the sub-dataset, and  $x_i$  is the element of the sub-dataset after normalization;

The normalization process yields a subset containing the normalized subdata set with the expression

$$X_q = [x_1 \ x_2 \ \dots \ x_l]^T$$

where  $l$  denotes the data length of this sub-data set.

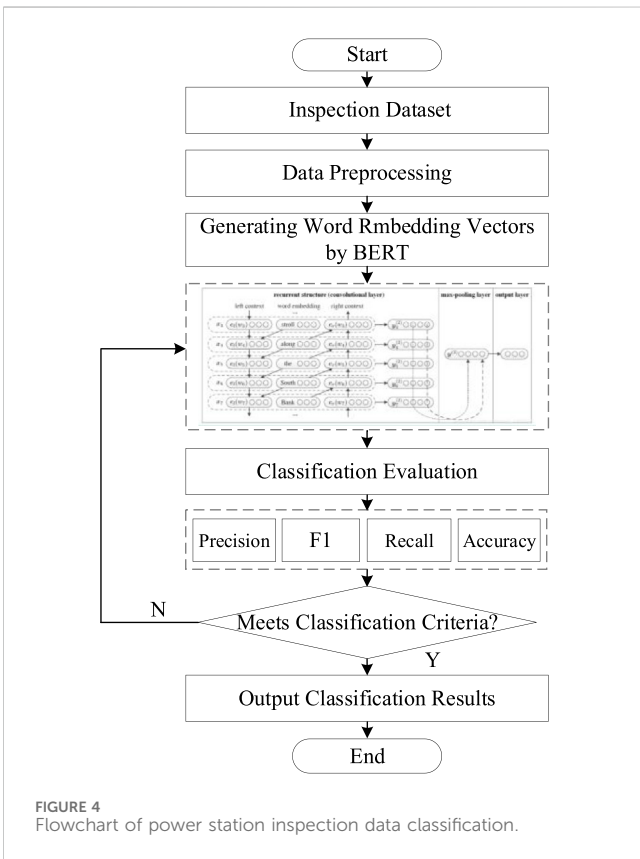
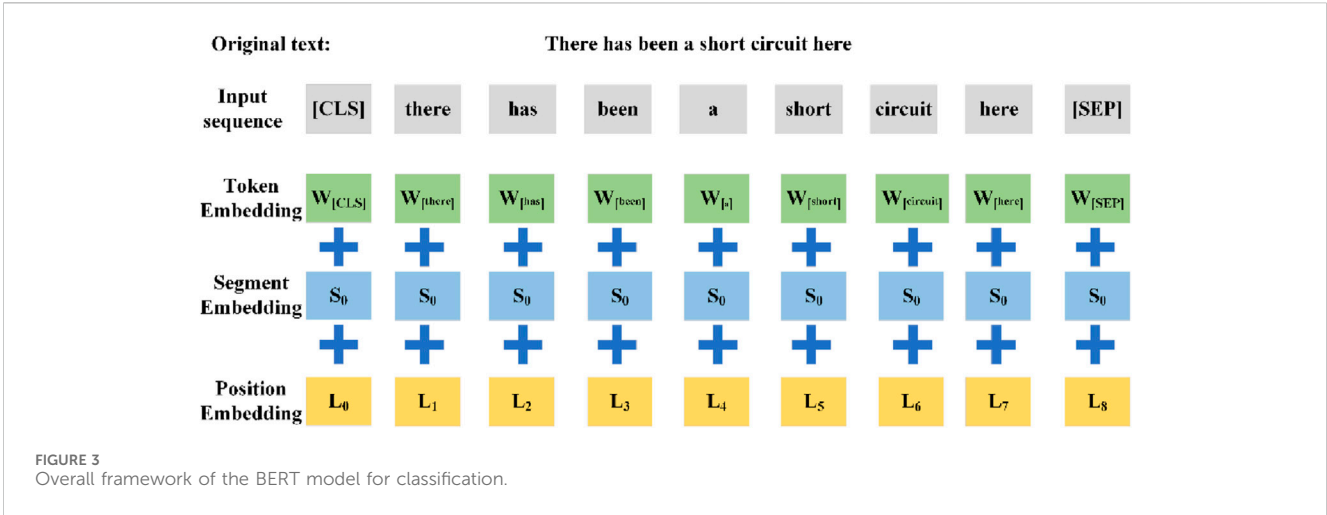
Select  $x_1 \sim x_n$  as the first row vector of the submatrix, and  $n$  is the row length of the submatrix. Shift the first row vector right by one step  $n$  to get the second row vector of the submatrix, and so on, to get the submatrix  $X_{sl}$  with the expression

$$X_{sl} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ x_{n+1} & x_{n+2} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{(m-1)n+1} & x_{(m-1)n+2} & \dots & x_{mn} \end{bmatrix}_{m \times n}$$

where  $X_{sl}$  is the submatrix,  $l$  is the number of subdata set elements,  $m$  is the number of columns of the submatrix,  $n$  is the number of rows of the submatrix, and  $m \times n = l$ .

### 3.3 Building BERT-TextRCNN classifier

Figure 3 shows the structure of the classification model based on the BERT pre-training model. The input data are preprocessed by



adding a special token [CLS] at the first place of the text sequence to indicate a text or a sentence pair, and adding [SEP] at the separation of the sentences to indicate the embedding vector of the mnemonic words, the embedding vector of the sentence words and the embedding vector of the positional words,  $T_{oki}$  indicates the  $i$ th token of the sentence, and randomly masks part of it during the training process.  $E_i$  denotes the embedding vector of the  $i$ th token, and  $T_i$  denotes the final feature vector of the  $i$ th token after processing by the model. In order to better extract textual knowledge, this article uses a BERT model trained on Chinese texts.

In order to apply BERT to the classification task of inspection case text, this paper adds a classification layer (fully connected layer) after the output layer, and the output of the classification layer will get the probability of three types of labels after the softmax function calculation, and the label corresponding to the final probability maximum is the result of the model classification. The fine-tuning process involves supervised training of the model in the dataset, which changes the weight matrix of the model. The weight matrix of the model will change depending on the training parameters, and multiple sets of training parameters are set to enable the model to achieve the best classification results in the dataset.

In this paper, we use the BERT pre-training model as the embedding method, and denote  $X$  and  $X'$  as two sentences such that:

$$X = \{x_1, x_2, \dots, x_m\}$$

$$X' = \{x'_1, x'_2, \dots, x'_n\}$$

Where  $X$  and  $X'$  are used as the input of the embedding layer, and the output is obtained after passing the BERT model [CLS] and [SEP] are two special symbols in BERT, where [CLS] is used as the starting character of the input and will be output as the hidden layer of the sentence pair after BERT operation [SEP] is the separator to distinguish two different sentences.

The word embedding vector obtained by the BERT model, in order to further explore the long-range dependencies and contextual semantic knowledge, the word embedding vector is fed into the bidirectional recurrent neural network BiLSTM to encode the feature vector from both positive and negative directions, and the feature extraction of semi-structured power text data is programmed according to the BiLSTM computational logic, and the BiLSTM model is calculated as follows:

$$z_t = \sigma(W_z \cdot [c_{t-1}, x_t])$$

$$r_t = \sigma(W_r \cdot [c_{t-1}, x_t])$$

$$h_t = \tanh(W \cdot [r_t \cdot c_{t-1}, x_t])$$

$$c_t = (1 - z_t) \cdot c_{t-1} + z_t \cdot h_t$$

where  $\sigma$  is the sigmoid function,  $W$  is the weight matrix,  $x_t$  denotes the input text data at moment  $t$ ,  $c_{t-1}$  is the input text data at the

previous moment,  $r_t$  denotes the reset gate,  $z_t$  is the update gate,  $h_t$  is the state to be activated, and  $c_t$  is the output at the current moment.

The update formula for the left-to-right recurrent neural network layer of the bidirectional BiLSTM is

$$\vec{h}_t = f(\vec{W}x_t + \vec{V}h_{t-1} + \vec{b})$$

The update formula for the right-to-left recurrent neural network layer is

$$\overleftarrow{h}_t = f(\overleftarrow{W}x_t + \overleftarrow{V}h_{t+1} + \overleftarrow{b})$$

The final output of the BiLSTM neural network is calculated as

$$y_t = g\left(U\left[\begin{matrix} \vec{h}_t \\ \overleftarrow{h}_t \end{matrix}\right] + c\right)$$

Where,  $W, V, U$  are weight matrices,  $b, c$  are bias matrices, and  $h_t$  is the state to be activated.  $f$  and  $g$  denote conventional mathematical functional relationships.

$$y = \max_{i=1}^n y_i$$

The max function here is a function expressed by element max, which means that the  $y$  obtained from the previous word expression is an  $n$ -dimensional vector, and the  $k$ th element of  $y$  in this step is the maximum value of the  $k$ th element of all vectors in the previous step  $y$ . The pooling layer converts text of different lengths into fixed length vectors. By using pooling layers, we can capture information throughout the entire text and attempt to identify the most important underlying semantic factors.

The last part of the model is the output layer:

$$y^{out} = Wy + b$$

## 4 Experiment and analysis

### 4.1 Experimental dataset

In this paper, through the first quarter of 2023, i.e., January 2023 March 2023, of a distribution substation, 2,831 maintenance data collected from the distribution room, of which about 58 inspection data from the ring room, 738 inspection data from the ring box, about 672 inspection data from the distribution room, and about 1,363 inspection data from the box substation, the inspection data are divided into training set, validation set and test set. Table 1 gives an example of the type of text for a distribution station check. For the proposed text corpus, the data set is preprocessed. The workstation configuration used in this paper is NVIDIA GeForce GTX 1660 SUPER, Python version 3.7, PyTorch version 1.5, and MySQL version 5.7.

### 4.2 Evaluation index

In this paper, the following evaluation metrics are introduced to assess the final classification effect of the classifier:

When doing text classification tasks, four evaluation metrics are often used to determine the merit of text classification results: accuracy, precision, recall and F1, which are calculated as follows.

$$\begin{aligned} accuracy &= \frac{TP + TN}{TP + FP + TN + FN} \\ precision &= \frac{TP}{TP + FP} \\ recall &= \frac{TP}{TP + FN} \\ F1 &= \frac{2 \times precision \times recall}{precision + recall} \end{aligned}$$

In the formula,  $TP$  represents that this type of sample is positive and the prediction result is positive;  $FP$  indicates that this type of sample is negative and the prediction result is positive;  $TN$  indicates that this type of sample is negative and the prediction result is negative;  $FN$  indicates that this type of sample is positive and the prediction result is negative.

### 4.3 Experimental results

In the BERT-TextRCNN model, adjusting parameters such as the number of epochs and learning rate can optimize the accuracy of the model training process. Therefore, to investigate the impact of the number of epochs on the model, we set the number of epochs to 100 and 200 respectively and obtained the training accuracy results for the inspection text as shown in Figure 4.

The effect of epoch period on the model is explored in Figure 5, and it is found that the accuracy of the model in the test set is 91.7% and the accuracy of the model in the training set is 94.8% when the epoch period is 100, and the accuracy of the model in the test set is 91.5% and the accuracy of the model in the training set is 96.1% when the epoch period is 200. The accuracy of the model can reach more than 90% in both epoch periods, and the accuracy of the model in the training set is improved by increasing the number of epochs. The reason for this is that the increase of epoch period makes more features extracted from the data set, but the accuracy of the model in the test set does not change with the increase of epoch number, and the model overfitting is considered.

Meanwhile, to explore the effect of learning rate on the model, the epoch period is set to 200, and the learning rate is set to 0.0001 and 0.001 for training respectively, and 25% of the training set is selected for validation, and the accuracy of the training results of the inspection text is derived as shown in Figure 6.

From the training results with different learning rates in Figure 6, it can be seen that the accuracy of the test set model is 88.8% and the accuracy of the training set model is 92.8% when the learning rate is at 0.0001, and the accuracy of the test set model is 91.3% and the accuracy of the training set model is 96.8% when the learning rate is at 0.001. A higher learning rate can improve the accuracy of model validation.

Considering that the model will be overfitted in the test set, a Dropout layer is inserted between the input and the first hidden layer. By setting the Dropout rate from 0.1 increment to 0.9, the results of automatic classification of distribution substation inspection text on the test set are as follows in Figure 7.

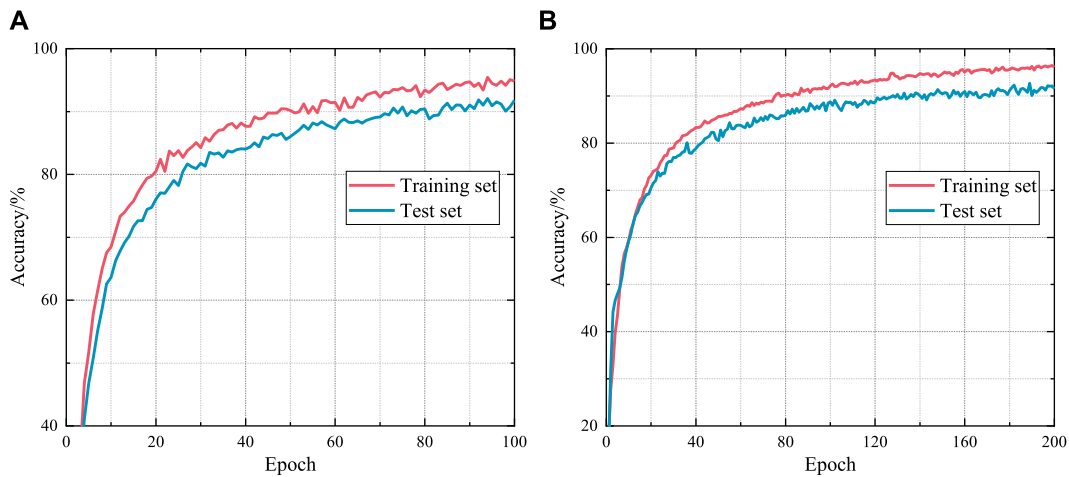


FIGURE 5 Training results of different epochs. (A) epoch period = 100, (B) epoch period = 300.

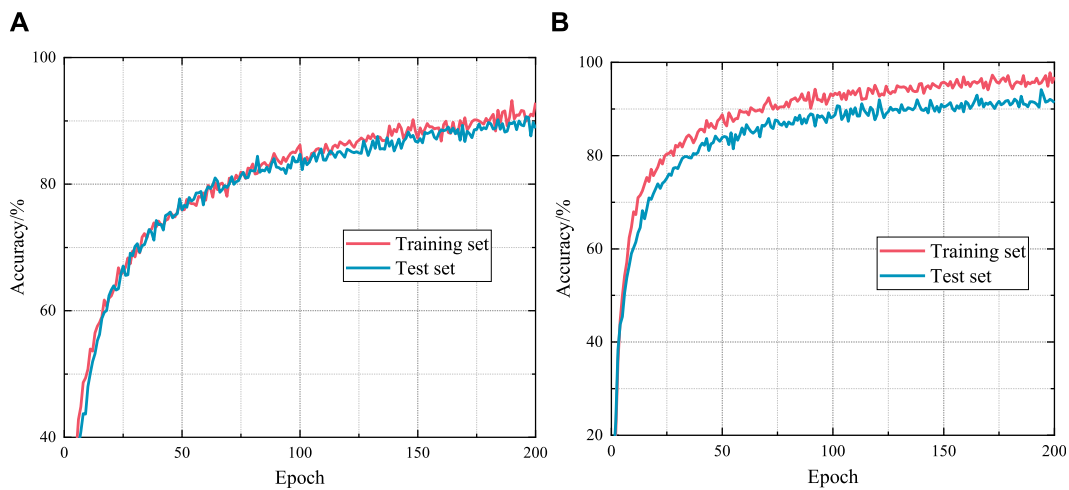


FIGURE 6 Training results with different learning rates. (A) learning rate = 0.0001, (B) learning rate = 0.001.

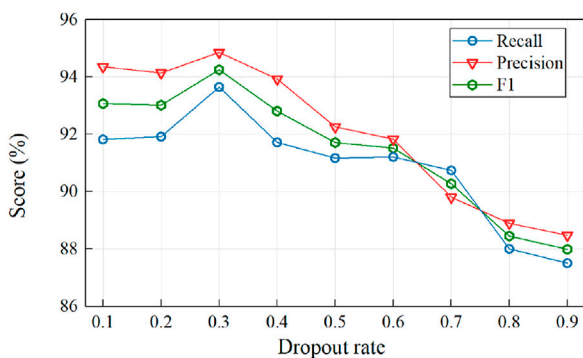


FIGURE 7 Training results at different Dropout rate.

In summary, by exploring the effects of the three parameters on the model, it is concluded that the network model is best trained when the epoch period is 200 and the learning rate is 0.0001, while the Dropout rate is taken as 0.3, the loss decreases and the accuracy rate increases, which finally improves the correct rate of the model to 0.947.

In this study, several classical text classification models, namely, BERT-TextRNN and BERT-TextCNN, were introduced. The training results of different models are summarized in Table 2.

The results from Figure 8 demonstrate that the proposed method outperforms other classification models in terms of recall and F1 score. Due to the adoption of a pretraining-finetuning paradigm, the BERT model's network structure is difficult to adjust, resulting in blind spots during training and poor performance in distinguishing certain samples in the vector

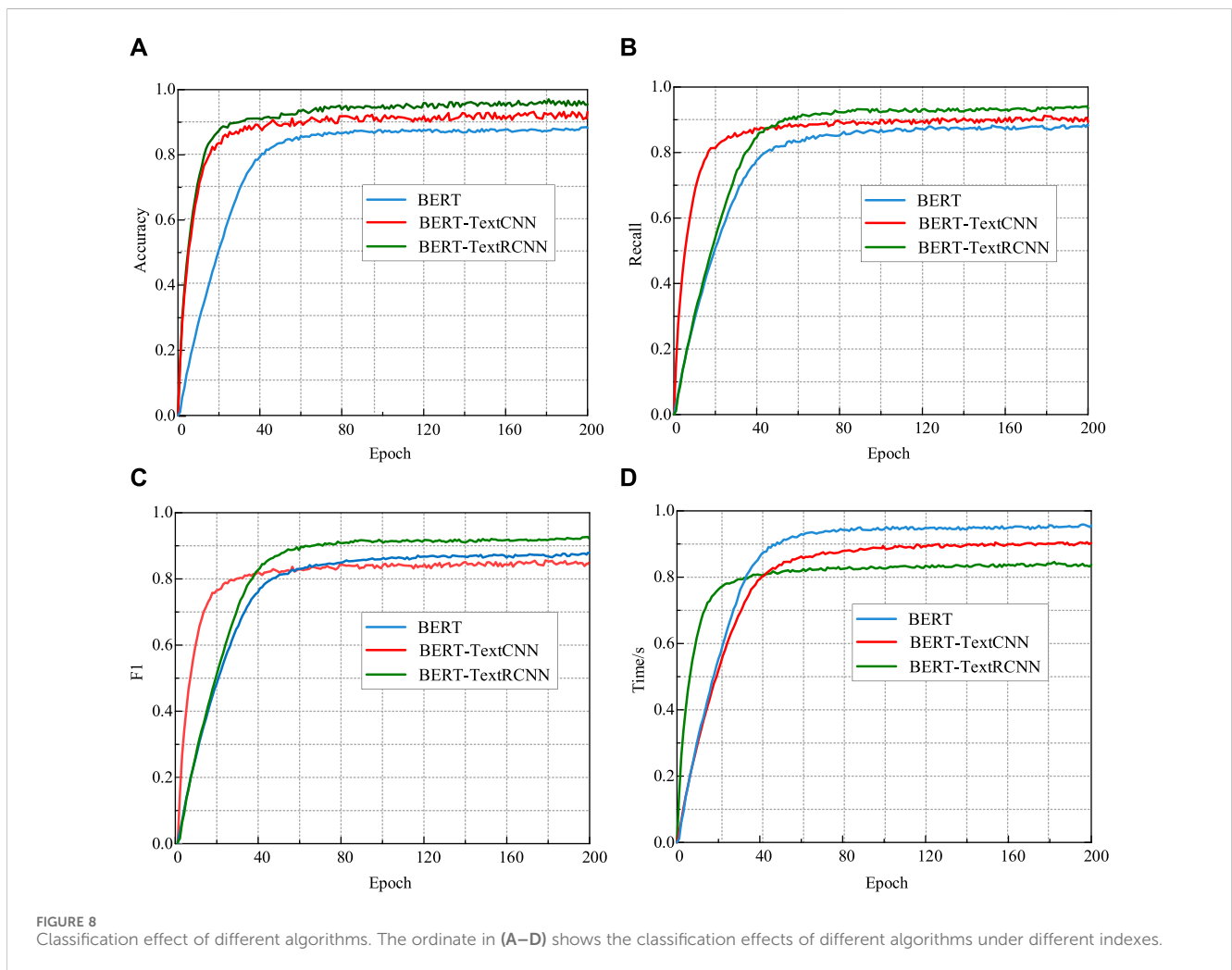
TABLE 1 Example of distribution station inspection text type.

Label	Text type	Text description
0-Time	Time	2023/01/01 00:00:01–2023/03/31 23:59:59
1-Type	Station Type	Types of stations: Ring Main Unit (RMU), Ring Main Room (RMR), Compact Substation, Distribution Room
2- Fault	Fault	Fault information: Switch position, Low-lying location, Missing signage, Tree obstruction, Event alarm

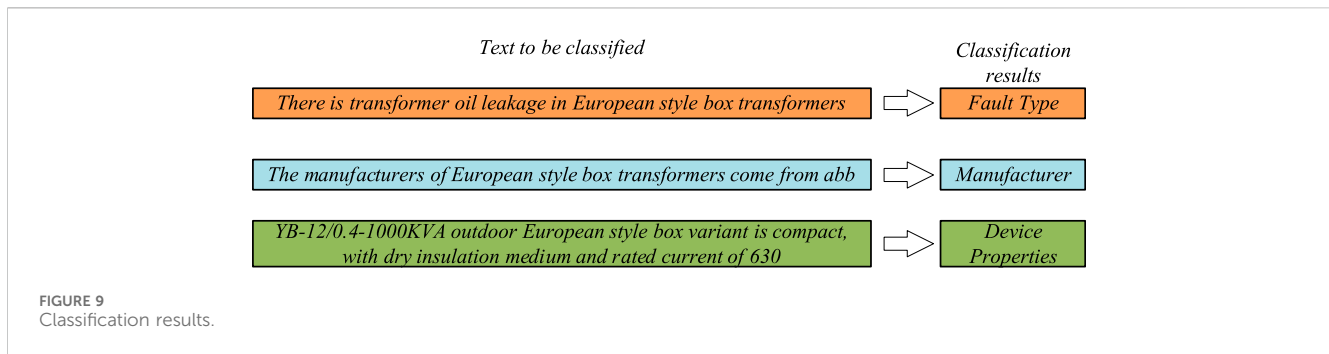
TABLE 2 Results of classification under different algorithms.

Algorithm name	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)
BERT	87.5	87.3	88.1	87.7
BERT-TextCNN	91.5	89.7	90.1	89.9
BERT-TextRCNN	94.7	92.0	92.3	92.1

The results indicate that this study combines BERT, with the TextRCNN, model. By incorporating the BERT, model for word embedding, the issue of different semantics and references of the same word in different contexts is addressed. Additionally, the LSTM, component in RCNN, possesses long-term memory capabilities, allowing for more effective capture of contextual information. Finally, by utilizing a pooling layer to capture the key information from the sequence, the effectiveness of text classification is further enhanced, achieving an accuracy of approximately 94%. Among different algorithms, the BERT-TextRCNN, model demonstrates superior classification performance for inspection text, as depicted in Figure 8.







space. In comparison to BERT-TextCNN, although BERT-TextRCNN shows similar accuracy, the proposed method exhibits a significant advantage in terms of training time. This is because BERT-TextCNN is a common network model for text classification tasks, but improper selection of the size of convolutional kernels can lead to training errors. A kernel that is too small may cause the loss of important information during the convolution operation, while a kernel that is too large increases the number of parameters and extends the training time. In contrast, BERT-TextRCNN addresses these issues. On one hand, TextRCNN utilizes a pooling layer to extract the core information from the sequence, automatically determining which features carry a greater weight in the classification task. Additionally, a Dropout layer is employed to prevent overfitting. On the other hand, TextRCNN adopts a bidirectional recurrent structure to capture contextual information, enabling the acquisition of a larger window of context during training. This not only reduces noise but also decreases information redundancy, thereby improving the efficiency of the algorithm. The classification results are shown in Figure 9.

## 5 Conclusion

In this paper, a BERT-TextRCNN model for text classification is proposed. BERT is utilized as the embedding layer, providing inputs to the TextRCNN model, which effectively extracts local features and enables the classification of inspection record text data. By employing pre-training and fine-tuning, BERT is applied to the classification task of inspection record text, resulting in three key information categories required for equipment maintenance and enhancing the analysis efficiency of inspection documents. Additionally, the BERT-TextRCNN model incorporates a bidirectional recurrent structure, which significantly reduces noise and maximizes the capture of contextual information compared to traditional window-based neural networks. Furthermore, the model demonstrates linear time complexity, directly proportional to the text length, resulting in a significant reduction in computation time and a substantial improvement in computational efficiency. Finally, experiments are designed to classify a dataset of 2,831 inspection record reports, providing data support for advanced applications of inspection text data. The experimental results demonstrate the automatic classification of substation text using the proposed method, achieving an accuracy

of 94.7%, precision of 92%, recall of 92.3%, and an F1 score of 92.1%.

## Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

## Author contributions

JL: Writing—original draft. ZR: Writing—original draft. YZ: Writing—original draft. DY: Writing—review and editing. SJ: Writing—original draft. YT: Writing—review and editing.

## Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by China Southern Power Grid Corporation under Grant (GDKJXM20210063). This work was supported by China Southern Power Grid Corporation under Grant (GDKJXM20210063).

## Conflict of interest

The authors declare that this study received funding from Guangdong Power Grid Limited Liability Company. The funder was involved in the whole study, including the establishment and training of Bert textrcnn model, and the source and cleaning of data from distribution stations.

## Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

## References

- Cai, Y., Xiao, X., Tian, H., Fu, Y., Wu, P., and He, H. (2021). A multi-source data collection and information fusion method for distribution network based on IOT protocol. *IOP Conf. Ser. Earth Environ. Sci.* 651 (2), 022076–022077. doi:10.1088/1755-1315/651/2/022076
- Chang, W. C., Yu, H. F., Zhong, K., Yang, Y., and Dhillon, I. (2020). “Taming pretrained transformers for extreme multi-label text classification,” in KDD’20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (ACM), 1–5.
- Choi, H., Kim, J., Joe, S., and Gwon, Y. (2021). “Evaluation of Bert and Albert sentence embedding performance on downstream nlp tasks,” in 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, January 10–15, 2021 (IEEE), 5482–5487.
- Coster, W., and Kauchak, D. (2011). “Simple English Wikipedia: a new text simplification task,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Portland, OR, June 19–24, 2011, (United States: Association for Computational Linguistics), 665–669.
- Devlin, J., Chang, M. W., Lee, K., and Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. Available at: <https://arxiv.org/pdf/1810.04805.pdf>.
- Du, X., Jiafeng, Q., Shiyao, G., et al. (2018). Text mining of typical fault cases in power equipment. *High. Volt. Technol.* 44 (4), 1078–1084. doi:10.13336/j.1003-6520.hve.20180329005
- Fang, W., Luo, H., Xu, S., Love, P. E., Lu, Z., and Ye, C. (2020). Automated text classification of near-misses from safety reports: an improved deep learning approach. *Adv. Eng. Inf.* 44, 101060–101065. doi:10.1016/j.aei.2020.101060
- Goh, Y., and Ubeynarayana, C. (2017). Construction accident narrative classification: an evaluation of text mining techniques. *Accid. Analysis Prev.* 108, 122–130. doi:10.1016/j.aap.2017.08.026
- Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.* 9 (8), 1735–1780. doi:10.1162/neco.1997.9.8.1735
- Hong, Z., Gui, F., and Hui, Y. (2019). Text classification feature selection method based on normalized document frequency. *J. East China Univ. Sci. Technol. Nat. Sci. Ed.* 45 (5), 809–814.
- Jing, S., Liu, X., Gong, X., Tang, Y., Xiong, G., Liu, S., et al. (2022). Correlation analysis and text classification of chemical accident cases based on word embedding. *Process Saf. Environ. Prot.* 158, 698–710. doi:10.1016/j.psep.2021.12.038
- Jorge, A., and Balazs, Y. (2019). “Gating mechanisms for combining character and word level word representations: an empirical study,” in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: (NAACL), Minneapolis, MN, 110–124.
- Liao, W., Wang, Y., Yin, Y., Zhang, X., and Ma, P. (2020). Improved sequence generation model for multi-label classification via CNN and initialized fully connection. *Neurocomputing* 382 (3), 8–195.
- Liao, W., Zeng, B., Yin, X., and Wei, P. (2021). An improved aspect-category sentiment analysis model for text sentiment analysis based on RoBERTa. *Appl. Intell.* 51, 3522–3533. doi:10.1007/s10489-020-01964-1
- Meng, F., Yang, S., Wang, J., Xia, L., and Liu, H. (2022). Creating knowledge graph of electric power equipment faults based on BERT-BiLSTM-CRF model. *J. Electr. Eng. Technol.* 17 (4), 2507–2516. doi:10.1007/s42835-022-01032-3
- Williams, R. J., and Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Comput.* 2 (1), 270–280. doi:10.1162/neco.1989.1.2.270
- Xu, W., and Tan, Y. (2019). Semi-supervised target-oriented sentiment classification. *Neurocomputing* 337, 120–128. doi:10.1016/j.neucom.2019.01.059
- Zeng, F., Bie, Z., Liu, S., Yan, C., and Li, G. (2020). Trading model combining electricity, heating, and cooling under multi-energy demand response. *J. Mod. Power Syst. Clean Energy* 8 (1), 133–141. doi:10.35833/mpce.2018.000454