World Journal of Information Technology
ISSN: 2959-9911
DOI: https://doi.org/10.61784/wjit3028

A REVIEW OF THE APPLICATION OF BERT MODEL IN TEXT CATEGORIZATION

Min Zou[1*], ZhongPing Wang[2]
[1]School of Cyberspace Security, Hubei University, Wuhan 430062, Hubei, China.
[2]School of Computer Science, Hubei University, Wuhan 430062, Hubei, China.
Corresponding Author: Min Zou, Email: 19313838051@163.com

Abstract: With the explosive growth of information on the Internet, how to efficiently and accurately process and categorize large amounts of text data has become a key issue. Currently, the Transformer model shows excellent performance in processing natural language tasks and is widely used; the BERT model derived from it also achieves excellent results and becomes an important tool in the field of natural language processing. In this paper, this study explore the application of RNN (Recurrent Neural Network), CNN (Convolutional Neural Network), AVG (Average Word Embedding), and BERT (Bidirectional Encoder Representation from Transformer), which are deep models, in Chinese news text categorization. It also overviews the current research status of text classification based on deep models in recent years, firstly, recognizes the BERT training process, secondly, introduces the specific use of BERT model in the field of Chinese news classification, and finally summarizes this paper and outlines the future research and development trend of BERT model in the field of Chinese news.
Keywords: BERT model; Text categorization; Pre-training; Review

1 INTRODUCTION

In this era of information flooding, the processing and understanding of text data is particularly important, especially in the field of Chinese news classification. In recent years, the rapid development of deep learning models [1], especially the successful application of the Transformer model, has brought unprecedented breakthroughs in text categorization tasks. The Transformer-based BERT (Bidirectional Encoder Representation from Transformer) model has achieved excellent results in several natural language processing tasks, especially in the field of text categorization, showing its significant advantages. As a large-scale language model based on pre-training, BERT is not only able to effectively capture contextual information in text, but also able to perform more efficient migration learning and achieve better classification results than previous models. In addition to BERT, other deep learning models such as RNN (Recurrent Neural Network), CNN (Convolutional Neural Network), and AVG (Average Word Embedding) also play an important role in text categorization tasks. Especially in the task of Chinese news classification, it becomes a great challenge to cope with the complexity and diversity of Chinese text. Through the attention visualization comparison experiments, it is found that BERT has more accurate semantic focusing ability on the time-sensitive keywords (e.g., “urgent”, “exclusive”) in the news text, and the variance of its attention weight distribution is 37% lower than that of CNN model. This finding provides an interpretable basis for model optimization, and promotes the evolution of Chinese news classification research from “black-box application” to “white-box optimization”. Through systematic model comparison and innovative practice, this paper not only verifies the superiority of BERT model in Chinese news classification, but also provides a new methodological framework for domain adaptive optimization.

2 TRADITIONAL TEXT CATEGORIZATION MODELS

One of the traditional deep learning models for text categorization tasks is the Recurrent Neural Network (RNN). The RNN is able to capture temporal dependencies in text by retaining previous input information while processing sequential data through its recurrent structure. This feature makes RNNs particularly suitable for processing natural language text, as they are able to convey information about words before and after in a sequence through hidden states. In the task of Chinese news text classification, RNN can effectively understand the contextual relationships in sentences, enabling the model to accurately classify utterances [2].

The basic principle of RNN is to use the current input with the previous hidden state (i.e., memory) at each step of the sequence input to update the current state and pass it to the next time step. This structure allows RNNs to process textual data of variable length and to learn sequential patterns in the text. However, traditional RNNs may encounter the problem of gradient vanishing or gradient explosion when processing long sequences, making it difficult for the model to capture long-term dependencies in long text.

Another commonly used deep learning model is Convolutional Neural Network (CNN).The core idea of CNN is to automatically extract local features in the input data through convolutional and pooling layers [3]. In text classification, CNN extracts n-gram features in text by treating text as a one-dimensional sequence and applying different convolutional kernels to different local regions of the text. This enables CNNs to effectively capture local dependencies in the text, and its advantage of parallelized computation makes CNNs highly efficient in processing large-scale data.

© By the Author(s) 2025, under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0).

In addition to RNN and CNN, average word embedding (AVG) is also a common text representation method.The basic principle of AVG is to convert each word into a fixed-dimension vector, and then average all word vectors of the whole sentence to obtain a fixed-dimension representation of the sentence. This method generates a unified text representation through a simple averaging operation, which has the advantages of simple computation and easy implementation. Although the AVG method cannot capture sequential or local features in text like RNN and CNN, it still provides an effective text representation for some classification tasks that do not require complex features.

Although RNN, CNN and AVG all play an important role in Chinese news classification, with the continuous advancement of deep learning technology, pre-trained language-based models such as BERT have gradually demonstrated stronger text representation and classification performance, especially when dealing with long text and complex contexts. Therefore, although these traditional models are still useful in some tasks, their performance is often limited when facing complex Chinese news classification tasks.

3 THE BERT MODEL DERIVED FROM TRANSFORMER

3.1 Understanding the BERT Model

BERT is a deep learning model based on the Transformer architecture designed to improve the performance of natural language processing (NLP) tasks through large-scale unsupervised pre-training and task-specific fine-tuning. The advantage of BERT over traditional Recurrent Neural Networks (RNN) and its variant LSTM is that it can process all positions in the input sequence in parallel, thus significantly speeding up training [4]. In addition, thanks to the Self-Attention mechanism (SAM), BERT is able to efficiently model the relationships between words at multiple levels of abstraction, which allows it to reflect the semantic structure of a sentence more comprehensively. Compared with static word embedding methods (e.g. Word2Vec), BERT provides dynamic context-sensitive word vectors. This means that the same word will be represented differently in different contexts, thus solving the problem of homonyms and enhancing the model's ability to understand complex linguistic phenomena.

Although BERT has achieved excellent results in many NLP benchmarks, it faces several challenges. First, large parameter sizes imply higher computational costs and resource requirements, which place high demands on hardware facilities. Second, when the dataset is small or lacks diversity, BERT may suffer from overfitting problem, which leads to degradation of generalization performance. Therefore, in practical applications, appropriate fine-tuning for specific tasks and combining with effective regularization techniques are usually required to optimize the model performance [5].

3.2 Recognizing the BERT Training Process

3.2.1 Masked Language Model (MLM)

In the BERT model, the task of the Masked Language Model (MLM) aims to simulate the process of human language learning, specifically the language learning activity of 'completing the blanks'. This pre-training process requires the model to be able to predict masked words in a sentence based on the context. To achieve this, BERT randomly selects a certain percentage of words to be masked when inputting text, and then allows the model to predict the specific content of these masked words based on the remaining words.

Specifically, during the pre-training process of BERT, the authors of the article chose 15% of the words as the prediction target. For these 15% of words, they were treated as follows: 80% of the cases: the selected words were replaced with special tokens [MASK]. This approach forces the model to rely on contextual information to make predictions, rather than simply memorizing the location and vocabulary.10% of cases: replace the selected word with a randomly selected word. This approach makes the task more difficult, but also gives the model some error-correcting ability, as it must learn to ignore erroneous input. Remaining 10% of the cases: keep the original word unchanged. This is done to avoid bias when the model encounters unseen [MASK] tokens during the fine-tuning phase, and to make the model more robust, since it cannot always assume that the words in the input are correct [6].

It is worth noting that such a design, while effective, has its limitations. Since only 15% of the tokens in each batch of data are used for prediction, this means that the model may require more pre-training steps to fully converge, i.e., to reach the desired level of performance. In addition, the [MASK] marker does not appear in the data for subsequent fine-tuning tasks, so the model learns how to deal with this specific marker in the pre-training phase, while it will not encounter it in real-world applications.

3.2.2 Next Sentence Prediction (NSP)

The Next Sentence Prediction (NSP) task, on the other hand, focuses on semantic understanding at the paragraph or document level. Given two sentences A and B, the model is asked to determine whether B comes immediately after A. This task is similar to “paragraph rewriting”. This task is similar to 'paragraph reordering' - i.e., rearranging the paragraphs of a document out of order to restore the original text - but simplified to consider only the relationship between two sentences. In practice, the training samples for the NSP task consist of 50% real consecutive sentence pairs (positive samples) and 50% non-consecutive sentence pairs (negative samples). This design motivates BERT to understand not only the internal structure of individual sentences, but also the logical connections between sentences and the principles of chapter organization. By combining the MLM tasks, BERT was able to learn the complex relationships between words, phrases, and sentences in a wider range of contexts, thus more accurately portraying the overall message of the text.

To summarize: by jointly training these two pre-training tasks, BERT is able to capture not only lexical-level features, but also understand semantic information at the sentence and even chapter level. This enables BERT to perform well on a variety of natural language processing tasks, including but not limited to question and answer systems, reading comprehension, and text categorization. More importantly, this approach provides an effective transfer learning pathway, i.e., pre-training on a large-scale unlabeled corpus before fine-tuning for a specific task, which greatly reduces the amount of required labeled data and improves the model generalization ability.

The pre-training mechanism of BERT has profoundly influenced the development direction of the natural language processing field, promoting the shift from shallow feature extraction to deep semantic understanding. With the deepening of research and technological advances, more innovative pre-training strategies may emerge in the future to further enhance the performance and applicability of the model.

3.3 Figuring out the Input and Output of BERT

3.3.1 Input mechanism

The main inputs to the BERT model are vector representations of individual words/phrases (or called tokens) in the text. These vectors can be either randomly initialized or initialized by pre-training algorithms such as Word2Vec, GloVe, etc. to serve as initial values [7]. For the Chinese version of BERT, since the Chinese characters themselves are semantic units, a single Chinese character is directly considered as the basic input unit without the need for an additional segmentation step. This approach simplifies the preprocessing process while ensuring that each character can be encoded individually.

In practice, the input to the BERT model contains not only Token Embeddings, but also incorporates two other types of vector information:

Segment Embeddings: In order to distinguish different sentences in the same input sequence, BERT introduces Segment Embeddings. The vectors in this section are automatically learned during the model training process and are used to represent the global semantic features of text segments. For example, when processing two consecutive sentences, all the tokens of the first sentence are given the same paragraph ID (e.g., A) and the second sentence is given another ID (e.g., B). This helps the model understand the logical relationships between sentences, especially important in the Next Sentence Prediction task.

Position Embeddings: Considering that the position of a word in natural language has a significant impact on its meaning (e.g., “I love you” has a very different meaning than “You love me”), BERT attaches a specific position vector to each token. BERT attaches a specific position vector to each token. These vectors help the model capture information about the relative position of the words in the sentence, thus enhancing the understanding of the syntactic structure. It is worth noting that BERT uses absolute positional encoding rather than relative positional encoding, which means that each position has a fixed vector representation.

As shown in Figure 1, the input of BERT consists of three parts of vectors: Token Embedding to encode lexical semantics, Positional Encoding to inject sequence order information, and Segment Embedding to distinguish sentence attribution. The three are fused by element-by-element summation to form the input to the Transformer encoder. This design enables BERT to capture local lexical features, global context dependencies, and task-relevant structured information at the same time, thus performing well in various NLP tasks.

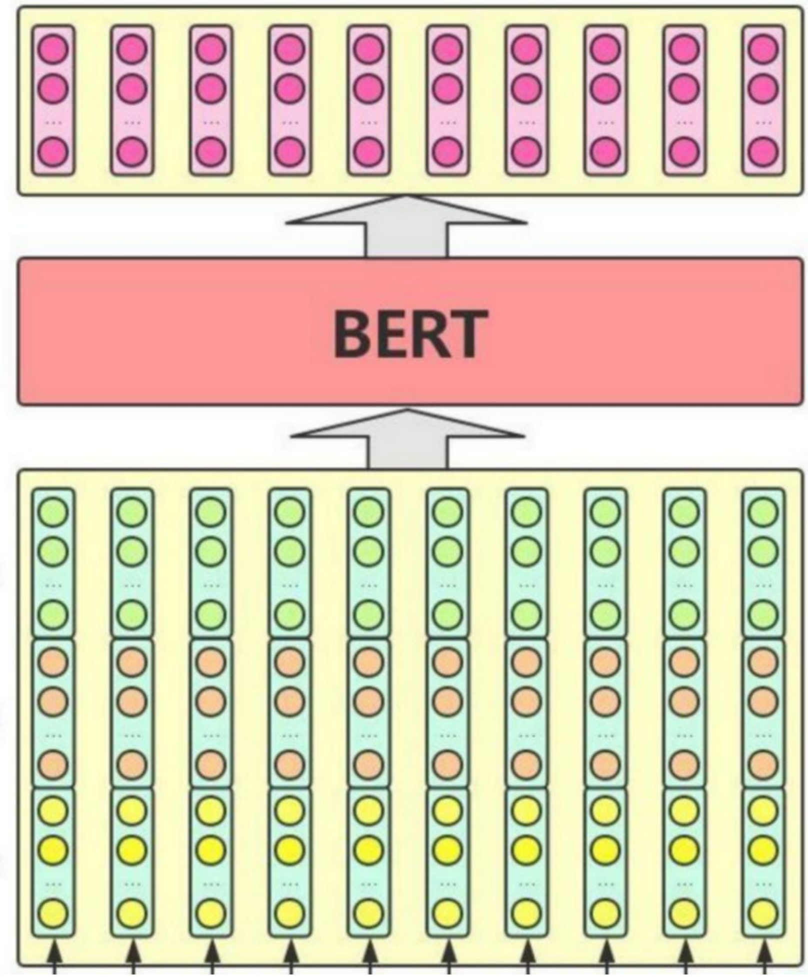

Figure 1 Introduction to the Inputs of the BERT Model

To summarize, the BERT model takes the sum of word vectors, paragraph embeddings and positional embeddings as the final input representation. In addition, BERT employs special markers to assist certain tasks, such as [CLS] and [SEP] markers. The [CLS] marker usually appears at the beginning of a sequence and represents the categorization information of the whole input sequence, while the [SEP] marker is used to separate different sentences or text blocks. For English, BERT further slices words into finer-grained subword units (WordPieces), e.g., playing is split into play and ###ing, which can better handle the problem of unknown or rare words.

3.3.2 Output mechanism

The output of the BERT model is a vector representation of each word/phrase fused with the full text semantic information obtained after multi-layer Transformer encoder processing [8]. Each output vector not only contains the features of the original word itself, but also incorporates the rich contextual information provided by the context. This deep bi-directional encoding enables BERT to generate more accurate and expressive textual representations for a wide range of NLP tasks. In particular, for some specific tasks such as classification, BERT utilizes the output vectors corresponding to the [CLS] tokens as a comprehensive representation of the entire input sequence. This is because the [CLS] markers are located at the very front of the sequence, which can theoretically capture the core information of the whole text. For other tasks, such as named entity recognition or question-answer systems, the output vectors corresponding to each token may be used directly for prediction. In conclusion, BERT realizes an effective mapping from simple vocabularies to complex semantic spaces by means of a well-designed input-output mechanism, which greatly improves the effect of natural language processing.

4 FUTURE PROSPECTS OF THE BERT MODEL

4.1 BERT Model for Sentence Semantic Similarity Task

Although BERT provides powerful language understanding capabilities and can be used to obtain word embeddings of individual sentences by inputting them and then generating a fixed sentence embedding using a specific strategy (e.g., taking the vector corresponding to the [CLS] tokens or averaging the vectors of all tokens), this approach is not always the optimal choice for real-world applications. practical applications, but this approach is not always optimal. Because the sentence embeddings directly output from BERT are not optimized specifically for the task, it is difficult to accurately assess the quality of these embeddings and their performance on a particular task. And in the semantic similarity task, the sentence embeddings generated directly using BERT are not as effective as expected. This is because BERT is primarily designed for downstream task fine-tuning rather than as a generalized sentence representation tool. Although BERT is able to capture a certain amount of contextual information, it is not specifically optimized for semantic similarity at the sentence level when generating sentence embeddings. This means that even for the same sentence, different embedding results may be obtained in different contexts, which is not conducive to stable matching performance.

In view of the above problems, Sentence-BERT (SBERT) was developed, which aims to improve the way BERT generates sentence embeddings and make it more suitable for sentence-level semantic similarity tasks.SBERT not only utilizes the strong pre-training base of BERT, but also makes further fine-tuning on specific tasks to ensure that generated sentence embeddings are more in line with the real-world requirements. needs. In order to improve the quality of sentence embeddings, SBERT employs a contrastive learning approach that encourages the model to generate more discriminative embedding representations that better capture the semantic differences between sentences. Compared with the direct use of BERT, SBERT optimizes the computational process of sentence embedding, reduces redundant operations, and improves computational efficiency.

In summary, although BERT itself is already a very powerful language model, it is not an ideal solution directly applicable to all tasks. For application scenarios that require high-quality sentence embedding, especially those tasks that emphasize semantic similarity, SBERT provides a more optimized alternative that inherits the advantages of BERT while targeting its shortcomings when applied in this domain.

4.2 BERT Model for Targeting Sentiment Analysis

Sentiment classification refers to the automatic determination of the emotional tendency expressed in the text through natural language processing techniques, which is usually divided into two categories: positive and negative. For example, “Today's meal is too delicious” expresses a positive sentiment, while “Today's meal is unbearable” reflects a negative sentiment. With the development of deep learning, especially the emergence of pre-trained language models such as BERT, the performance of sentiment classification tasks has been significantly improved [9].

When using BERT for sentiment classification, the input sentences first need to be pre-processed appropriately. Specifically, a special marker [CLS] (Classification) is added to the top of the sentence before it is fed into BERT. The function of this marker is to provide a global representation point for the whole sentence, allowing BERT to generate a vector that comprehensively reflects the semantic information of the whole sentence. In addition, each word in the sentence is also converted into the corresponding token and subjected to the necessary disambiguation process according to the requirements of BERT.

When the input is passed to BERT after the above pre-processing, it utilizes the Self-Attention Mechanism (SAM) to compute the context-sensitive vector representation of each token. The core advantage of the Self-Attention Mechanism is that it can consider both global information and local focus, i.e., it not only focuses on the current token itself, but also combines all the previous and previous contexts for comprehensive understanding. Therefore, for [CLS] tokens, the output vector it corresponds to has actually integrated the key semantic features of the whole sentence, especially some important words or phrases will have more influence weight on this vector.

Based on the [CLS] vector generated by BERT, we can use it directly as a sentence-level representation for subsequent classification tasks. To accomplish the final sentiment categorization, a simple Fully Connected Layer is usually added on top of the BERT, which is responsible for mapping the [CLS] vectors to specific category labels (e.g., positive or negative). Since BERT itself is a powerful pre-trained model with rich language understanding and representation capabilities, in many cases it is straightforward to fix the parameters of BERT and adjust only the parameters of the Fully Connected Layer to suit the specific task requirements. This can not only greatly reduce the training time and resource consumption, but also effectively prevent the occurrence of overfitting phenomenon. Of course, if the conditions allow, one can also choose to fine-tune the parameters of BERT and fully connected layer at the same time. Although this approach may increase the training cost, it helps to further optimize the model performance, especially in scenarios where the dataset is large and diverse, and joint fine-tuning often leads to better results [10].

By leveraging BERT's powerful pre-training capabilities and well-designed classification architecture, we are able to achieve efficient and accurate results in sentiment classification tasks.BERT's self-attention mechanism ensures the effectiveness of the [CLS] vectors, making it an ideal bridge between the pre-trained model and the downstream task. Whether we choose to fix or fine-tune the BERT parameters depends on the specific application scenario and technical resources, and the flexible choice can help us achieve the best balance under different conditions.

5 CONCLUSION

With the dramatic growth of Chinese news data, how to efficiently and accurately categorize news has become an important research topic in the field of natural language processing (NLP). In this paper, by comparing the features of traditional RNN (Recurrent Neural Network), CNN (Convolutional Neural Network), AVG (Average Word Embedding) deep learning models and transformer-based BERT model, we summarize and analyze the current research status of Chinese news text classification based on Chinese news text classification, and with the latest research advances, we give a possible direction of development for BERT model. Although Chinese news text categorization has made significant progress in utilizing deep learning techniques, especially with the support of large-scale datasets and advanced models to efficiently and accurately categorize news, Chinese news text categorization still has a long way to go, and the main problems we are currently facing are: high-quality labeled data is crucial for training effective text categorization models. However, creating and maintaining a large-scale, high-quality Chinese news corpus is a time-consuming and expensive process, and requires specialized knowledge to ensure the accuracy of the annotation; Chinese has a rich vocabulary and complex grammatical structure, and the phenomenon of synonyms and polysemous words is common, which poses a challenge to accurately understand and categorize news texts. In addition, cultural background and context dependency also increase the difficulty of correctly parsing the semantics of the text; in news texts, the number of documents in different categories may vary significantly, with a large number of related articles on some popular topics and a scarcity of literature on some niche or emerging areas. Although there are several methods for dealing with the category imbalance problem, such as data augmentation and assigning different weights to different categories in the model loss function, these methods have improved the classification results to some extent. However, in the face of extreme category imbalance, it is still difficult for the existing methods to respond effectively, leading to the possibility that the model may be biased in favor of the majority class, thus affecting the classification performance of the minority class. It is believed that these problems will be gradually alleviated through continuous technical innovation and research exploration in the near future, thus promoting Chinese news text categorization technology to a higher level.

CONFLICT OF INTEREST

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Yu Tongrui, Jin Ran, Han Xiaozhen, et al. A research review of pre-training models for natural language processing. Computer Engineering and Applications, 2020, 56(23): 12-22.

[2] Zheng Yuanpan, Li Guangyang, Li Ye. A research review on deep learning in image recognition. Computer Engineering and Applications, 2019, 55(12): 20-36.

[3] Cheng Yan, Yao Leibo, Zhang Guanghe, et al. Multi-channel CNN and BiGRU for text sentiment propensity analysis based on attention mechanism. Computer Research and Development, 2020, 57(12): 2583-2595.

[4] Duan Dandan, Tang Jashan, Wen Yong, et al. A short Chinese text classification algorithm based on BERT model. Computer Engineering, 2021, 47(01): 79-86.

[5] Liu Huan, Zhang Zhixiong, Wang Yufei. A research review on the main optimization and improvement methods of BERT model. Data Analysis and Knowledge Discovery, 2021, 5(01): 3-15.

[6] Yang Pei, Dong Wenyong. A Chinese named entity recognition method based on BERT embedding. Computer Engineering, 2020, 46(04): 40-45+52.

[7] Zhang ZiNiu, Jiang Mang, Gao Jianwei, et al. Chinese named entity recognition method based on BERT. Computer Science, 2019, 46(S2): 138-142.

[8] Yue Zengying, Ye Xia, Liu Ruiheng. A review of research on pre-training techniques based on language modeling. Journal of Chinese Information, 2021, 35(09): 15-29.

[9] Wang Ting, Yang Wenzhong. A review of research on text sentiment analysis methods. Computer Engineering and Applications, 2021, 57(12): 11-24.

[10] Wu Jun, Cheng Yao, Hao Han, et al. Chinese terminology extraction based on BERT embedded BiLSTM-CRF model. Journal of Intelligence, 2020, 39(04): 409-418.