

Research Article

Qian Wang* and Delin Chen

Microblog sentiment analysis method using BTCBMA model in Spark big data environment

<https://doi.org/10.1515/jisys-2023-0020>

received February 09, 2023; accepted November 16, 2023

Abstract: Microblogs are currently one of the most well-liked social platforms in China, and sentiment analysis of microblog texts can help further analyze the realization of their media value; however, the current task of sentiment analysis based on microblog information suffers from low accuracy due to the large size and high redundancy of microblog data, a microblog sentiment analysis method using Bidirectional Encoder Representation from Transformers (BERT)–Text Convolutional Neural Network (TextCNN)–Bidirectional Gate Recurrent Unit (BiGRU)–Multihead-Attention model in Spark big data environment is proposed. First, the Chinese pre-trained language model BERT is used to convert the input data into dynamic character-level word vectors; then, TextCNN is used to effectively obtain local features such as keywords and pool the filtered features; then, BiGRU is introduced to quickly capture more comprehensive semantic information; finally, a multi-headed attention mechanism is implemented to emphasize the most significant features in order to accomplish the sentiment classification of microblog information task precisely. By comparing the existing advanced models, the proposed model demonstrates an improvement of at least 4.99% and 0.05 in accuracy and F1-score evaluation indexes, respectively. This enhancement significantly enhances the accuracy of microblog sentiment analysis tasks and aids pertinent authorities in comprehending the inclination of individual's attitude toward hot topics. Furthermore, it facilitates a prompt prediction of topic trends, enabling them to guide public opinion accordingly.

Keywords: sentiment analysis, BiGRU, TextCNN, BERT, M-HA

1 Introduction

Social media platforms have proliferated in tandem with the Internet's development and have become the primary means by which individuals communicate their opinions and emotions on the web due to their rapid dissemination, wide audience, and convenient use. A microblog has emerged as a significant platform for users to acquire and distribute information due to its rapid transmission rate and substantial social impact. Presently, it is also a representative online social media platform in China at present, with a huge user group [1–4]. Weibo, being an open-source social media platform, is accessible to all users. On Weibo, users have the ability to update their status through text or other means, as well as share their thoughts on various products, events, or individuals [5,6].

Weibo text data is complex and chaotic on the surface, but it contains subjective sentiment information of the masses in many fields. A small range of sentiment expression may affect a large range of users' sentiment

* **Corresponding author: Qian Wang**, School of Mechanical and Automotive Engineering, Kaifeng University, Kaifeng, Henan, 475000, China, e-mail: wangqian@kfu.edu.cn

Delin Chen: Center of Experimental and Training, Kaifeng University, Kaifeng, Henan, 475000, China, e-mail: chendelin2008@126.com

preferences for different events, products, and characters [7,8]. In case of emergencies, it is also easy to generate online public opinion. If we can dig and use this information in depth, it may have unpredictable value for both society and individuals. It can provide reference information for consumers and producers; conduct sentiment analysis on the relevant comments of a given product, which will help to understand the advantages and disadvantages of the product; and improve the customer satisfaction.

By analyzing the user's comments, we can determine the user's daily preferences and provide personalized suggestions [9–11]. In addition, it can also provide support for the government and other relevant departments, facilitate the government's public opinion monitoring, curb the spread of false information, and maintain social stability and prosperity. Conduct sentiment analysis on the content of microblog and monitor the sentiment tendencies contained therein, so as to understand the users' comments on the products or characters on microblog, or the degree of attention and sentiment changes to the events. It provides a real-time scientific theoretical basis for decision-makers to guide online public opinion in a timely and effective manner, and it can also timely control negative information when generating online public opinion to prevent further expansion of online public opinion [12–14]. Therefore, the monitoring, analysis, and reasonable guidance of public opinion on the microblog network is of great significance and can create value for the country, collective, or individual life [15–17].

As a relatively large open-source online social media platform, microblog is rich in content and large in data. It is difficult to conduct sentiment orientation statistics by artificial means. It is necessary to use the method of sentiment analysis to explore the sentiment orientation of its content [18]. Text sentiment analysis can excavate the opinions or evaluation information contained in the subjective text with subjective sentiment features or with commendatory and derogatory tendencies. By analyzing the content of the text, we can predict the sentiment tendency contained in the text and express it in a more intuitive way. Therefore, a fast and effective sentiment analysis of massive and unstructured microblog text data is a hot topic of current research [19].

The current methods exhibit drawbacks such as inadequate extraction of semantic information, insensitivity to multi-sense words, and overly simplistic model structures that fail to account for generalization. In response, this study proposes a microblog sentiment analysis method using the Bidirectional Encoder Representation from Transformers (BERT)-Text [20] Convolutional Neural Network (TextCNN) [21]-Bidirectional Gate Recurrent Unit (BiGRU)-Multihead-Attention (BTCBMA) model within the Spark big data environment. Multi-headed attention (M-HA) [22], BERT, TextCNN, and BiGRU [23] are its primary components. The following are the four primary contributions of this article:

- (1) For feature extraction, the BERT pre-trained language model is employed to map each sentence to an appropriate dimension and generate dynamic character-level word vectors that are characterized by high dimensionality and abundant semantic information.
- (2) By convolution, the TextCNN model extracts the local features of the text. The maximum pooling layer is added to derive the significant features in the text.
- (3) Using the BiGRU model, we use multiple clause rule recognition to solve the sentiment word disambiguation problem, which achieves more bidirectional acquisition of contextual feature information and greatly reduces the computational complexity.
- (4) In order to enhance the accuracy of sentiment analysis and capture the key information in sentences, M-HA is used to integrate multiple single-headed attention.

The subsequent sections of this article are structured as follows: initially, the second section examined the pertinent research on deep learning as it relates to emotional analysis. Then, the recommended BTCBMA was introduced methodically in Part 3. A thorough experimental comparison was performed on the suggested model in Part 4. Part 5 concludes with a summary of the article and a discussion of potential future research.

2 Related works

At present, many relevant personnel have carried out corresponding work on the analysis and research of text sentiment [24]. Dashtipour et al. [25] suggested an automatic feature engineering approach that leverages deep

learning and combines long short-term memory (LSTM) and convolutional neural networks (CNN) models to classify the sentiment of Persian movie reviews. The suggested approach uses a composite model that combines CNN and LSTM. The outcomes of the simulation demonstrate that the suggested approach significantly enhances the precision of sentiment classification. Nevertheless, the approach lacks comprehensiveness with regard to the extraction of features for sentiment classification. Wang et al. [26] introduced a deep learning approach for sentiment-based sentiment classification. This approach employs weakly labeled data for model training, thereby mitigating the detrimental effects of noisy samples in the weakly labeled data and enhancing the overall performance of the sentiment classification model. The experimental findings demonstrate that the suggested approach exhibits superior classification performance in the sentiment classification task of online hotel reviews compared to the conventional depth model, all while maintaining the same labor cost. Nevertheless, the pace at which the approach converges models must be enhanced. Balakrishnan et al. [27] suggested a deep learning model with sentiment embedding for dynamic analysis of cancer patients' sentiment in online health communities, using a bi-directional LSTM (BiLSTM) model for sentiment dynamic analysis of user posts to measure changes in user satisfaction. The efficacy of the method is demonstrated through experimental results in comparison with other established methods. Nevertheless, the approach is deficient in its capacity to capture contextual semantic information. By combining the outputs of CNN, LSTM, BiLSTM, and Gated Recurrent Unit (GRU) models through stacked integration with logistic regression serving as the meta-learner, Mohammadi and Shaverizade [28] suggested a new method to aspect-based sentiment analysis using deep integration learning. In comparison with fundamental deep learning approaches, this approach enhances the accuracy of aspect-based predictions by 5–20%. However, this approach is too redundant. Cheng et al. [29] suggested a polymorphism-based CNN model. The CNN input matrix is generated by the model through the combination of word vector information, word sentiment information, and word location information. Throughout the training process, the model modifies the weight control to modify the significance of various feature information. Using a multi-objective sample dataset, the efficiency of the suggested model in the sentiment analysis assignment of relevant objects is evaluated in terms of classification effect and training performance. Nevertheless, this methodology is incapable of comprehensively capturing and employing contextual data for sentiment analysis. Elfaik and Nfaoui [30] examined an effective BiLSTM that encapsulated contextual information of Arabic feature sequences forward backward, which improved the results of Arabic sentiment analysis. The experimental outcomes derived from six benchmark datasets for sentiment analysis demonstrate that the suggested method outperforms both baseline traditional machine learning methods and state-of-the-art deep learning models. Nonetheless, this method fails to adequately represent the local features of the text.

In summary, although certain improvements are obtained in some tasks in the aforementioned literature, there are still some limitations, which include low prediction accuracy, inadequate extraction of semantic information, inability to effectively handle multiple meaning words, and overly simple model structure lacking generalization. Consequently, a microblog sentiment analysis method using the BTCBMA model in Spark big data environment is proposed in this article. In order to assess the efficacy of the proposed method, we employed a publicly available benchmark dataset and obtained significant performance in sentiment analysis tasks in the microblogging domain, which effectively solved the problems in the aforementioned literature.

3 Method

First, the sentiment of microblog text is divided into six categories: positive, angry, sad, scared, surprised, and heartless. According to the features of short text, a fine-grained sentiment analysis BTCBMA model based on TextCNN is proposed. The BTCBMA model is summarized as multiple embedded vectors. It uses emoticons and sentiment labels of text to transform into vectors to make sentiment expression stronger and realizes the full use of the relationship between text and labels. BiGRU network is combined with the M-HA mechanism to obtain more comprehensive and deep sentiment features. The last part is the output module. The structure of the BTCBMA model is depicted in Figure 1.

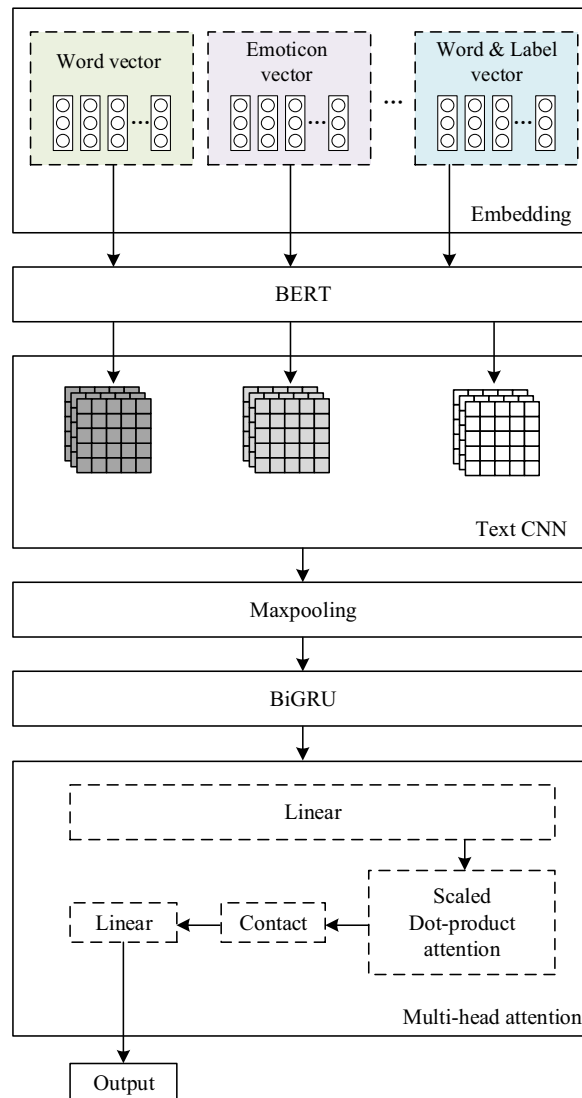


Figure 1: Structure of BTCBMA.

The BTCBMA is mainly divided into five parts:

- (1) **Input layer.** It is used to process the input text; convert the word vector, emoticon, and sentiment label into vector matrix through Word2vec model; and input them to the subsequent network layer.
- (2) **TextCNN layer.** The local information of the text is captured through multi-channel CNN network, and more comprehensive features are captured through different vectors of three channels.
- (3) **BiGRU layer.** Captures context information for text.
- (4) **M-HA layer:** As a supplement to the BiGRU layer, it can fully obtain the global features in the text.
- (5) **Output layer.** After splicing the results of the attention mechanism, the resulting matrix is input into the fully connected network, and finally, the sentiment classification results are output through the activation function.

3.1 BERT pre-trained language models

A dynamic pre-training model is BERT. The fundamental component of BERT is the Transformer encoder, as illustrated in Figure 2 [21].

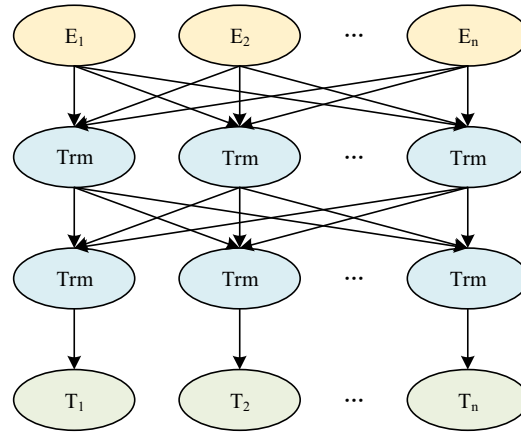


Figure 2: Structure of BERT.

The structure of the BERT model mainly includes three layers:

- (1) Embedding word vector coding layer. Unlike the previous word vector coding, the BERT model will generate three types of embedding for each token in the text input, namely, the token embedding containing the meaning of the word itself, the segment embedding containing the sentence information and the sequence information between sentences, and the position embedding containing the sequence of the words in the sentence. Each token is represented by the sum of the three embedded types. In addition, a [CLS] mark is added at the beginning of each sentence. The [SEP] mark set between sentences is used to distinguish sentences.
- (2) Pre-trained pre-training layer. Two tasks are defined. The first is an integrated two-way language model. The corresponding method in BERT is called Masked language model, or MLM task for short. This method has the capability to simultaneously train two distinct transformer models from right to left and left to right. It is capable of extracting more comprehensive context features from the text, generating a more robust semantic representation, and implementing an integrated two-way language model. The other is Next presence prediction, which is called Next Sentence Prediction task for short. This task is mainly aimed at judging the correlation between sentence pairs with sentence pairs as input in downstream tasks. The basic idea is to select sentence pairs. For example, the sentence pair contains sentence A and sentence B, and then judge whether the next sentence of sentence A is sentence B or other sentences in the corpus. This is a two-class problem. The key of this task is to artificially add a [CLS] mark in front of each sentence. After training, the information is integrated into the vector of the position, which can be used to represent each sentence, so the next sentence can be predicted according to the previous sentence.
- (3) Fine-tuning layer. The task of sentiment classification in this article is to classify a single sentence. It only needs simple transformation on the output of Transformer at the last layer of BERT. Connect a full-connection layer and sigmoid or Softmax function to the last layer of Transformer output corresponding to the CLS position of the starting symbol.

3.2 TextCNN model

CNN consists of three primary layers: convolution, pooling, and full connection. The convolution kernel is employed in the core convolution layer to extract features. The pooling layer's objective is to reduce the dimension of features in order to mitigate overfitting and simplify the computations required by the Softmax classifier. As a result, CNN possesses the capabilities of weight sharing, dimensionality reduction, local feature extraction, and multi-level structure. CNN focuses on capturing local features.

Some scholars have made improvements to CNN and proposed TextCNN for sentiment analysis. TextCNN model has a simpler structure and can be better applied to the field of text sentiment analysis, which is shown in Figure 3 [22].

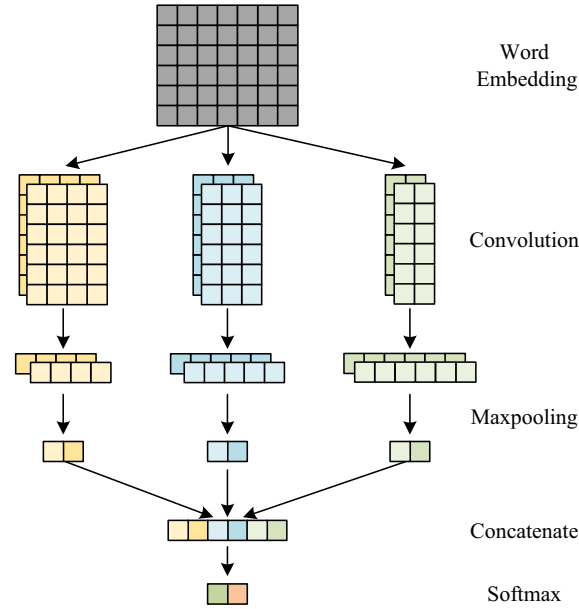


Figure 3: Structure of the TextCNN.

3.3 BiGRU model

Neural networks are frequently employed in sentiment analysis tasks to extract additional text features. In order to classify the sentiment of microblog text, it is necessary to take into account the overall semantics of the text. RNN, LSTM, and other models are typically employed to acquire additional contextual feature information. While LSTM and GRU share comparable architectures and performance metrics, the GRU network exhibits a reduced computational complexity. GRU is more suitable than LSTM for sentiment analysis of microblogs due to the computationally intensive nature of text-related sentiment analysis tasks in general.

Based on the semantic expression features of Chinese, usually the meaning of a word or word is not only related to the preceding text, but also related to the following text, so here we choose to use the BiGRU network structure to extract features. BiGRU consists of both reverse and forward GRU. It extracts from the context the underlying sentiment features of the text. The internal architecture is illustrated in Figure 4 [23].

In Figure 4, $[W_1, W_2, \dots, W_{N-1}, W_N]$ refers to the word vector generated by the transformation of vector representation model, and N refers to the sentence length. $[h_1, h_2, \dots, h_{N-1}, h_N]$ refers to the output of BiGRU, which is obtained by splicing the output $[\vec{h}_1, \vec{h}_2, \dots, \vec{h}_{N-1}, \vec{h}_N]$ of the forward GRU and the output $[\overleftarrow{h}_1, \overleftarrow{h}_2, \dots, \overleftarrow{h}_{N-1}, \overleftarrow{h}_N]$ of the reverse GRU. BiGRU inherits the advantages of simple and fast GRU training and can combine context to eliminate ambiguity and read the entire text more accurately.

3.4 M-HA model

The attention mechanism operates on the principle of mapping the Query and Key-Value sets; its output is determined by performing a weighted sum operation on the weight value and the value associated with the Key. M-HA does not only calculate attention once, but also performs multiple Scaled Dot-Product Attention (SDPA) calculations in parallel. Each attention can learn feature representation in multiple representation spaces after splicing all outputs after independent calculation. The different emphasis of learning in each representation subspace results in different extracted features. Therefore, M-HA has more powerful performance in feature representation. SDPA reduces the result in order to prevent the gradient from disappearing due to excessive internal product. The structure of M-HA is depicted in Figure 5 [20].

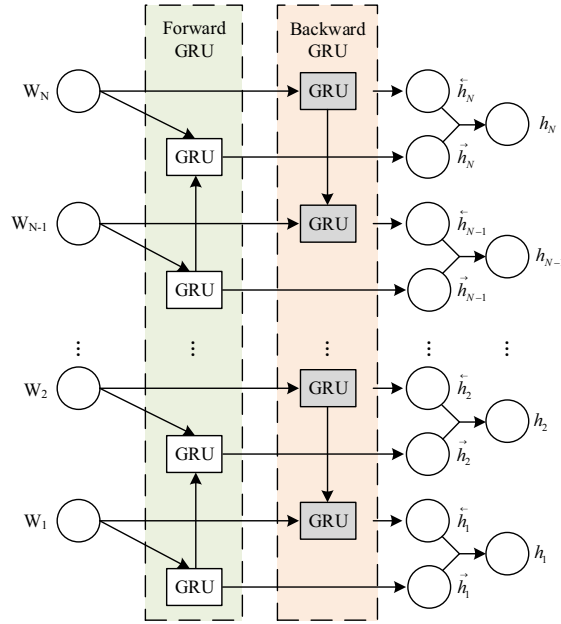


Figure 4: Structure of the BiGRU model.

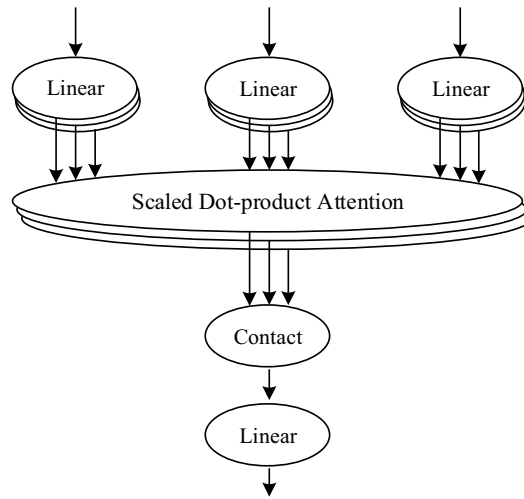


Figure 5: Structure of the M-HA model.

The calculation process of M-HA is as follows:

- (1). First, the Query, Key, and Value are linearly mapped in different ways.
- (2). Perform SD-PA operation on m times of different linear maps.
- (3). The output obtained in step (2) is spliced and input to the linear mapping layer to get the result.

M-HA is shown in the following formula:

$$M_H(Q, K, V) = \text{Concat}(h_1, h_2, \dots, h_m)W_0, \tag{1}$$

where $h_i = \text{Atte}(Q, K, V)$, $i = 1, 2, \dots, m$, and W_0 represents the weight of linear mapping layer.

3.5 Maxpooling layer

The feature graph resulting from the convolution operation has a substantial dimension. By incorporating a pooling layer into the parameter matrix, it is possible to reduce its dimension. This can effectively prevent overfitting while retaining the required features. The maximum pooling method and the average pooling method use the maximum value and the average value in pooling window, and their principles are illustrated in Figure 6.

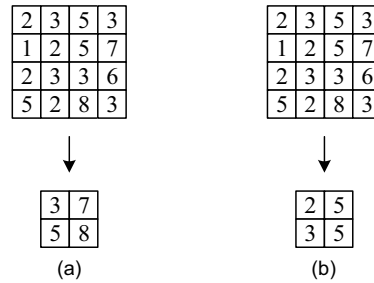


Figure 6: Pooling method: (a) maximum pooling and (b) mean pooling.

After transferring the processed features from the pooling layer to the full connection layer, the Softmax is applied to the feature vectors of the full connection layer for classification. With average pooling, it is simple to blur text information. However, maximum pooling can inexpensively substitute for the convolution layer, and its translation invariance is another reason why it is so useful in CNN. A translational model signifies that the location of the object is inconsequential; it will be identified regardless. By incorporating translation invariance into the model, its predictive capability will be significantly enhanced, as it will be superfluous to provide information regarding the precise position of the object. Consequently, in this case, the maximal pooling operation is chosen.

4 Experiment and analysis

4.1 Experimental environment

The computer provided by the laboratory is used as the experimental platform. The hardware configuration and system environment, respectively, of the platform are detailed in Tables 1 and 2.

Table 1: Hardware configuration of the experimental platform

Parameters	Configuration
CPU	Intel(R) Core i5-750
CPU memory	16G @ 2666 MHz
GPU	NVIDIA Geforce GT 710
GPU memory	1TB HDD + 128GB SSD

Table 2: Experimental system environment

Parameters	Configuration
Operating system	Windows 10
Python version	Python 3.6.10
CUDA version	CUDA 8.0.61
Deep learning framework	PyTorch 1.1

4.2 Dataset

The experimental data used in this study is the Weibo negative sentiment dataset, a newly self-labeled collection. The dataset comprises 15 sentiment labels, each containing 20,000 data points, for a cumulative count of 300,000 items. Comparative experiments are performed in this article on various dataset division ratios in light of the analysis of conventional textual multi-sentiment classification tasks. In the proportion of 6:2:2, the dataset is partitioned into a training set, validation set, and test set.

In order to verify the validity of the proposed model, we also use a publicly available benchmark dataset from the following source: https://github.com/SophonPlus/ChineseNlpCorpus/blob/master/datasets/simplifyweibo_4_moods/intro.ipynb. The simplifyweibo_4_moods data consists of more than 360,000 sentiment-labeled Sina Weibo messages, including four sentiments: joy, anger, disgust, and depression, with about 200,000 joy messages and 50,000 anger, disgust, and depression messages. In the ratio of 6:2:2, the benchmark dataset is grouped into training set, validation set, and test set.

4.3 Evaluating indicator

Here, the accuracy (A) and F1 values are chosen as the evaluation indicators of sentiment classification. By dividing the sentiment classification results into positive, neutral, and negative, the sentiment classification task is programmed into three categories of multi-class tasks. Therefore, it can be regarded as three dichotomous tasks, i.e., each category is regarded as “positive” and the other two categories are regarded as “negative.” Then, calculate the three categories A. The formula for calculating the A of each category is as follows (2):

$$A = \frac{T_P + T_N}{T_P + T_N + F_P + F_N}, \quad (2)$$

where T_P represents the number of samples that the classifier classifies the exception data into exception types, T_N represents the number of samples that the classifier classifies normal data into normal types, F_P represents the number of samples that the classifier classifies normal data into attack types, and F_N represents the number of samples that the classifier classifies abnormal data into normal types.

In multi-class tasks, F1 can be calculated in two ways: Micro-F1 and Macro-F1. Among them, Micro-F1 is suitable for unbalanced data and Macro-F1 is more suitable for general multi-class tasks. The sentiment classification task here is three categories. After preprocessing, the proportion of the three types of sentiment tendency data is as follows: “positive:neutral:negative = 28.41%:50.32%:21.27%,” about 1:2:1, which is not unbalanced data. Therefore, Macro-F1 is selected as the evaluation index.

Macro-F1 divides the comment text of N classification into N comment text of two classification and then calculates the F1 of each two classifications. After N F1 are obtained, average them. The specific solution process is shown in the following equation:

$$\begin{cases} \text{macro_F1} = \frac{F1_{(1)} + F1_{(2)} + F1_{(3)}}{3} \\ F1_{(k)} = \frac{2P_k \cdot R_k}{P_k + R_k}, \quad k = 1, 2, 3 \end{cases}, \quad (3)$$

where P_k is the precision, k is the category, and the calculation process of P_k for each category is shown in the following equation:

$$P_k = \frac{TP_k}{TP_k + FP_k}, \quad (4)$$

where R_k is the recall rate. The calculation process of R_k for each category is shown in the following equation:

$$R_k = \frac{TP_k}{TP_k + FN_k}. \quad (5)$$

4.4 Model training

In order to evaluate the efficacy of the model during the experiment, the accuracy of the test set and the Macro-F1 was chosen. The loss value of the proposed BTCBMA model on the dataset, which was used for both training and sentiment classification, is illustrated in Figure 7.

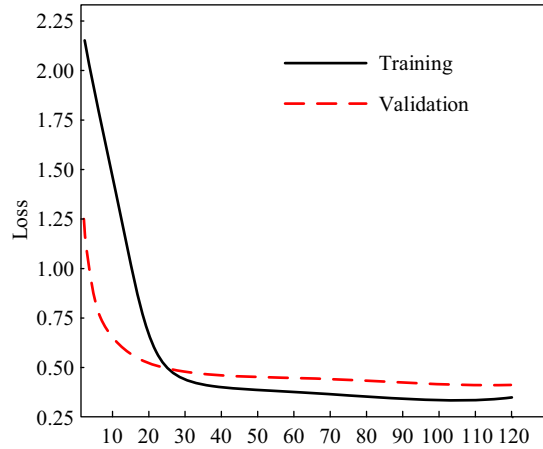


Figure 7: Training loss of the BTCBMA model.

The results in Figure 7 show that the proposed model gradually converges after training for 33 epochs and completely converges after training for 100 epochs.

The following is an analysis of the impact of convolution kernels. During the experiment, it is ensured that the other parameters are consistent except for the number of convolution kernels. Generally, when it is 2 to the n -th power, the GPU parallel computing space can be fully used. By changing it from 16 to 256, we can observe the most suitable number. Macro-F1 value of the observed results is shown in Figure 8.

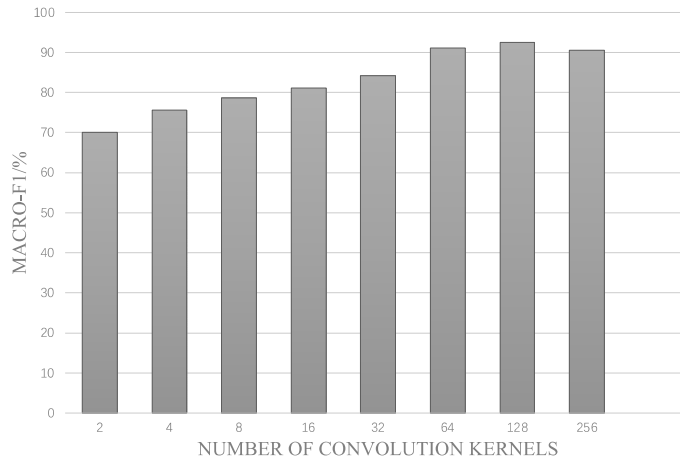


Figure 8: Variation of Macro-F1 under different convolution kernels.

In Figure 8, in the process of increasing the number of convolution cores from 2 to 128, Macro-F1 keeps rising and the rising rate gradually decreases. When the number of convolution cores further increases,

Macro-F1 starts to decline, i.e., when the number of convolution cores is 128, the value of Macro-F1 is the highest, and the model performs at its peak. Therefore, the number of convolution kernels of the model is set to 128 in the next experiment.

In the BTCBMA model, each Epoch will contain all the operations required for training. Theoretically, the more training rounds, the greater the A of the model. However, when the number of training rounds has reached a certain limit, the problem caused by our use of over-fitting will greatly reduce its A and may also seriously affect the computer’s operational efficiency. So, we hope to adjust and replace the number of Epoch to explore and verify whether we have used a reasonable number of training times. The experimental results of A with various number of Epochs are depicted in Figure 9.

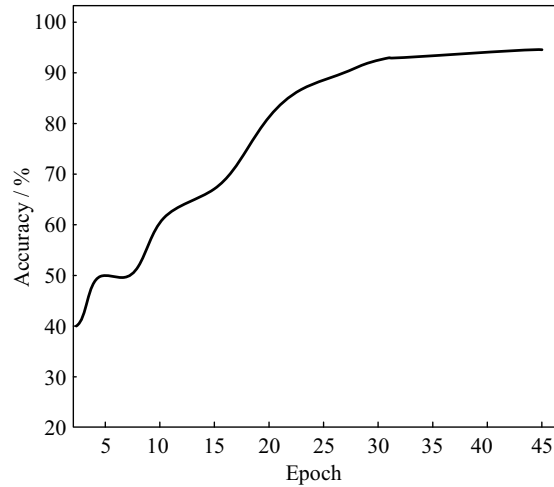


Figure 9: A at different number of Epoch.

In Figure 9, the A will decrease when Epoch increases from 5 to 10. However, after 10 rounds, the A has been rising rapidly. In the process of increasing Epoch from 25 to 30 times, the test A began to stabilize. When Epoch reaches 33 times, the A reaches the peak. It is thus verified that the Epoch value of the model is 33 times in the experiment.

In order to study the value of Dropout, comparative experiments were conducted by setting its value to different values to observe its effect on the model. The experimental results of Macro-F1 value variation with the value of Dropout are shown in Figure 10.

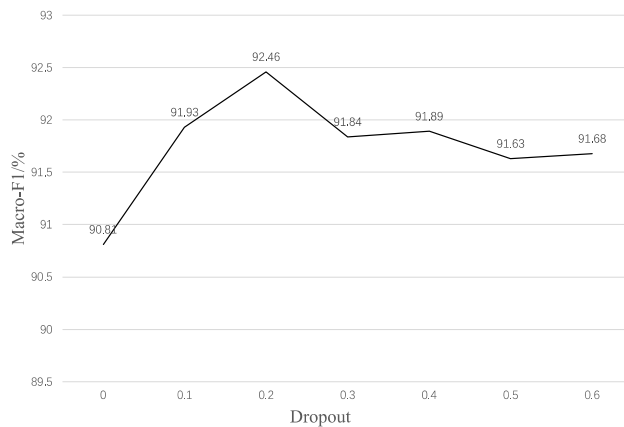


Figure 10: Trends in Macro-F1 values with different Dropout.

The Macro-F1 value exhibits an upward trend as the Dropout value increases from 0 to 0.2, as depicted in Figure 10. However, it begins to decline once the Dropout reaches 0.2. Therefore, 0.2 is chosen as the Dropout value for this experiment.

Based on hyperparameter experiments, in the suggested BTCBMA model, the number of convolutional kernels parameter is set to 128, Epoch parameter takes the value of 33, and the Dropout value is 0.2.

4.5 Comparative analysis

First, in the word vector coding layer of the model, the Word2Vec and BERT are used to encode the word vector input. The experimental results of using these two methods as pre-training models are depicted in Table 3.

Table 3: Comparison of experimental results with different pre-training models

Pre-training model	Model	A (%)	Macro-F1
Word2Vec	Word2Vec + BiLSTM	70.36	0.6983
	Word2Vec + TextCNN	74.22	0.7256
	Word2Vec + TextCNN + Attention	83.49	0.8224
BERT	BERT + BiLSTM	72.68	0.7135
	BERT + TextCNN	80.46	0.8021
	BTCBMA (ours)	93.45	0.9246

In Table 3, when Word2Vec is used as the word vector encoder, the model with the best sentiment classification effect is Word2Vec + TextCNN + Attention, its A is 83.49%, and Macro-F1 is 0.8224. When BERT is used as a word vector encoder, the best sentiment classification model is the BTCBMA model proposed. Its A is 93.45% and Macro-F1 is 0.9246, which is 9.96% and 0.1022 higher than Word2Vec + TextCNN + Attention, respectively. The superior capability of the BERT pre-training model to extract features from the input text is evident. Using BERT can not only obtain more comprehensive information about the meaning, between words and between sentences of each word, but also dynamically adjust the word vector context information to obtain a more comprehensive word vector representation.

To verify the effectiveness of microblog sentiment analysis using BTCBMA in Spark big data environment proposed in this study, two comparative experiments are designed. The methods were also compared and analyzed with those in the literature [28,29] and [30]. The sentiment classification A and Macro-F1 obtained using different methods under different datasets are shown in Table 4 and Figure 11.

Table 4: Sentiment classification results of different methods with different datasets

Dataset	Method	A (%)	Macro-F1
Self-built dataset	BTCBMA (ours)	93.45	0.9246
	Ref. [28]	88.46	0.8746
	Ref. [29]	82.31	0.8132
	Ref. [30]	84.65	0.8359
Benchmark dataset	BTCBMA (ours)	95.34	0.9435
	Ref. [28]	91.74	0.9046
	Ref. [29]	90.65	0.8925
	Ref. [30]	88.15	0.8776

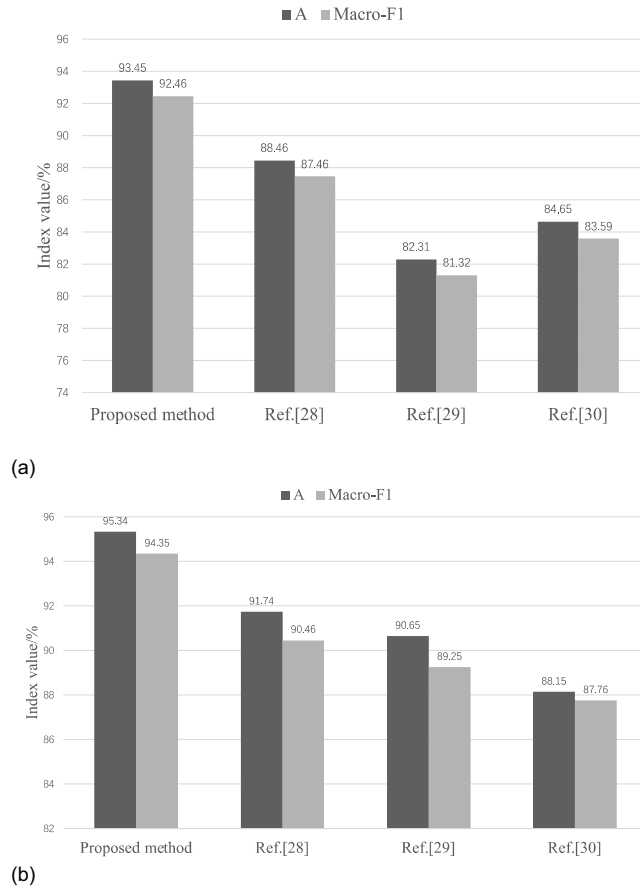


Figure 11: Sentiment classification results of different methods: (a) self-built dataset and (b) benchmark dataset.

In Figure 11, the bar graph represents A and the line graph represents Macro-F1 values. In both Table 4 and Figure 10, the proposed sentiment analysis method outperforms the other methods under different datasets. In the self-constructed dataset, A and Macro-F1 reach 93.45% and 0.9246, respectively, which is at least 4.99% and 0.05 improvement compared to other existing methods; in the benchmark dataset, A and Macro-F1 reach 95.34% and 0.9435, respectively, which is at least improved by at least 3.6% and 0.0389. BTCBMA adopted in this study integrates BERT, TextCNN, BiGRU, and M-HA. Compared with Refs [28–30] methods, it can more comprehensively extract semantic information from comment data. Using BERT, the inability of the comparison method to resolve the issue of word polysemy can be remedied. It is possible to extract local features in text data more effectively than CNN using the minimum granularity of TextCNN, which is determined by the words. Additionally, the issue of BiLSTM operating too slowly can be resolved by employing BiGRU. This not only facilitates the complete learning of contextual semantic connections in text data, but also enhances the model's execution speed. In conclusion, the M-HA algorithm was employed to allocate weights to distinct features, thereby emphasizing critical features while disregarding irrelevant feature data. This implementation successfully enhanced the precision of sentiment analysis in Weibo comments.

In order to further explain the importance and difference of each part of the model, the ablation experiment is carried out for the BTCBMA model proposed. The simulation analysis is carried out for BERT-TextCNN (BTC), BERT-TextCNN-BiGRU (BTCB), BERT-TextCNN-Multihead-Attention (BTCMA), and BTCBMA models, respectively. The ablation experiment results of the proposed BTCBMA are listed in Table 5.

The results presented in Table 5 demonstrate that the A and Macro-F1 values for a single BTC model are at their minimum, while the performance of the BTCMA and BTCB models surpasses that of BTC. These results suggest that the implementation of BiGRU or M-HA can significantly enhance the accuracy of sentiment classification and the overall performance of the model. The results indicate that the performance of the

Table 5: Ablation experiment results of BTCBMA

Model	A (%)	Macro-F1
BTC	80.46	0.8021
BTCB	85.28	0.8438
BTCMA	86.47	0.8592
BTCBMA (ours)	93.45	0.9246

BTCMA model is marginally superior to that of the BTCB model, suggesting that the enhancement introduced by the BiGRU is more substantial than that of the M-HA model. When both are simultaneously incorporated into the model, the *A* and Macro-F1 values of the BTCBMA model are at their maximum. This indicates that the model exhibits the highest level of efficacy in sentiment classification. It is capable of harnessing the benefits of both BiGRU and M-HA in order to significantly enhance performance.

5 Conclusions

In view of the inaccuracy of current sentiment analysis methods for microblog texts, which cannot classify sentiment accurately, a microblog sentiment analysis method based on the BTCBMA model in Spark big data environment is proposed. Compared with existing methods, it can more comprehensively extract semantic information from comment data. Using BERT, the inability of the comparison method to resolve the issue of word polysemy can be remedied. It can extract local features in text data better than CNN by using the minimum granularity of TextCNN based on words, and by using BiGRU, it is possible to circumvent the issue that BiLSTM operates too slowly. In addition to acquiring a comprehensive understanding of the contextual semantic relationships within textual data, it also enhanced the model's execution speed. In conclusion, the M-HA was employed to allocate weights to distinct features, thereby emphasizing critical features while disregarding irrelevant feature data. This implementation successfully enhanced the accuracy of sentiment analysis in Weibo comments.

There are still certain areas where the research presented in this study could be improved. For instance, the BERT pre-training model used in this study is excessively large and requires a performance enhancement. In addition, the sentiment classification in this study focuses on the three-classification task, and the classification granularity is coarse, belonging to the conventional “negative,” “positive,” and “neutral” sentiment classifications; however, the sentiment categories are rich, and there are two categories of sentiment labels under “positive” and “negative.” However, the emotion categories are rich and include more detailed and diverse emotions under the “positive” and “negative” emotion labels, such as “happy, excited, angry, sad.” Future work will focus on further investigating how the emotion labels of “negative,” “positive,” and “neutral” emotions can be combined with the emotion labels of “happy, excited, angry, sad.” Subsequent research will concentrate on expanding the suggested method to encompass multi-categorization and fine-grained classification challenges, with the aim of more accurately representing the sentiments and inclinations of Internet users during an abrupt or trending occurrence.

Funding information: This work was supported by the National Natural Science foundation of China (51565029) and the Young Innovative Talents Cultivation Fund Project of Kaifeng University (No. KDQN-2020-GK003).

Author contributions: Qian Wang: Conceptualization, Methodology, Formal analysis, Writing – Original Draft, Writing – Review & Editing; Delin Chen: Methodology, Writing – Original Draft.

Conflict of interest: The authors declare that there is no conflict of interest regarding the publication of this article.

Data availability statement: The data used to support the findings of this study are included within the article.

References

- [1] Kiefer C. Sample-level sound synthesis with recurrent neural networks and conceptors. *PeerJ Comput Sci.* 2019;5(7):205–13.
- [2] Tashiro D, Matsushima H, Izumi K, Sakaji H. Encoding of high-frequency order information and prediction of short-term stock price by deep learning. *Quant Financ.* 2019;19(9):1499–506.
- [3] Jang B, Kim I, Kim JW. Word2vec convolutional neural networks for classification of news articles and tweets. *PLoS One.* 2019;14(8):209–18.
- [4] Etaimi W, Suleiman D, Awajan A. Deep learning based techniques for sentiment analysis: A survey. *Informatica-J Comput Inform.* 2022;45(7):89–96.
- [5] Zhu L, Xu M, Bao Y, Xu Y, Kong X. Deep learning for aspect-based sentiment analysis: a review. *PeerJ Comput Sci.* 2022;8(12):67–75.
- [6] Liu HY, Chatterjee I, Zhou MC, Lu XS, Abusorrah A. Aspect-Based sentiment analysis: A Survey of deep learning Methods. *IEEE Trans Comput Soc Syst.* 2021;7(6):1358–75.
- [7] Demotte P, Senevirathne L, Karunanayake B, Munasinghe U, Ranathunga S. Sentiment analysis of Sinhala News Comments using Sentence-State LSTM Networks. 6th International Multidisciplinary Moratuwa Engineering Research Conference (MERCon). Moratuwa, SRI LANKA: Univ Moratuwa; 2020. p. 283–8.
- [8] Saleh H, Mostafa S, Alharbi A, El-Sappagh S, Alkhalifah T. Heterogeneous ensemble deep learning model for enhanced Arabic sentiment analysis. *Sensors.* 2022;22(10):28–37.
- [9] Ayyub K, Iqbal S, Nisar MW, Munir EU, Alarfaj FK, Almusallam N. A feature-based approach for sentiment quantification using machine learning. *Electronics.* 2022;11(6):305–15.
- [10] Seo S, Kim C, Kim H, Mo K, Kang P. Comparative study of deep learning-Based sentiment classification. *IEEE Access.* 2020;8(13):6861–75.
- [11] Wadawadagi R, Pagi V. Sentiment analysis with deep neural networks: comparative study and performance assessment. *Artif Intell Rev.* 2022;53(8):6155–95.
- [12] Dang NC, Moreno-Garcia MN, De la Prieta F. Sentiment analysis based on deep learning: a comparative study. *Electronics.* 2020;9(3):53–62.
- [13] Naqvi U, Majid A, Abbas SA. UTSA: Urdu Text sentiment analysis Using deep learning Methods. *IEEE Access.* 2021;9(5):114085–94.
- [14] Suriah GK, Hadiansah R, Murfi H, Ardaneswari G, Cahya LD, Nurhaniifah F. Analysis of Performance Long Short-Term Memory-Convolutional Neural Network (LSTM-CNN) for Lifelong Learning on Indonesian sentiment analysis. 3rd International Conference on Advanced Intelligent Systems for Sustainable Development (AI2SD). Tangier, Morocco; 2022. p. 990–1003.
- [15] Abdullaha EF, Alasadib SA, Al-Jodac AA. Text mining based sentiment analysis using a novel deep learning approach. *Int J Nonlinear Anal Appl.* 2021;12(5):595–604.
- [16] Chandrasekaran G, Antoanela N, Andrei G, Monica C, Hemanth JD. Visual sentiment analysis Using deep learning Models with Social Media Data. *Appl Sci-Basel.* 2022;12(3):72–81.
- [17] Wang S, Zhang H, Pan Y. Autoencoder with improved SPNs and its application in sentiment analysis for short texts. *J Harbin Eng Univ.* 2020;41(3):411–9.
- [18] Yadav A, Vishwakarma DK. A deep learning architecture of RA-DLNet for visual sentiment analysis. *Multimed Syst.* 2020;26(4):431–51.
- [19] Kumari S, Agarwal B, Mittal M. A deep neural network model for cross-domain sentiment analysis. *Int J Inf Syst Model Des.* 2021;12(2):1–16.
- [20] Liu X. POI Recommendation Model Using Multi-Head Attention in Location-Based Social Network Big Data. *Int. J. Inf. Technol. Syst. Approach.* 2023;16(2):1–16.
- [21] Lin S, Kung Y, Leu F. Predictive intelligence in harmful news identification by BERT-based ensemble learning model with text sentiment analysis. *Inf. Process. Manag.* 2022;59:102872.
- [22] Wu C, Zhang Y, Lu S, & Xu G. Short Text Sentiment Analysis Based on Multiple Attention Mechanisms and TextCNN-BiLSTM. 2023 IEEE 13th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China. 2020. p. 124–8.
- [23] Fan X. Artificial Intelligence Technology-Based Semantic Sentiment Analysis on Network Public Opinion Texts. *Int. J. Inf. Technol. Syst. Approach.* 2023;16(2):1–14.
- [24] Yuan J, Wu Y, Lu X, Zhao Y, Qin B, Liu T. Recent advances in deep learning based sentiment analysis. *Sci China Technol Sci.* 2020;63(10):1947–70.
- [25] Dashtipour K, Gogate M, Adeel A, Larijani H, Hussain A. Sentiment analysis of Persian Movie Reviews Using deep learning. *Entropy.* 2021;23(5):203–12.
- [26] Wang CT, Yang XX, Ding LK. Deep learning sentiment classification Based on Weak Tagging Information. *IEEE Access.* 2021;9(3):66509–18.

- [27] Balakrishnan A, Idicula SM, Jones J. Deep learning based analysis of sentiment dynamics in online cancer community forums: An experience. *Health Inform J.* 2021;27(2):45–53.
- [28] Mohammadi A, Shaverizade A. Ensemble deep learning for Aspect-based sentiment analysis. *Int J Nonlinear Anal Appl.* 2021;12(5):29–38.
- [29] Cheng QY, Ke Y, Abdelmouty A. Negative sentiment diffusion and intervention countermeasures of social networks based on deep learning. *J Intell Fuzzy Syst.* 2020;39(4):4935–45.
- [30] Elfaik H, Nfaoui E. Deep bidirectional LSTM Network learning-based sentiment analysis for Arabic text. *J Intell Syst.* 2021;30(1):395–412.