Retraction

Retracted: Network Public Opinion Risk Prediction and Judgment Based on Deep Learning: A Model of Text Sentiment Analysis

Computational Intelligence and Neuroscience

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] H. Yang, "Network Public Opinion Risk Prediction and Judgment Based on Deep Learning: A Model of Text Sentiment Analysis," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 1221745, 9 pages, 2022.

Research Article

Network Public Opinion Risk Prediction and Judgment Based on Deep Learning: A Model of Text Sentiment Analysis

Hairuo Yang [ID]

Wuhan University of Technology, Wuhan 430000, China

Correspondence should be addressed to Hairuo Yang; ysys0772@whut.edu.cn

Under the background of the gradual development and popularization of mobile Internet information technology, this paper realizes network public opinion monitoring and emotion analysis based on the deep learning method, aiming at the research needs of people's ideological changes and emotional trends. Aiming at the shortcomings of sentiment dictionaries or machine learning methods in sentiment analysis tasks, this paper builds a sentiment classification model based on deep learning methods. First, the current main text preprocessing methods are introduced, and then a sentiment classification model, BCBL, is proposed, combining BERT, CNN, and Bi LSTM. Compared with traditional models, BCBL can better complete text sentiment classification tasks on standard datasets. Next, in view of the problem that BCBL does not consider the distribution of vocabulary weights, an attention mechanism is introduced to improve BCBL, and then the BCBL-Att model is proposed. Set up multiple sets of comparative experiments again and find that the classification effect and overall performance of BCBL-Att on standard datasets are better than BCBL, indicating that BCBL-Att has more advantages in text sentiment classification tasks.

1. Introduction

The increasing popularity of the Internet and the number of netizens make society enter the information explosion era. Because the public opinion environment of the social networking platform is more open, free, and private than that of real society, and because public publishing and obtaining information in the network is more convenient and faster than exploring, publicizing, or querying information in the real world, more and more people pay attention to daily hot spots or public events through the Internet [1–5]. When people find the information they are interested in in the process of browsing the Internet, many people will express their own opinions and opinions on the content of the information, and the information containing the emotional tendencies and subjective attitudes of netizens constitutes a new network public opinion information [6–9]. By analyzing some network public opinion with a strong emotional attitude, we can find that there is not only a lot of positive public opinion information with healthy, positive, and positive energy but also some false, pessimistic, and reactionary negative opinion information.

Due to the lack of objective and reasonable judgment of the event, negative public opinion information is usually fragmented [10]. Such negative public opinion information is easy to cause network rumors, and the wanton dissemination of such negative public opinion information is the cause of many network violent events. At the same time, when people speak and comment on a real-time event on the Internet, the negative information will be more easily noticed by netizens. That is, the higher the number of views and clicks of the content, the greater the impact of negative emotions on the people [11–13]. If the public sentiment trend in the public opinion environment is not paid attention to and correctly guided in a timely manner, netizens may be eroded by these negative emotions and begin to complain about the current situation and even the current national conditions. Such a result will often lead to more serious network security problems. To sum up, it is very important to monitor and analyze public opinion information on the Internet. By understanding the changes in people's emotional trends towards popular events, the government can timely find and formulate corresponding

measures to protect the public opinion environment on the Internet, so as to maintain the harmony and stability of society and the country. In view of the actual need for online public opinion monitoring and emotion analysis under the current social background, this paper conducts the research.

2. Data Preprocessing

Remarks with subjective judgments and emotional tendencies made by netizens in response to certain public events or current affairs hotspots in real life are part of online public opinion information. To realize the monitoring and analysis of network public opinion, it is necessary to collect a large amount of public opinion data on the Internet to ensure the reliability of the results. However, these raw public opinion data stored locally in the form of text usually contain a lot of meaningless special symbols, links, expressions, and illegal characters. Before performing text sentiment analysis, data preprocessing on the original text is required. Common data preprocessing methods in NLP tasks include data cleaning, stop word removal, and Chinese word segmentation [14–16]. These three preprocessing methods are introduced below.

The main purpose of data cleaning is to transform the original data that is messy and full of meaningless characters, into concise and clear data that is conducive to subsequent text feature extraction. The main contents include removing duplicate data, removing unrecognized special symbols in the text, and removing unrecognized expressions. Deduplication can help the model maximize its training speed while reducing the overall data volume. The purpose of removing special symbols and unrecognizable expressions is to filter out the useless information in the text to make the text more streamlined as a whole, thereby reducing the difficulty of sentiment classification for subsequent text input models. The experimental part of this paper mainly uses Python language-related functions and methods to clean the original data.

Removing stop words is a common method for the input preprocessing part of NLP tasks. Stop words refer to words that appear more frequently in the text and make the text speak fluently but without great significance. Taking Chinese documents as an example, words such as "then," "for example," "in," and "next" in Chinese texts are widely used and frequently in the text but have little practical significance for sentiment analysis. The method of stop-words improves the extraction and analysis efficiency of text features. When removing stop words, it is usually necessary to prepare a stop-word table. When any word in the table exists in the cleaned text, the word will be automatically defined as a stop word and filtered out. For Chinese NLP tasks, there are already some standard stop-word lists in China. The experimental part of this paper uses Baidu stop-use-word lists to process the text to realize the function of removing stop-use words.

Chinese word segmentation is a key step in NLP tasks, and word segmentation in data preprocessing is very important for public opinion sentiment monitoring and analysis. The lack of word segmentation accuracy will directly affect the quality of the dataset and even lead to the wrong use of the dataset. There is a big difference between Chinese and English word segmentation. English generally uses spaces or some specific symbols as decomposers to distinguish each word, while Chinese text not only distinguishes words through spaces but also appears with other symbols such as commas, ellipses, quotation marks, and many other symbols used to express the meaning of the text. Moreover, there are usually multiple meanings for characters in Chinese texts. A word containing a certain character may have completely different meanings when the character is paired with a word composed of another character. Chinese word segmentation is very important for Chinese NLP tasks. For some specific NLP tasks, the choice of word segmentation tools will directly affect the overall performance of the model. Therefore, many research institutions, experts, and deep learning laboratories of high-tech companies have been working on developing Chinese word segmentation tools with higher accuracy and more complete functions. At present, the more mature Chinese word segmentation tools or systems on the market include Alibaba Cloud NLP, THULAC, HanLP, Jieba word segmentation, etc.

3. BCBL Sentiment Classification Model

Online public opinion information generally appears on online social platforms in the form of text, and the essence of sentiment analysis of online public opinion is reflected in the study of public opinion texts [17–19]. The text sentiment classification task usually consists of two parts: text vectorized representation and sentiment classification. Through the Transformer structure and pretraining, the BERT model can convert large-scale network public opinion text sequences into global feature word vectors containing contextual semantics for downstream network input. But for the downstream classification part, compared with deep learning models, traditional text classification models related to machine learning such as clustering, SVM, and decision trees are more complicated to extract text features from, their generalization ability and classification effect are poor, and they are unable to adapt to the increasingly rapid vocabulary iteration status and the increasing volume of data in the era of big data. Therefore, the sentiment classification model based on deep learning gradually replaces the traditional model and becomes the main way of text sentiment classification in the NLP field.

In the field of deep learning, convolutional neural networks and bidirectional long-short term memory are the current research hotspots for NLP text sentiment classification tasks. CNN can effectively extract the local features of the input text, and because the CNN network structure is relatively simple and has few parameters, it has a fast training speed. However, CNN also has obvious shortcomings. First, the CNN result contains pooling, through which the word vector will lose many relatively important features and sentence order, making the spatiality of the features disappear. Second, CNNs lack consideration of textual contextual semantic information, which is contrary to what
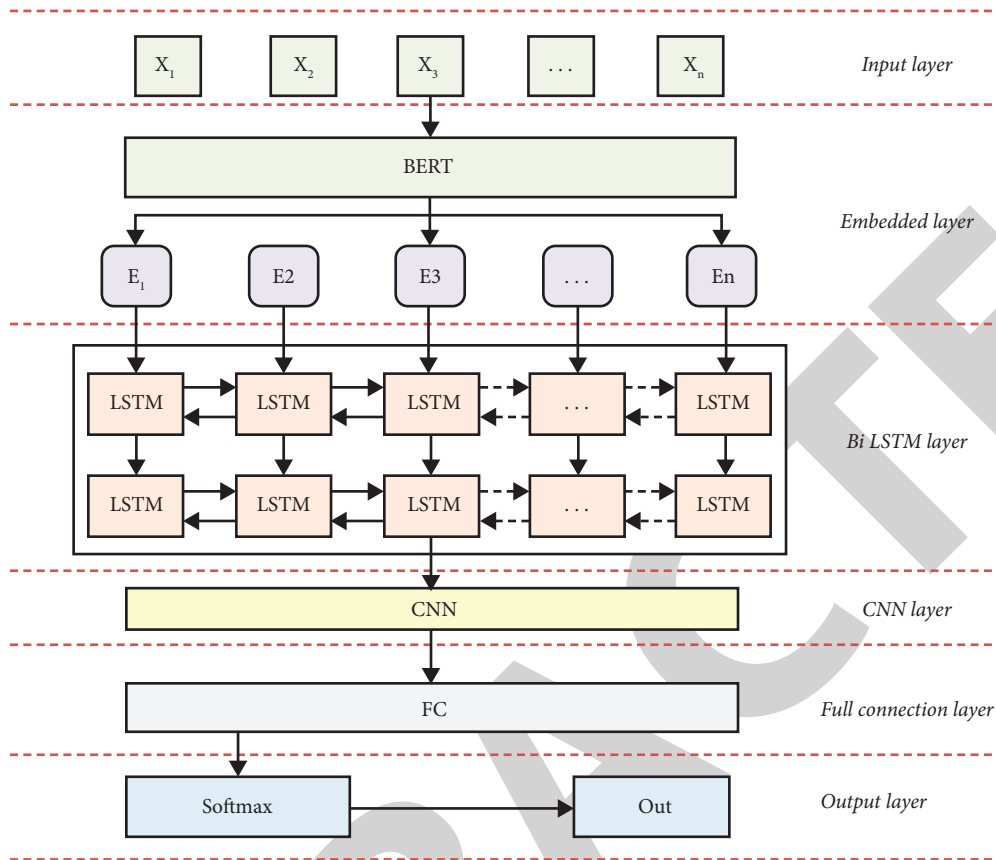
Figure 1: Overall structure of the BCBL model.

people actually consider for text sentiment classification tasks. Compared with CNN, Bi-LSTM fully considers the characteristics of bidirectional semantic information so that it can better extract the global features of the text. However, Bi-LSTM is less effective for extracting local features, and there are many parameters in Bi-LSTM, and the training time is long.

To sum up, in order to better solve the problem of online public opinion sentiment classification, this paper proposes a BCBL model that combines BERT, CNN, and Bi-LSTM technologies [20–24].

The design idea of the BCBL model is to first use BERT to complete the word vector conversion of the input text, then input the text word vector results output by BERT into the Bi-LSTM and CNN fusion models, and pass a fully connected layer after the context feature information and local features are obtained. The features extracted by the two models are spliced and integrated to obtain a global feature vector that completely contains the essence of the vocabulary itself and contextual semantic information. Finally, sentiment classification of the final text is done by a SoftMax classifier.

The overall structure of the BCBL model is shown in Figure 1.

As shown in Figure 1, the BCBL upstream BERT structure will continue to pay attention to the context during the transformation of the input text word vector and help the model to obtain the word vector representation with multilevel features. The downstream Bi-LSTM and CNN

structures focus on feature extraction of text word vectors. It should be noted that microblog posts are generally in the form of short text, and the characteristic of short text information is that the features are concentrated in a certain part [25, 26]. While considering the impact of contextual semantics, extracting local features more effectively will allow the model to learn more complete blog postfeatures. It can be seen that the combination of CNN and Bi-LSTM can help the model better complete the task of sentiment classification. In summary, the overall process of text sentiment classification by the BCBL model is shown in Figure 2.

4. Experiments and Analysis

In order to verify the feasibility of the sentiment classification model BCBL constructed above, on the basis of selecting the same dataset and objective evaluation indicators, three groups of parameters were set up for comparison experiments to select the optimal parameters of BCBL. Then, the classification effect of BCBL and the overall performance of the model are analyzed by designing a comparative experiment between BCBL and several different sentiment classification models.

4.1. Experimental Setup. Before officially starting the experiment, we first explain the experimental environment, dataset, and evaluation indicators for this experiment.
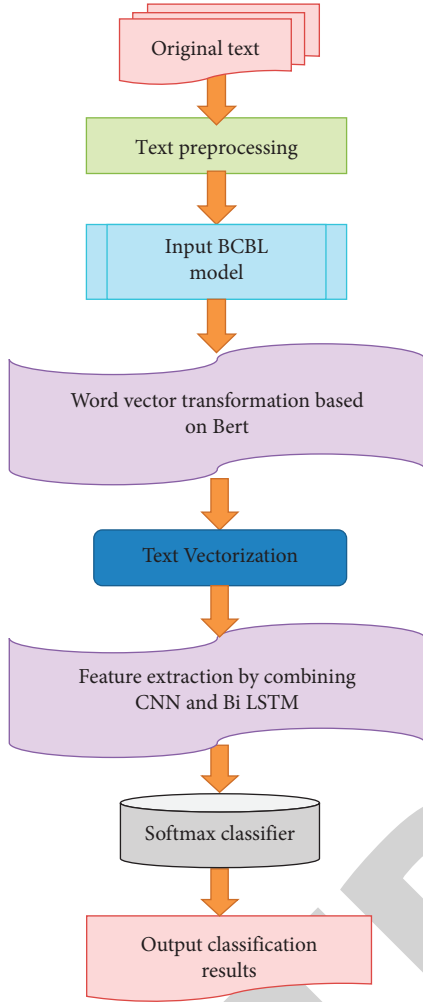
TABLE 2: Statistics of data distribution.

Data distribution	Train	Valid	Test
Positive emotion	16455	2038	2022
Negative emotion	15620	1971	1966
Total amount	31859	3972	3958

FIGURE 2: Text sentiment classification by the BCBL model.

TABLE 1: Experimental environment configuration table.

Experimental environment	Specific configuration
PC-side operating system	Window10 64 bit
Processor	Intel core i7-8750H
GPU	Nvidia tesla T4
CPU memory	16 G
Compile the software	Google colab
Editing language	Python3
Deep learning framework	Keras

4.1.1. Experimental Environment. The experimental part of the environment configuration in this paper is shown in Table 1.

4.1.2. Data Set Selection. In order to ensure the objectivity of the results, the source of the dataset for the sentiment analysis task needs to be a representative website or social platform. Sina Weibo is one of the most popular online social platforms in China. It has a high number of daily active users, and the feature of real-time updates also allows the Weibo platform to contain a wealth of online public opinion information. After analysis, the Simplifyweibo_4_moods dataset was selected as

the standard dataset in the experimental part of this chapter. The following introduces the basic situation of the dataset and performs some processing on the dataset according to the task requirements.

The Simplifyweibo_4_moods dataset is a standard dataset based on the Weibo platform, which contains more than 380,000 Weibo data points with 4 types of sentiment labels. Among them, about 200,000 pieces of data are marked with joy emotional labels, and about 100,000 pieces of data are marked with three emotional labels: disgust, anger, and depression. Since disgust, anger, and depression are all bad emotional experiences, the descriptions of some words and sentences are relatively similar, and the model learning is difficult [27–29]. Therefore, according to the different emotional tendencies of the data in the dataset, the label items of the entire dataset are simplified and divided into positive and negative sentiments for reclassification. The specific processing method is to first integrate the data marked with disgust, anger, and low emotional labels into the dataset, and then remove or reprocess the data with a certain ambiguity and mark these negative emotional data as negative emotional data. Similarly, ambiguity is also performed on the data marked as joy emotional labels in the dataset, and these data containing positive emotions are marked as positive emotional data. Among them, positive sentiment data is marked as 1, and negative sentiment data is marked as 0.

After the dataset is relabeled and classified, part of the data in the standard dataset is randomly selected for model training. The dataset distribution statistics in this experiment are shown in Table 2.

4.1.3. Evaluation Indicators. An objective and reasonable model evaluation method is the key to judging the performance of each deep learning model in the text sentiment classification problem. Therefore, four commonly used model evaluation indicators are selected: accuracy, precision, recall, and F1 value as the evaluation criteria for this experiment. Before introducing each evaluation index, we first define some representation methods:

When the sample instance is a positive emotion, the predicted result is also a positive emotion, and this situation is set as the real class, which is represented by TP; when the sample example is a negative emotion, the predicted result is also a negative emotion, and this situation is set as a true negative class, which is represented by TN; when the sample instance is a positive emotion, the predicted result is a negative emotion, and this situation is set as a false negative class, which is represented by FN; and when the sample instance is a negative emotion, the predicted result is a positive emotion, and this case is set as a false positive class,

Table 3: Classification representation relationship description table.

Sample instance	Positive prediction	Negative prediction
Positive result	TP	FN
Negative result	FP	TN

represented by FP. The descriptions of the four classification representation relationships are shown in Table 3.

The accuracy rate A refers to the ratio of the number of correctly predicted data to the total data. The calculation method of the accuracy rate A is as follows:

$$A = \frac{TP + TN}{TP + FP + TN + FN}. \tag{1}$$

The accuracy rate P refers to the ratio of the number of data items that are predicted to be positive and that are correctly predicted to the total number of data items that are predicted to be positive. The calculation method of the accuracy rate P is as follows:

$$P = \frac{TP}{TP + FP}. \tag{2}$$

The recall rate R refers to the ratio of the number of data items that are predicted to be positive and correctly predicted to the number of data items that are actually positive. The recall rate R is calculated as follows:

$$R = \frac{TP}{TP + FN}. \tag{3}$$

The $F1$ value is a public evaluation index that combines precision and recall. The calculation method of the $F1$ value is as follows:

$$F1 = 2 \times \frac{PR}{P + R} = \frac{2TP}{2TP + FP + FN}. \tag{4}$$

4.2. BERT-Based Text Vectorization. The raw text data in the standard dataset is crawled and stored directly from the Weibo platform. Although sentiment labels are manually annotated in the dataset, the original Weibo text content is not processed [30–32]. In order to view the original data in the standard dataset, an example of randomly selected data is selected.

There are still meaningless special symbols and illegal characters in the original data, and there may be some duplicate data in other parts of the dataset that are not shown. Therefore, text preprocessing is required before input into the model. The preprocessing method is as described above. Data cleaning and the removal of stop words are used as the first stage of preprocessing in this experiment, and word segmentation is used as the second stage of preprocessing in this experiment.

The data cleaning part mainly uses the regular expressions of the remodule in Python to filter out illegal characters, punctuation, invalid expressions, and links in the dataset. The standard dataset is deduplicated by the deduplication function drop_duplicates () in Python. In the part

of removing stop words, the Baidu stop word table is used to remove stop words from the cleaned data, which makes the text more compact when retaining meaningful words.

The second stage of processing is word segmentation. Chinese vocabulary has the characteristics of multimodality and multiword meaning. Segmenting Chinese text by word form may lose the essential features of the sentence, resulting in ambiguity. At the same time, since the bidirectional Transformer structure in BERT can take into account the contextual semantic information of each word, therefore, this experiment adopts the unigram segmentation method; that is, the text data is divided into words, and the preprocessed text sequence will enter the BCBL upstream BERT structure in the form of word vectors.

After a series of preprocessing of the input text, the BERT structure is responsible for the vectorized representation of the input text sequence. Google mainly provides two pretrained models, BERTBASE and BERTLARGE, and for this experiment, we use the BERTBASE model. In addition, BERT officially provides preset parameters, Chinese dictionaries, and a series of configuration files for Chinese word vector training tasks. For the text results that have been preprocessed, the results are first formed into a dictionary, and the words are converted into corresponding IDs through the vocabulary loaded by Token. After the ID sequence is input to BERT, a high-dimensional space word vector corresponding to the input text is generated.

It should be noted that BERT is a multilayer bidirectional Transformer structure, each layer of the Transformer structure has a corresponding output, and the text vectorization representation part uses the output result of the last layer of the Transformer as the final result of BERT vectorization. Since the BERTBASE hidden layer dimension is fixed at 768, each vector output is a word vector of the format (1,768). After the word vector is obtained, the word vector representation of the text in the same format (1,768) can be obtained by adding the word vectors. At the same time, each piece of text data consists of one or more sentences, and each sentence consists of multiple words. Assume that the sentence contains X words at this time; that is, each sentence can be represented as a sentence vector with the format (X, 768) by splicing word vectors, and the vectorized representation of the entire input text can be obtained by obtaining all the sentence vectors.

4.3. BCBL Model Parameter Selection. In order to verify the performance of the BCBL model proposed in this paper in sentiment analysis tasks and strive to achieve better text sentiment classification results, it is necessary to select appropriate model parameters through multiple experiments. After the comparison of the parameter adjustment experiments, it is found that the parameters that have a greater impact on the model performance are the activation function, the number of LSTM hidden layer neurons (LSTM Hidden Size), and the number of model iterations (epoch). Therefore, these three key parameters are being investigated.

First, the number of model iterations is preset to 15, the number of LSTM hidden layer units is preset to 80, and

Table 4: Classification accuracy results after each model selects different activation functions.

Model	Positive data			Negative data		
	Sigmoid	Relu	Tanh	Sigmoid	Relu	Tanh
BERT-CNN	0.7133	0.7053	0.7002	0.7205	0.7272	0.7232
BERT-Bi LSTM	0.6947	0.7025	0.6887	0.7212	0.7269	0.7215
BCBL	0.7176	0.7015	0.7041	0.7323	0.7234	0.7263

Table 5: Influence of LSTM hidden size on the classification accuracy of the BCBL model.

Size	Average spent time (s/epoch)	Accuracy
128	730	0.7281
64	662	0.7296
32	577	0.7230
16	502	0.6974

Sigmoid, Relu, and Tanh are selected as the activation functions of the model. Then, the standard CNN network and the LSTM network are connected as the downstream network of BERT. We built a BERT-CNN model combining BERT and CNN, and a BERT-Bi LSTM model combining BERT and Bi-LSTM as a comparison model for the BCBL model to explore the impact of different activation functions on the classification effects of the three models under the standard dataset. The results are shown in Table 4.

From the experimental results in Table 4, we can see that the classification accuracy obtained by the same deep learning model using different activation functions or different deep learning models using the same activation function is different. Compared with other models, BCBL has a higher classification accuracy when using the Sigmoid activation function and the Tanh activation function to classify the forward data of the standard dataset. When using the Relu activation function to classify forward data, the classification accuracy of BERT-CNN is higher than other models. Compared with other models, BCBL has higher classification accuracy when using the Sigmoid activation function and Tanh activation function to classify negative data in standard datasets. While using the Relu activation function to classify negative data, the BERT-Bi-LSTM classification accuracy is high. As for the accuracy results, when BCBL uses the Sigmoid function as its activation function, the classification accuracy of positive data in the standard dataset is 0.7176, and the classification accuracy of negative data is 0.7323. These two results have the highest classification accuracy among different classes of data, respectively. After analysis, compared with the other two activation functions, the reason for the higher accuracy of using the Sigmoid function is that the output range of the Sigmoid is between (0, 1), the optimization is stable, and it is very suitable for application in the text sentiment output task of binary classification. Therefore, Sigmoid is chosen as the activation function of BCBL.

After determining the appropriate activation function of the BCBL model, continue to explore the impact of the number of LSTM hidden layer units on the performance of BCBL sentiment classification. The larger the number of hidden sizes selected, the greater the complexity of the model and the larger the overall calculation, which is likely to cause the model to overfit. However, if the number of hidden variables is too small, it cannot reflect the optimal performance of the model. For this purpose, the Sigmoid function was selected as the activation function of the BCBL model. On the basis of the same other parameters, we

set the LSTM Hidden Size of the BCBL model to 16, 32, 64, and 128 to conduct four sets of control experiments, and the classification accuracy results obtained are shown in Table 5.

The experimental results in Table 5 show that when the LSTM hidden size is set to 64, the classification accuracy of the BCBL model under the standard dataset is the highest. The optimal classification accuracy was not achieved when the hidden size was set to 32 or 128. The results show that a reasonable setting of the hidden size can improve the classification effect of the model to a certain extent. At the same time, when the number of hidden units is 128, the average consumption time of each epoch of the model is 662 s, and when the number of hidden units is 32, the average consumption time of each epoch is 577 s. Setting the number of hidden units to 128 takes longer than setting it to 64. According to the needs of sentiment analysis tasks, more attention should be paid to the influence of parameters on model accuracy and overall performance when selecting sentiment classification model parameters than model training time and resource occupancy. Therefore, the number of hidden layers of the BCBL model is chosen to be 64.

Finally, it is necessary to select an appropriate number of iterations for the BCBL model. The choice of the number of iterations is very important. A lower number of iterations will result in incomplete model training, and the evaluation index results obtained from incomplete training cannot reflect the true performance of the model. A higher number of iterations can cause the model to take too long to train, consuming too many computer resources. By judging the size of the dataset and the experimental environment, we select an epoch in the range of 0 to 15, set the activation function as the Sigmoid function, and set the LSTM hidden size to 64. Other parameters are consistent with the previous experiments. The accuracy of the BCBL model after each epoch is recorded in Figure 3 and the loss is shown in Figure 4.

As can be seen from Figure 3, when the number of iterations reaches 8 and above, the model accuracy gradually stabilizes. When it reaches 10, the classification accuracy of the BCBL model is the highest, and the accuracy of the model decreases as the number of iterations continues to increase. As can be seen from Figure 4, when the number of iterations is 10, the loss rate of the model reaches its lowest level, and as the number of iterations continues to increase, the loss rate of the model increases. The goal of setting the epoch parameter is to expect the model to have high accuracy and a low loss rate in the classification task.
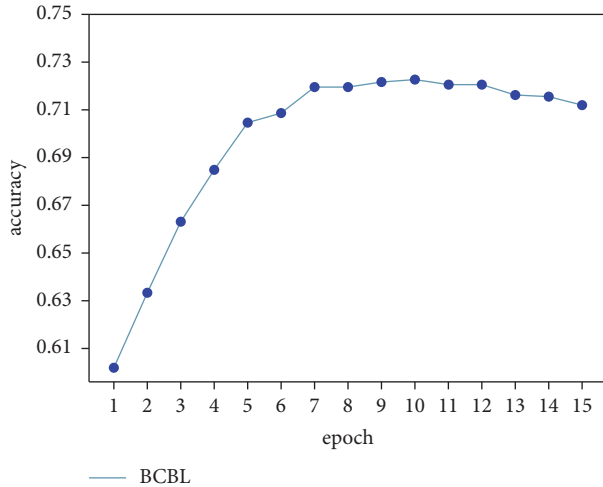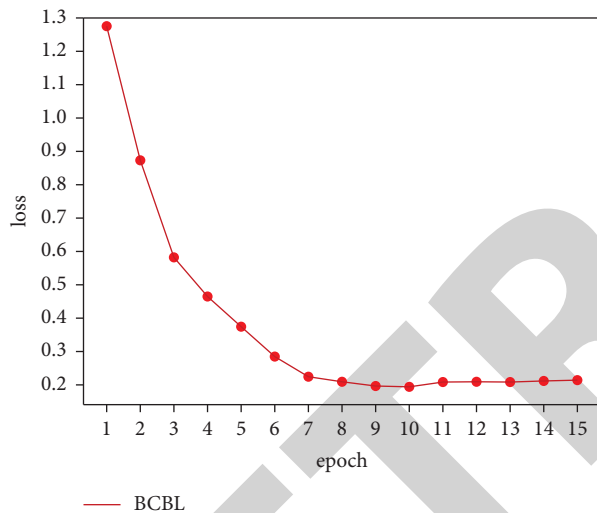
Figure 3: Accuracy of the BCBL model.

Figure 4: Loss of the BCBL model.

Combining the two results, the overall performance of the model is the best when the epoch is 10. Therefore, we set the number of iterations of the BCBL model to 10.

4.4. Analysis and Discussions. After selecting the appropriate parameters of BCBL through multiple comparative experiments, this paper selects the activation function of the BCBL model as Sigmoid, hidden size as 64, epoch as 10, builds a BERT-CNN model combining BERT and CNN, a BERT-Bi-LSTM model combining BERT and Bi-LSTM as two comparison models, and carries out the overall performance comparison experiment of the model. The settings of model parameters are shown in Table 6, and the comparison results of the classification predictions of different classification models under the standard dataset are shown in Table 7.

As shown in Table 7, comparing the BERT-CNN and BERT-Bi-LSTM models, BCBL has higher numerical values in the three evaluation indicators of precision, recall, and *F*1 value. Therefore, it can be seen that the performance of

Table 6: Model parameters.

Parameter	BERT-CNN	BERT-Bi-LSTM	BCBL
Word vector dimension	768	768	768
Convolutional kernel size	[3–5]		[3, 4]
Number of filters	256		256
LSTM hidden size		64	64
Dropout	0.5	0.5	0.5
Epoch	10	10	10
Batch	24	24	24

Table 7: Classification prediction results of different classification models under standard datasets.

Parameter	Precision	Recall	F1
BERT-CNN	0.7156	0.6811	0.7021
BERT-Bi LSTM	0.7121	0.6675	0.6915
BCBL	0.7293	0.6962	0.7186

BCBL in the standard dataset sentiment classification task is better than that of the traditional neural network model. The experimental results also further verified the feasibility of the BCBL model constructed in this paper.

5. Conclusions

With the progress and development of Internet technology, more and more people regard the Internet as the primary medium for them to understand and discuss public events, and the influence of network public opinion on social public opinion is gradually increasing. However, because different netizens view events from different angles, their public opinion information often has obvious subjective emotional tendencies. When a large amount of subjective and one-sided public opinion information is spread in the network, some too radical and negative emotional information will easily affect the emotions and mentality of other netizens. In the long run, it will damage the network's public opinion environment. Therefore, the monitoring of network public opinion and reliable emotional analysis play an important role in maintaining the network environment and promoting social stability. Based on the deep learning method, this paper proposes a text sentiment classification model, BCBL, that combines BERT, CNN, and Bi-LSTM. Then, as BCBL does not consider the distribution of lexical sentiment weights, we introduce an attention mechanism to improve it, build a BCBL-Att model, and verify the effectiveness of the model on the preprocessed standard dataset.

The BCBL model proposed in this paper mainly uses the structure of the pretraining model Bert for text vectorization. Although the Bert base model with fewer layers is used in the experiment, the model still contains a large number of parameters, and more training time will be spent on parameter adjustment and gradient synchronization in the model training process. Aiming at the problem that the parameters of the Bert model are too large, the improved model Albert, recently launched by Google, was found through searching and analyzing relevant data. The Albert model is equivalent to a slimmed-down version of the Bert

model. Compared with Bert, although Albert has fewer parameters and faster model training speed, it performs even better than Bert in some NLP tasks. In the future, we can try to study the structure of the Albert model to improve the emotion classification model constructed in this paper.

In addition, the main structure of the model constructed in this paper is that the upstream Bert word vector conversion part is connected with the downstream Bi-LSTM and CNN feature extraction parts. The fusion structure of Bi-LSTM and CNN of the model can be further optimized. In the experimental part of this paper, the traditional CNN structure or Bi-LSTM structure under Bert is taken as two comparison models. Subsequently, the number of comparison models can be increased by changing the network structure of the model or introducing other deep learning models to obtain more experimental comparison results for the analysis and evaluation of the overall performance of the BCBL-Att model.

Data Availability

The data used to support the findings of this study are available upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] L. Li, Y. Wan, D. Plewczynski, and M. Zhi, "Simulation Model on Network Public Opinion Communication Model of Major Public Health Emergency and Management System Design," *Scientific Programming*, vol. 2022, Article ID 5902445, 2022.

[2] S. Yang, "Analysis of Network Public Opinion in New Media Based on BP Neural Network Algorithm," *Mobile Information Systems*, vol. 2022, 2022.

[3] M. Li and A. Li, "A Network Public Opinion Trend Estimation Model Using a Scale-free Network Algorithm," *Mobile Information Systems*, vol. 2022, 2022.

[4] Y. Zhang, "Analysis of Dynamic Influence Mechanism of Network Public Opinion Based on Simulation Feature Extraction," *Security and Communication Networks*, vol. 2022, 2022.

[5] He Liu, Z. Yu, X. Zhong, and H. Yu, "Network Public Opinion Monitoring System for Agriculture Products Based on Big Data," *Scientific Programming*, vol. 2021, 2021.

[6] W. Wang, "Textual Information Classification of Campus Network Public Opinion Based on BILSTM and ARIMA," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 8323083, 2022.

[7] G. Dong, W. Zhang, H. Tan, R. Yadav, and S. Tan, "Eom-Nposess Emergency Ontology Model Based on Network Public Opinion Spread Elements," *Security and Communication Networks*, vol. 2021, 2021.

[8] Y. Zhu, "Interoperability of multimedia network public opinion knowledge base group based on multisource text mining," *Advances in Multimedia*, pp. 2021–9, 2021.

[9] X. Liu, A. Cao, and C. Li, "Novel network public opinion prediction and guidance model based on "s-Curve": taking the loss of contact with Malaysia airlines," *Mathematical Problems in Engineering*, pp. 2021–13, 2021.

[10] Y. Zhu, "Network public opinion prediction and control based on edge computing and artificial intelligence new paradigm," *Wireless Communications and Mobile Computing*, vol. 2021, 11 pages, 2021.

[11] Y. Tong and W. Sun, "Multimedia Network Public Opinion Supervision Prediction Algorithm Based on Big Data," *Complexity*, vol. 2020, Article ID 6623108, p. 2020.

[12] S. Zhang, Z. Ma, X. Liu, Z. Wang, and L. Jiang, "An integrated consensus improving strategy based on PL-Wasserstein distance and its application in the evaluation of network public opinion emergencies," *Complexity*, pp. 1–24, 2020.

[13] S. Xu, J. Liu, K. Chen, and Y. Yang, "Research on the communication path of public opinion in university ideological and political network for big data analysis," *Journal of Sensors*, pp. 1–9, 2022.

[14] F. Wan, Y. Yang, D. Zhu et al., "Semantic role labeling integrated with multilevel linguistic cues and Bi-LSTM-CRF," *Mathematical Problems in Engineering*, pp. 1–8, 2022.

[15] Y. Li and L. H. Li, "Enhancing the optimization of the selection of a product service system scheme: a digital twin-driven framework," *STROJNISKI VESTNIK-JOURNAL OF MECHANICAL ENGINEERING*, vol. 66, no. 9, pp. 534–543, 2020.

[16] H. Sun, J. Sun, K. Zhao, L. Wang, and K. Wang, "Data-Driven ICA-Bi-LSTM-Combined Lithium Battery SOH Estimation," *Advanced Dynamics, Control and Management in Renewable Energy and Storage Systems*, vol. 2022, Article ID 9645892.

[17] Source, "Mathematical Problems in Engineering," vol. 2022, 2022.

[18] B. Liu and Q. Zhao, "Implementation of Financial Audited Robot Question and Answer Technology of Feature Processing and Improved Bi-LSTM," *Scientific Programming*, vol. 2022, Article ID 7213395, 2022.

[19] L. h. Li, J. c. Hang, Y. Gao, and C. y. Mu, "Using an integrated group decision method based on SVM, TFN-RS-AHP, and TOPSIS-CD for cloud service supplier selection," *Mathematical Problems in Engineering*, vol. 2017, pp. 1–14, 2017.

[20] S. Zhao, Q. Pan, Q. Zou, Y. Ju, L. Shi, and Xi Su, "Identifying and classifying enhancers by dinucleotide-based auto-cross covariance and attention-based Bi-LSTM," *Computational and Mathematical Methods in Medicine*, pp. 1–11, 2022.

[21] D. Zhou, Y. Wu, and H. Zhou, "A Nonintrusive Load Monitoring Method for Microgrid EMS Using Bi-LSTM Algorithm," *Complexity*, vol. 2021, 2021.

[22] L. Li, J. Hang, H. Sun, and L. Wang, "A conjunctive multiple-criteria decision-making approach for cloud service supplier selection of manufacturing enterprise," *Advances in Mechanical Engineering*, vol. 9, no. 3, Article ID 168781401668626, 2017.

[23] K. Zeberga, M. Attique, B. Shah, F. Ali, Y. Z. Jembre, and T.-S. Chung, "A Novel Text Mining Approach for Mental Health Prediction Using Bi-LSTM and BERT Model," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 7893775, 2022.

[24] G. Xu, D. Zhou, and J. Liu, "Social Network Spam Detection Based on ALBERT and Combination of Bi-LSTM with Self-Attention," *Security and Communication Networks*, vol. 2021, Article ID 5567991, 2021.

[25] L. Li, T. Qu, Y. Liu et al., "Sustainability assessment of intelligent manufacturing supported by digital twin," *IEEE Access*, vol. 8, pp. 174988–175008, 2020.

[26] L. Li and C. Mao, "Big data supported PSS evaluation decision in service-oriented manufacturing," *IEEE Access*, vol. 8, no. 99, pp. 154663–154670, 2020.

[27] G. Kumaravel and S. Sankaranarayanan, "PQPS: Prior-Art Query-Based Patent Summarizer Using RBM and Bi-LSTM," *Mobile Information Systems*, vol. 2021, Article ID 2497770, 2021.

[28] Z. Yang, C. Yang, X. Li, and C. Min, "Pattern Recognition of the Vertical Hydraulic Fracture Shapes in Coalbed Methane Reservoirs Based on Hierarchical Bi-LSTM Network," *Complexity*, vol. 2020, Article ID 1734048, 2020.

[29] L. Li, B. Lei, and C. Mao, "Digital twin in smart manufacturing," *Journal of Industrial Information Integration*, vol. 26, no. 9, Article ID 100289, 2022.

[30] L. Li, C. Mao, H. Sun, Y. Yuan, and B. Lei, "Digital twin driven green performance evaluation methodology of intelligent manufacturing: hybrid model based on fuzzy rough-sets AHP, multistage weight synthesis, and PROMETHEE II," *Complexity*, vol. 2020, no. 6, pp. 1–24, 2020.

[31] H. Zhao, C. Hou, H. Alrobassy, and X. Zeng, "Recognition of transportation state by smartphone sensors using deep Bi-LSTM neural network," *Journal of Computer Networks and Communications*, pp. 1–11, 2019.

[32] C.-Z. Xiong and M. Su, "IARNN-based Semantic-Containing Double-Level Embedding Bi-LSTM for Question-And-Answer Matching," *Computational Intelligence and Neuroscience*, Article ID 6074840, p. 2019, 2019.