Sentiment analysis for e-commerce product reviews in Chinese based on sentiment lexicon and deep learning

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 26, 2019, accepted January 15, 2020, date of publication January 27, 2020, date of current version February 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2969854

Sentiment Analysis for E-Commerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning

LI YANG[1], (Member, IEEE), YING LI[1], JIN WANG[1,2], (Senior Member, IEEE), AND R. SIMON SHERRATT[3], (Fellow, IEEE)

[1] Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation, School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410004, China
[2] School of Information Science and Engineering, Fujian University of Technology, Fuzhou 350118, China
[3] School of Systems Engineering, University of Reading, Reading RG6 6AY, U.K.

Corresponding author: Jin Wang (jinwang@csust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61602060, Grant 61772454, Grant 61811530332, and Grant 61811540410, and in part by the Open Research Fund of Hunan Provincial Key Laboratory of Intelligent Processing of Big Data on Transportation under Grant 2015TP1005.

ABSTRACT In recent years, with the rapid development of Internet technology, online shopping has become a mainstream way for users to purchase and consume. Sentiment analysis of a large number of user reviews on e-commerce platforms can effectively improve user satisfaction. This paper proposes a new sentiment analysis model-SLCABG, which is based on the sentiment lexicon and combines Convolutional Neural Network (CNN) and attention-based Bidirectional Gated Recurrent Unit (BiGRU). In terms of methods, the SLCABG model combines the advantages of sentiment lexicon and deep learning technology, and overcomes the shortcomings of existing sentiment analysis model of product reviews. The SLCABG model combines the advantages of the sentiment lexicon and deep learning techniques. First, the sentiment lexicon is used to enhance the sentiment features in the reviews. Then the CNN and the Gated Recurrent Unit (GRU) network are used to extract the main sentiment features and context features in the reviews and use the attention mechanism to weight. And finally classify the weighted sentiment features. In terms of data, this paper crawls and cleans the real book evaluation of dangdang.com, a famous Chinese e-commerce website, for training and testing, all of which are based on Chinese. The scale of the data has reached 100000 orders of magnitude, which can be widely used in the field of Chinese sentiment analysis. The experimental results show that the model can effectively improve the performance of text sentiment analysis.

INDEX TERMS Attention mechanism, CNN, BiGRU, sentiment analysis, sentiment lexicon.

I. INTRODUCTION

With the rapid development and popularization of e-commerce technology, more and more users like to shop on various e-commerce platforms. Compared with the way of off-line shopping in physical stores, users can shop at any time and any where, and do not have to wait for the weekend to go shopping, which saves time and effort. Moreover, the products on e-commerce platforms are full of varieties and styles, and consumers can buy the desired products without leaving home [1]. However, while online shopping brings convenience to consumers, due to the virtuality of the e-commerce platforms, there are many problems in the products sold on the platforms, such as inconsistency between descriptive information and real goods, poor quality of goods, imperfect after-sales of goods and so on [2]. Therefore, it is of great significance to conduct sentiment analysis on the commodity evaluation of the purchased products on electronic commerce platforms.

Analyzing the sentiment tendency of consumer evaluation can not only provide a reference for other consumers but also help businesses on e-commerce platforms to improve service quality and consumer satisfaction.

Sentiment analysis for product reviews, also known as text orientation analysis or opinion mining, refers to the process of automatically analyzing the subjective commentary text with

The associate editor coordinating the review of this manuscript and approving it for publication was Alberto Cano.

the customer's emotional color and deriving the customer's emotional tendency [3].

At present, the main methods of text mining are rule-based method, machine learning method and combined method. Among them, the rule-based method also includes the lexicon-based method. Machine learning methods include traditional machine learning methods such as conditional random fields and deep learning methods. In the rest of this paper, the machine learning techniques we mentioned all refer to traditional machine learning techniques. Deep learning methods have been widely used in various fields, such as image recognition [4], [5], object detection [6], [7], transportation [8], network optimization [9], sensor networks [10]–[14], system security [15], etc. In recent years, many researchers have integrated traditional machine learning methods and deep learning methods into the field of text sentiment analysis by constructing the sentiment lexicon, and achieved good results [16].

The core of the sentiment lexicon-based approach is to construct a sentiment lexicon. The corresponding sentiment lexicon is constructed by selecting appropriate sentimental words, degree adverbs, and negative words, and the sentimental intensity and sentimental polarity are marked for the constructed sentiment lexicon. After the text is input, the words in the input text are matched with the sentiment words in the sentiment lexicon, and the matched sentiment words are weighted and summed to obtain the sentiment value of the input text, thereby determining the sentimental polarity of the input text according to the sentiment value.

Although there are already some methods to automatically obtain the word vector features of the text such as Word2Vec, FastText, and Glove, the traditional machine learning method still needs to extract the emotional features of the structured data from the input text through human intervention, vectorize the text, and then use the traditional machine learning model to classify the sentiment text features [17]. This method usually requires human intervention to obtain the sentiment category of the input text. Traditional machine learning methods commonly used include naive bayes, support vector machine, maximum entropy, random forest and conditional random fields model [18], [19].

In recent years, deep learning has made great achievements in many fields. Compared with traditional machine learning methods, deep learning does not need human intervention features, but deep learning needs massive data as support. Deep learning-based methods automatically extract features from different neural network models and learn from their own errors [20]. The neural network model is usually composed of multiple hierarchies, which can be a layer-by-layer abstraction, and the layers can be mapped by nonlinear activation functions, so it can fit very complex features and learn the hidden deep features between texts [21]. The deep learning models commonly used in the field of text sentiment analysis are CNN, Recurrent Neural Network (RNN), LSTM, and Gated Recurrent Unit (GRU) [22].

In order to improve the performance of existing sentiment analysis models in the sentiment analysis field of product reviews, this paper proposes a SLCABG model based on the advantages of the sentiment lexicon and deep learning techniques. The main contributions of this paper are as follows:

1. We propose a new sentiment analysis model based on the advantages of sentiment lexicon, word vectors, CNN, GRU and the attention mechanism, and experiment on the book review dataset from the real e-commerce book website to verify the effectiveness of the model.
2. We analyzed the influence of related factors such as the size of the thesaurus, the length of the input sentence and the number of iterations of the model on the performance of the model, and conducted experiments to optimize our model.

The rest of the paper is organized as follows: Section II introduces the relevant research progress of text sentiment analysis, Section III describes our proposed SLCABG model in detail, and Section IV describes the experimental process and results of our validation of the SLCABG model. Section V and Section VI discusses and summarizes the experiments we conducted.

II. RELATED WORKS

This section reviews the work done in the field of text sentiment analysis from three aspects: sentiment analysis methods based on sentiment lexicon, sentiment analysis methods based on machine learning and sentiment analysis methods based on deep learning.

A. SENTIMENT ANALYSIS BASED ON SENTIMENT LEXICON

Taboada *et al.* [23] proposed the Semantic Orientation Calculator to extract sentiment from text using dictionaries of words annotated with polarity and strength. Jurek *et al.* [24] proposed a new lexicon-based sentiment analysis algorithm using namely sentiment normalization and evidence-based combination function. In addition to using sentiment terms, Asghar *et al.* [25] also integrated emoticons, modifiers and domain specific terms to analyze sentiment analysis of online user comments. Bandhakavi *et al.* [26] proposed a unigram mixture model (UMM) based DSEL by using labeled and weakly-labeled emotion text to extract effective features for emotion classification. Dhaoui *et al.* [27] used the LIWC2015 lexicon and RTextTools machine learning package to compare the sentiment analysis method based on lexicon and machine learning. Khoo and Johnkhan [28] proposed a general sentiment lexicon called WKWSCI Sentiment Lexicon and compares it with the existing sentiment lexicons. Zhang *et al.* [29] analyzed text sentiment of Chinese microblog by using extended sentiment lexicons of added degree adverb lexicon, network word lexicon and negative word lexicon. Keshavarz and Abadeh [30] used a combination of corpus and lexicons to construct adaptive sentiment vocabulary to improve the sentiment classification accuracy of Weibo. Feng *et al.* [31] constructed a two-layer graph model using emoji and candidate sentiment words, and selected the top words in the model as sentiment words. Although the sentiment lexicon-based approach can achieve great performance, there are significant limitations in different fields and its manual maintenance requires extremely high costs. Therefore, the machine learning-based method that automatically extracts sentiment features with a little manual intervention has become a better choice for researchers.

B. SENTIMENT ANALYSIS BASED ON MACHINE LEARNING

Manek *et al.* [32] proposed the method of feature extraction based on gini index and classification by support vector machine (SVM). Hai *et al.* [33] proposed a new probabilistic supervised joint emotion model (SJSM), which could not only identify semantic sentiments from the comment data but also infer the overall sentiment of the comment data. Singh *et al.* [34] used naive bayes, J48, BFTree and OneR four machine learning algorithms for text sentiment analysis. Huang *et al.* [35] proposed a multi-modal joint sentiment theme model. Based on the introduction of user personality features and sensitive influence factors, the model uses latent dirichlet allocation (LDA) model to analyze the hidden user sentiments and topic types in Weibo text. Huq *et al.* [36] used SVM and k-nearest neighbors (KNN) algorithms to analyze the sentiment of twitter data. Long *et al.* [37] used SVM to classify stock forum posts using additional samples containing prior knowledge. Although the machine learning-based method can automatically extract features, it often relies on manual feature selection. However, the deep learning-based approach does not require manual intervention at all. It can automatically select and extract features through the neural network structure and can learn from its own errors.

C. SENTIMENT ANALYSIS BASED ON DEEP LEARNING

Jianqiang *et al.* [38] used the contextual semantic features and the co-occurrence statistical features of the words in the tweet and the n-gram feature input convolutional neural network to analyze the sentiment polarity. Hyun *et al.* [39] proposed a target-dependent convolutional neural network (TCNN). The model uses the distance relationship between the target word and the surrounding words to learn the influence of surrounding words on the target words. Attention mechanisms arise because each word in a sentence has a different effect on the emotional polarity of the sentence. In order to combine the dominant and recessive features in the sentence, Ma *et al.* [40] proposed an extended LSTM called Sentic LSTM. The model unit includes a separate output gate for inserting token level memory and concept level input. Based on the mathematical theory of regression neural network, Chen *et al.* [41] proposed the LSTM model for the detailed emotional analysis of Chinese product reviews. Wen *et al.* [42] proposed a memristor-based long short-term memory (MLSTM) network hardware design using memristor crossbars. Abid *et al.* [43] proposed a joint structure that combines CNN and RNN. The structure uses the RNN to locate the CNN and uses the global average pool layer to capture long-term dependencies with CNN. Chen *et al.* [44] proposed a divide-and-conquer method, which first uses a neural network-based sequence model to classify sentences, and then inputs each set of sentences into a convolutional neural network for sentiment classification. Hu *et al.* [45] performed sentiment analysis of short texts by constructing a keyword vocabulary and combining the LSTM model.

III. METHODS

To improve the accuracy of sentiment analysis on product reviews, we combined the advantages of sentiment lexicon, CNN model, GRU model and attention mechanism to propose SLCABG model. First, the sentiment lexicon is used to enhance the sentiment features in the reviews. Then the CNN and GRU networks are used to extract the main sentiment features and context features in the reviews and use the attention mechanism to weight. Finally, the weighted sentiment features were classified. The model consists of six layers: an embedded layer, a convolutional layer, a pooled layer, a BiGRU layer, an attention layer, and a fully connected layer. The model structure is shown in Figure 1.

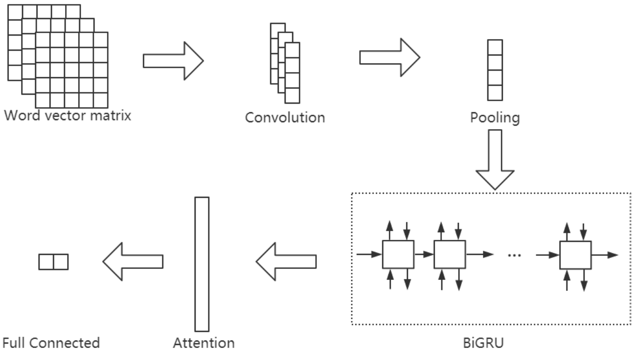

FIGURE 1. The structure of the SLCABG model.

For the rest of this section, we will describe the SLCABG model in detail.

Suppose the input text statement is $S = \{w_1, w_2, \cdots, w_i, \cdots, w_n\}$, where w_i represents a word in S, and the task of our model is to predict the sentimental polarity P of the statement S.

A. CONSTRUCTING AN SENTIMENT LEXICON

The function of the sentiment lexicon is to give each word w_i in S a corresponding sentimental weight sw_i.

Commonly used Chinese open source sentiment dictionaries are: HowNet [46], the sentiment vocabulary ontology library of Dalian University of Technology [47] and the simplified Chinese sentiment polarity lexicon of Taiwan University(NTUSD) [48]. Commonly used English open source sentiment dictionary is mainly WordNet [49].

Our sentiment lexicon is based on the emotional vocabulary ontology library of Dalian University of Technology.

Remove the sentiment words that represent neutrality and both sexes, and retain the sentiment words that represent derogatory and derogatory, that is, retain words with polarities of 1 and 2. Sentiment words are divided into five categories according to their sentiment intensity, namely, 1, 3, 5, 7 and 9, with their sentiment intensity as their sentiment weight, and sentiment words with negative sentiment polarity multiply their sentiment weight by -1.

The expression of the sentiment weight of the word after construction is:

$$\text{senti}\,(w_i) = \begin{cases} sw_i, & w_i \in SD \\ 1, & w_i \notin SD \end{cases} \tag{1}$$

where w_i represents a word, sw_i represents the weight of the word w_i in the sentiment lexicon, and SD represents the sentiment lexicon.

B. EMBEDDED LAYER

The main function of this layer is to represent the text statement S as a weighted word vector matrix.

In traditional natural language processing tasks, words in text data are usually represented by discrete values, namely One-Hot encoding. The One-Hot encoding method combines all the words in the lexicon to form a long vector. The dimension of the vector is equal to the number of words in the lexicon. Each dimension corresponds to one word.

The value of a word corresponding to a dimension is 1, and the value of other dimensions is 0. The biggest advantage of One-Hot encoding is that it is simple. However, the One-Hot vector of each word is independent and cannot reflect the relationship between words and words. Moreover, when the number of words in the lexicon is large, the dimension of the word vector will be very large, and a dimensional disaster will occur.

To solve the problem of one-Hot coding, the researchers proposed the encoding of word vectors [50]. The core idea is to represent words as a low-dimensional continuous dense vector, and words with similar meanings will be mapped to similar positions in the vector space. Commonly used word vector implementation models are Word2Vec [51], Glove [52], ELMo [53] and BERT [54].

The BERT model is a new pre-trained language model proposed by Google for use in the field of natural language processing. It is a model that truly implements a bidirectional language and has better performance than other word vector models.

In our model, we use the BERT model to train word vectors.

Each word w_i in S is converted to a word vector v_i using a BERT model, where v_i is a 768-dimensional vector. Then, weigh the word vector using sentiment weights.

$$v'_i = v_i * senti(w_i) \tag{2}$$

We use the weighted word vector matrix as the output of the embedded layer.

C. CONVOLUTION LAYER

The main function of this layer is to extract the most important local features of the input matrix [55]. In the field of natural language processing, the word vector representation of a word is usually a whole. Therefore, the convolution kernel width in the convolutional layer usually takes the dimension of the word vector.

For the input matrix $\mathrm{V} = [v'_1, v'_2, \cdots, v'_i, \cdots, v'_n]$, the convolution operation is:

$$v''_i = f(W \cdot V_{[i:i+k-1]} + b) \tag{3}$$

where $\mathrm{W} \in R^{k*m}$ represents a weight matrix, k and m represents the height and width of the convolution kernel, b represents the offsets, and f represents the activation function ReLU.

After the convolution operation is completed, the eigenvector matrix V' is expressed as

$$V' = [v_1{}'', v_2{}'', \cdots, v_i{}'', \cdots, v_{n-k+1}{}''] \tag{4}$$

D. POOLING LAYER

The main function of this layer is to compress the text features obtained by the convolutional layer and extract the main features. Pooling operations are usually divided into average pooling and max pooling. For text sentiment analysis, the most influential is usually a few words or phrases in the sentence, so we use the k-max pooling.

For the input vector $v_i{}''$, its k-max pooling operation is:

$$\mathrm{x} = [x_1, x_2, \cdots, x_i, \cdots, x_{m-k+1}] \tag{5}$$

$$x_i = \max(v_i{}'', v_{i+1}{}'', \cdots, v_{i+k-1}{}'') \tag{6}$$

where m represents the dimension of the vector v''_i, and max represents the maximum function.

E. BiGRU LAYER

The main function of this layer is to extract the context features of the input matrix. The GRU model is a variant of the recurrent neural network model and is commonly used to process sequence information. It can combine the historical information of the previous moment to influence the current output, and extract the context features in the sequence data. In the text data, both the preceding and the following words affect the current word, so we use the BiGRU model to extract the contextual features of the input text.

The BiGRU consists of a forward GRU and a reverse GRU, which are used to process forward and reverse information, respectively. For the input x_t at time t, the hidden states obtained by the forward GRU and the reverse GRU are h'_t and h''_t, respectively.

$$h'_t = \overrightarrow{GRU}(x_t, h'_{t-1}) \tag{7}$$

$$h''_t = \overleftarrow{GRU}(x_t, h''_{t-1}) \tag{8}$$

The combination h'_t and $h_t{}''$ is $h_t = [h'_t; h_t{}'']$ as the hidden state output at time t.

F. ATTENTION LAYER

In a text statement, each word has a different influence on the sentiment polarity of the whole sentence. Some words have a decisive effect on the sentiment of the whole sentence, while others do not affect the sentence sentiment. So we use the attention mechanism to give different weights to different words in a sentence.

For the hidden state h_i of the BiGRU layer output, the weight a_i is expressed as:

$$u_i = \tanh(W \cdot h_i + b) \tag{9}$$

$$a_i = \frac{e^{u_i^T \cdot u_w}}{\sum_i e^{u_i^T \cdot u_w}} \tag{10}$$

where W represents the weight matrix, b is the offset, and u_w represents a global context vector (the parameters to be learned).

The weight a_i and the hidden layer output h_i are weighted and summed as a feature vector representation of the input sentence S.

$$\mathrm{X} = \sum\nolimits_i a_i \cdot h_i \tag{11}$$

G. FULLY CONNECTED LAYER

The main function of this layer is to classify the input feature matrix.

Its output is defined as:

$$\mathrm{Y} = f\,(W \cdot X) + b \tag{12}$$

where f represents the activation function sigmoid, w represents the weight matrix, and b represents the offset.

This layer maps the input feature to a value in the interval [0,1]. The closer the value is to 0, the closer the sentiment polarity of the input text S is to the negative direction. Conversely, if the value is closer to 1, it represents the input text S. The sentiment polarity is closer to the positive.

IV. EXPERIMENTS

In this section, we evaluate our model for sentiment analysis tasks.

A. DATASET

The dataset used in this experiment is the data of book reviews collected from Dangdang using web crawler technology. The book reviews in the original data are divided into five levels, one to five stars, we divide the five levels into two categories, 1-2 stars are defined as negative reviews, 3-5 stars are defined as positive reviews. We manually screen these product reviews by star rating to ensure that all reviews in the positive dataset are positive reviews and all reviews in the negative dataset are negative reviews. We take the dataset after manual processing as the dataset of this paper. The dataset includes 100000 reviews, of which 50000 are positive and 50000 are negative. We have submitted the data and code to the open source community (https://github.com/ly2014/sentimen-analysis-based-on-sentiment-lexicon-and-deep-learning.git), which can be used by other researchers in the Chinese sentiment analysis field.

B. PERFORMANCE METRICS

The model evaluation metrics used in this paper are accuracy, precision, recall, and F1 score, which are consistent with those used in other studies.

The calculation parameters are defined as follows:

(1) TP: the number of comments categorizing positive merchandise comments as positive.

(2) FP: the number of comments that classify negative product comments as positive.

(3) TN: the number of negative comments classified as negative comments.

(4) FN: the number of comments categorizing positive merchandise reviews as negative.

(5) Accuracy: the ratio of correctly predicted comments to the total comments.

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

(6) Precision: the ratio of correctly predicted positive comments to the total predicted positive comments.

$$\text{precision} = \frac{TP}{TP + FP} \tag{14}$$

(7) Recall: the ratio of correctly predicted positive comments to the all comments in actual class.

$$\text{recall} = \frac{TP}{TP + FN} \tag{15}$$

(8) F1: the weighted average of precision and recall.

$$\text{F1} = \frac{2 * precision * recall}{precision + recall} \tag{16}$$

C. DATA PREPROCESSING

(1) Use the python word segmentation tool jieba package to perform word segmentation on the comment data in the data set. Add the emotional words in the sentiment dictionary to jieba's custom dictionary to prevent the emotional words in the sentence from being separated into two words.

(2) Remove stop words and non-Chinese characters (including English characters and Chinese characters) in the word segmentation results.

(3) The number of different words in the dataset after the pre-processing is counted, the number of occurrences of each word, the length of the largest word included in each review, and the length of the word contained in each review in the calculated dataset. The average review length is used as the fixed length of words in each review. If the review length is larger than the fixed length, it is intercepted, and if the review length is smaller than the fixed length, 0 will be added.

TABLE 1. The model parameters.

parameters	value
the length of the input sentence	648
the dimension of the word vector	768
the thesaurus size	35000
the size of the convolution kernel	3x3
the number of hidden neurons in convolution layer	128
the number of hidden neurons in BiGRU layer	128
dropout	0.4

TABLE 2. The cross-validation results.

method	accuracy	precision	recall	F1
10-fold cross-validation	93.2%	92.5%	93.5%	93.0%
5*2 cross-validation	93.3%	93.2%	93.2%	93.2%

TABLE 3. The effect of the fixed length of the input statement on the model.

sentence_length	accuracy	precision	recall	F1
12	91.9%	91.4%	92.0%	91.7%
648	93.4%	92.9%	93.8%	93.3%

TABLE 4. The impact of the thesaurus size on the model.

The number of words	accuracy	precision	recall	F1
50000	93.5%	93%	93.6%	93.3%
45000	93.1%	91.8%	94.2%	93%
40000	93.4%	92 %	94.2%	93.1%
35000	93.8%	92%	95%	93.6%
30000	93.5%	91.7%	95.1%	93.4%
25000	93.3%	92.1%	94.1%	93.1%
20000	93.5%	93%	93.4%	93.2%

TABLE 5. The effect of the number of iterations on the model.

epoch	accuracy	precision	recall	F1
3	93.2%	92.7%	93.5%	93.1%
5	93.5%	93.5%	92.9%	93.2%
8	93.9%	93.3%	94.3%	93.8%
10	93.3%	93.3%	93.1%	93.2%
12	93.0%	92.8%	93.4%	93.1%
15	92.6%	91.8%	93.2%	92.5%

TABLE 6. The impact of the dropout value on the model.

dropout	accuracy	precision	recall	F1
0.2	93.1%	92.1%	94.1%	93.1%
0.4	93.8%	93.7%	93.5%	93.6%
0.5	93.5%	93.5%	92.9%	93.2%
0.6	93.5%	93.8%	93.2%	93.5%
0.8	92.7%	92%	93.6%	92.8%

D. EXPERIMENTAL RESULTS

The model parameters used in this experiment are shown in Table 1.

In order to more accurately evaluate the performance of our proposed SLCABG model, we use 10-fold cross-validation [56] and 5*2 cross-validation [57] to divide the dataset we use. In the 10-fold cross-validation method, we randomly divide the data set into 10 parts, using 9 of them as the training set in turn, and the remaining 1 part as the verification set, taking the mean of these 10 results as the evaluation result of our model. In the 5*2 cross-validation method, we randomly divided the dataset into two parts, used one of them as the training set in turn, and the remaining part as the test set, and the average of the five results was taken as the evaluation result of our model. Tables 2 show the experimental results of the SLCABG model under 10-fold cross-validation and 5*2-fold cross-validation, respectively.

Since the length of the text statement in the dataset is different, we will take the length of the statement to a certain value when we input the model. We select the maximum sentence length and the average sentence length in the dataset to conduct experiments. The experimental results are shown in Table 3. We find that using the average sentence length as the fixed length of the input sentence results in the loss of a part of the context feature for sentences longer than the average sentence length, which in turn affects the performance of the model.

In the experiment, we found that the number of words in the thesaurus has a certain impact on the performance of the model. We start with the number of words in the thesaurus starting from 50000, and the frequency of occurrence of the words in the thesaurus is reduced from the words with the lowest frequency, and an experiment is repeated for every 5000 words. The experimental results are shown in Table 4 and Figure2. As can be seen from the table, when the number of words in the lexicon is in the appropriate number of 35,000 words, the performance of the model is optimal. As the number of words in the thesaurus increases or decreases, the performance of the model decreases.

In the experiment, different iterations of the model will also affect the performance of the model. As the number of iterations of the model increases, the performance of the model will first rise and then fall. It can be seen from Table 5 and Figure 3 that when the number of iterations of the model is less than 8 times, the performance of the model increases with the number of iterations. When the number of iterations of the model is greater than 8 times, the model gradually overfits, resulting in the model. The performance is degraded.

To improve the generalization performance of our model, we used dropout in the model. By selecting different dropout values for experiments, we found that when the value of dropout is 0.4, the performance of the model is optimal. The experimental results are shown in Table 6 and Figure 4.

In order to explore the influence of the word vector weighted by the sentiment lexicon on our model, we use the weighted word vector and the unweighted word vector to experiment. The comparison results are shown in Table 7. It can be seen from the table that the word vector weighted

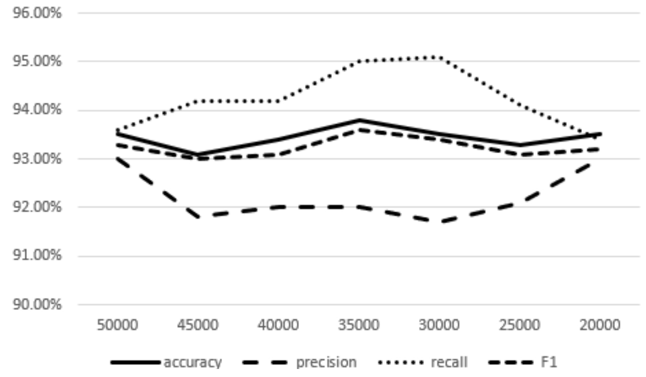

FIGURE 2. The impact of the thesaurus size on the model.

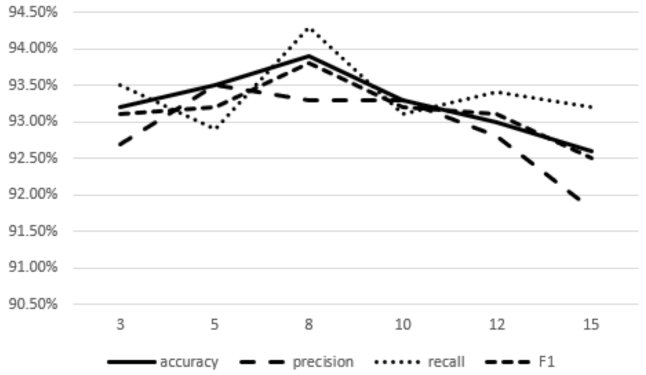

FIGURE 3. The effect of the number of iterations on the model.

TABLE 7. The impact of the weighted word vector on the model.

type	accuracy	precision	recall	F1
the weighted word vector	93.5%	93%	93.6%	93.3%
the unweighted word vector	92.8%	92.1%	93.1%	92.6%

by the sentiment lexicon can enhance the sentiment features expressed in the sentence, so the model can obtain better performances than the ordinary word vector.

We compared the sentiment analysis effects of the SLCABG model with the common sentiment analysis models (NB, SVM, CNN, and BiGRU) on the dataset. The comparison results are shown in Table 8 and Figure 5. The experimental results show that the classification performance of the deep learning model (CNN and BiGRU) is significantly better than the machine learning model (NB and SVM). Adding the attention mechanism based on the deep learning model can improve the classification performance of the model. The classification performance of the SLCABG model proposed by our comprehensive sentiment dictionary, CNN, BIGRU and Attention are also improved compared with the commonly used deep learning model.

V. DISCUSSION

This paper presents a new sentiment analysis model (SLCABG). Before inputting the word vector matrix of the text into the network model, the sentiment dictionary is used to weight the word vector of the sentiment words in the text to enhance the sentiment features in the text. Then use CNN to extract the important features in the input matrix, then use BiGRU to consider the order information of the input text, extract the text context features, and then use the attention mechanism to assign different weights to different input features, highlight the sentiment features of the text, and finally use Fully connected to classify sentiment features. Compared with other methods, our method enhances the sentiment features of the input text, and integrates the text context features and main features to enhance the classification performance of the sentiment analysis model.

In addition, we explored the impact of the length of the input text statement on the performance of the model. We selected the maximum sentence length and the average sentence length in the data set as the fixed length of the input sentence. We found that the performance of the model is better than the average sentence length when the input length

TABLE 8. Performance comparison of different sentiment analysis models.

model	accuracy	precision	recall	F1
NaiveBayes [58]	57.9%	55.6%	79.2%	65.3%
SVM [36]	67.7%	93.8%	38.4%	54.5%
CNN [59]	90.9%	91%	90.2%	90.6%
CNN+Attention [60]	91.4%	90.8%	91.6%	91.2%
BiGRU [61]	92.6%	91.1%	94.1%	92.6%
BiGRU+Attention [62]	93.1%	92.8%	93.2%	93%
SLCABG (Ours)	93.5%	93%	93.6%	93.3%

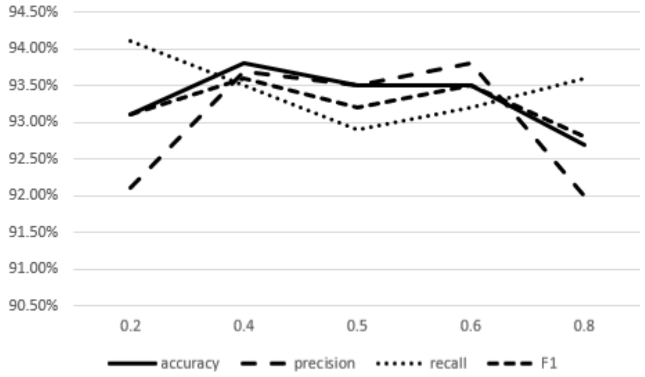

FIGURE 4. The impact of the dropout value on the model.

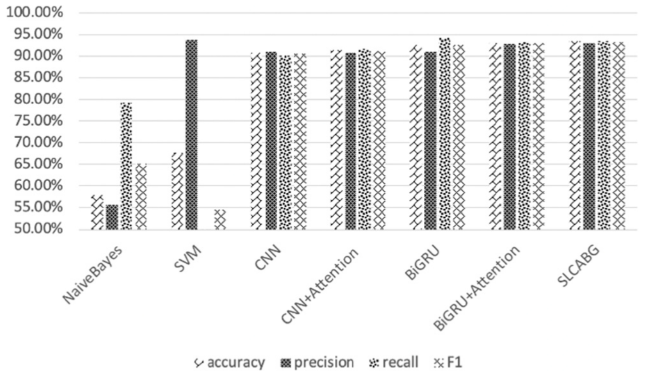

FIGURE 5. Model performance comparison.

is fixed to the maximum sentence length. When the input length is averaged, the statement with a length greater than the average sentence length loses some of the context features, which affects the final performance of the model. The size of the lexicon we selected will also affect the performance of the model. Through experiments, we found that the performance of the model is optimal when the size of the word in the lexicon takes a certain intermediate value. Because some words in a sentence belong to a universal word and do not affect the sentiment features of the sentence, we should exclude these words when constructing the thesaurus. The difference in the number of iterations of the model also affects the performance of the model. Initially, as the number of iterations increases, the model can better fit the data, and the performance of the model is gradually improved. After reaching a certain value, the performance of the model is optimal. Then, as the number of iterations increases, the model appears to have a fitting phenomenon, and the excessive fitting of the training data leads to a gradual decrease in the performance of the model on the test set. In order to improve the generalization performance of the model, we used dropout in the model. We experimented with different dropout values. The experimental results show that when the value of dropout is 0.4, our model performance is optimal.

VI. CONCLUSION

With the rapid development of e-commerce platforms in recent years, the sentiment analysis technology of product reviews has gained more and more attention. In this paper, a SLCABG model for sentiment analysis on product reviews is constructed using sentiment dictionary, BERT model, CNN model, BiGRU model, and attention mechanism. First, the sentiment lexicon is used to enhance the sentiment features in the reviews. Then the CNN and GRU networks are used to extract the main sentimental and contextual features of the reviews, and attention mechanism is used to weight them. Finally, the weighted sentiment features are classified. By analyzing the experimental results, it can be found that the model has better classification performance than other sentiment analysis models. By using our model to analyze user reviews, we can help merchants on e-commerce platforms to obtain user feedback in time to improve their service quality and attract more customers to patronize.

Besides, with the continuous enrichment of the sentiment lexicon and the increase of the dataset, the classification accuracy of the model will gradually improve.

However, the approach proposed in this paper can only divide sentiment into positive and negative categories, which is not suitable in areas with high requirements for sentiment refinement. Therefore, the next step is to study the sentiment fineness classification of text.

REFERENCES

[1] R. Liang and J.-Q. Wang, "A linguistic intuitionistic cloud decision support model with sentiment analysis for product selection in E-commerce," *Int. J. Fuzzy Syst.*, vol. 21, no. 3, pp. 963–977, Apr. 2019.

[2] P. Ji, H.-Y. Zhang, and J.-Q. Wang, "A fuzzy decision support model with sentiment analysis for items comparison in E-commerce: The case study of PConline.Com," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 10, pp. 1993–2004, Oct. 2019.

[3] D. Zeng, Y. Dai, F. Li, J. Wang, and A. K. Sangaiah, "Aspect based sentiment analysis by a linguistically regularized CNN with gated mechanism," *J. Intell. Fuzzy Syst.*, vol. 36, no. 5, pp. 3971–3980, May 2019.

[4] Y. Chen, J. Wang, S. Liu, X. Chen, J. Xiong, J. Xie, and K. Yang, "Multiscale fast correlation filtering tracking algorithm based on a feature fusion model," *Concurrency Comput., Pract. Exper.*, to be published, doi: 10.1002/cpe.5533.

[5] Y. Chen, R. Xia, Z. Wang, J. Zhang, K. Yang, and Z. Cao, "The visual saliency detection algorithm research based on hierarchical principle component analysis method," *Multimedia Tools Appl.*, to be published.

[6] J. Zhang, Y. Wu, W. Feng, and J. Wang, "Spatially attentive visual tracking using multi-model adaptive response fusion," *IEEE Access*, vol. 7, pp. 83873–83887, 2019.

[7] Y. Chen, J. Wang, R. Xia, Q. Zhang, Z. Cao, and K. Yang, "The visual object tracking algorithm research based on adaptive combination kernel," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 12, pp. 4855–4867, Dec. 2019.

[8] J. Zhang, W. Wang, C. Lu, J. Wang, and A. K. Sangaiah, "Lightweight deep network for traffic sign classification," *Ann. Telecommun.*, to be published, doi: 10.1007/s12243-019-00731-9.

[9] Y. Tu, Y. Lin, J. Wang, and J.-U. Kim, "Semi-supervised learning with generative adversarial networks on digital signal modulation classification," *Comput. Mater. Continua*, vol. 55, no. 2, pp. 243–254, 2018.

[10] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim, "An intelligent data gathering schema with data fusion supported for mobile sink in wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 3, Mar. 2019, Art. no. 155014771983958.

[11] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim, "Energy efficient routing algorithm with mobile sink support for wireless sensor networks," *Sensors*, vol. 19, no. 7, p. 1494, 2019.

[12] J. Wang, X. Gu, W. Liu, A. K. Sangaiah, and H.-J. Kim, "An empower hamilton loop based data collection algorithm with mobile agent for WSNs," *Hum.-Centric Comput. Inf. Sci.*, vol. 9, no. 1, p. 18, Dec. 2019.

[13] J. Wang, Y. Gao, X. Yin, F. Li, and H.-J. Kim, "An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks," *Wireless Commun. Mobile Comput.*, vol. 2018, pp. 1–9, Dec. 2018.

[14] J. Wang, Y. Gao, K. Wang, A. K. Sangaiah, and S.-J. Lim, "An affinity propagation-based self-adaptive clustering method for wireless sensor networks," *Sensors*, vol. 19, no. 11, p. 2579, Jun. 2019.

[15] Z. Tang, X. Ding, Y. Zhong, L. Yang, and K. Li, "A self-adaptive Bell–LaPadula model based on model training with historical access logs," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 8, pp. 2047–2061, Aug. 2018.

[16] Y. Chen, J. Xiong, W. Xu, and J. Zuo, "A novel online incremental and decremental learning algorithm based on variable support vector machine," *Cluster Comput.*, vol. 22, no. S3, pp. 7435–7445, May 2019.

[17] L. Wang, X.-K. Wang, J.-J. Peng, and J.-Q. Wang, "The differences in hotel selection among various types of travellers: A comparative analysis with a useful bounded rationality behavioural decision support model," *Tourism Manage.*, vol. 76, Feb. 2020, Art. no. 103961.

[18] L. Yang, Y. Zhou, and Y. Zheng, "Annotating the Literature with Disease Ontology," *Chin. J. Electron.*, vol. 26, no. 6, pp. 1261–1268, Nov. 2017.

[19] M. Ahmad, S. Aftab, S. S. Muhammad, and S. Ahmad, "Machine learning techniques for sentiment analysis: A review," *Int. J. Multidiscip. Sci. Eng*, vol. 8, no. 3, pp. 27–32, 2017.

[20] D. Zeng, Y. Dai, F. Li, R. Sherratt, and J. Wang, "Adversarial learning for distant supervised relation extraction," *Comput., Mater. Continua*, vol. 55, no. 1, pp. 121–136, 2018.

[21] L. Yang, J. Wang, Z. Tang, and N. N. Xiong, "Using conditional random fields to optimize a self-adaptive Bell-LaPadula model in control systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.

[22] L. Zhang, S. Wang, and B. Liu, "Deep learning for sentiment analysis: A survey," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 4, 2018, Art. no. e1253.

[23] M. Taboada, J. Brooke, M. Tofiloski, K. Voll, and M. Stede, "Lexicon based methods for sentiment analysis," *Comput. Linguistics*, vol. 37, no. 2, pp. 267–307, 2011.

[24] A. Jurek, M. D. Mulvenna, and Y. Bi, "Improved lexicon-based sentiment analysis for social media analytics," *Secur. Informat.*, vol. 4, no. 1, p. 9, 2015.

[25] M. Z. Asghar, A. Khan, S. Ahmad, M. Qasim, and I. A. Khan, ''Lexicon-enhanced sentiment analysis framework using rule-based classification scheme,'' *PLoS ONE*, vol. 12, no. 2, Feb. 2017, Art. no. e0171649.

[26] A. Bandhakavi, N. Wiratunga, D. Padmanabhan, and S. Massie, ''Lexicon based feature extraction for emotion text classification,'' *Pattern Recognit. Lett.*, vol. 93, pp. 133–142, Jul. 2017.

[27] C. Dhaoui, C. M. Webster, and L. P. Tan, ''Social media sentiment analysis: Lexicon versus machine learning,'' *J. Consum. Marketing*, vol. 34, no. 6, pp. 480–488, Sep. 2017.

[28] C. S. Khoo and S. B. Johnkhan, ''Lexicon-based sentiment analysis: Comparative evaluation of six sentiment lexicons,'' *J. Inf. Sci.*, vol. 44, no. 4, pp. 491–511, Aug. 2018.

[29] S. Zhang, Z. Wei, Y. Wang, and T. Liao, ''Sentiment analysis of Chinese micro-blog text based on extended sentiment dictionary,'' *Future Gener. Comput. Syst.*, vol. 81, pp. 395–403, Apr. 2018.

[30] H. Keshavarz and M. S. Abadeh, ''ALGA: Adaptive lexicon learning using genetic algorithm for sentiment analysis of microblogs,'' *Knowl.-Based Syst.*, vol. 122, pp. 1–16, Apr. 2017.

[31] S. Feng, K. Song, D. Wang, and G. Yu, ''A word-emoticon mutual reinforcement ranking model for building sentiment lexicon from massive collection of microblogs,'' *World Wide Web*, vol. 18, no. 4, pp. 949–967, Jul. 2015.

[32] A. S. Manek, P. D. Shenoy, M. C. Mohan, and K. Venugopal, ''Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier,'' *World Wide Web*, vol. 20, no. 2, pp. 135–154, Mar. 2017.

[33] Z. Hai, G. Cong, K. Chang, P. Cheng, and C. Miao, ''Analyzing sentiments in one go: A supervised joint topic modeling approach,'' *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 6, pp. 1172–1185, Jun. 2017.

[34] J. Singh, G. Singh, and R. Singh, ''Optimization of sentiment analysis using machine learning classifiers,'' *Hum.-Centric Comput. Inf. Sci.*, vol. 7, no. 1, p. 32, 2017.

[35] F. Huang, S. Zhang, J. Zhang, and G. Yu, ''Multimodal learning for topic sentiment analysis in microblogging,'' *Neurocomputing*, vol. 253, pp. 144–153, Aug. 2017.

[36] M. R. Huq, A. Ali, and A. Rahman, ''Sentiment analysis on Twitter data using KNN and SVM,'' *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, pp. 19–25, 2017.

[37] W. Long, Y.-R. Tang, and Y.-J. Tian, ''Investor sentiment identification based on the universum SVM,'' *Neural Comput. Appl.*, vol. 30, no. 2, pp. 661–670, Jul. 2018.

[38] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, ''Deep convolution neural networks for Twitter sentiment analysis,'' *IEEE Access*, vol. 6, pp. 23253–23260, 2018.

[39] D. Hyun, C. Park, M.-C. Yang, I. Song, J.-T. Lee, and H. Yu, ''Target-aware convolutional neural network for target-level sentiment analysis,'' *Inf. Sci.*, vol. 491, pp. 166–178, Jul. 2019.

[40] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, ''Sentic LSTM: A hybrid network for targeted aspect-based sentiment analysis,'' *Cogn. Comput.*, vol. 10, no. 4, pp. 639–650, Aug. 2018.

[41] H. Chen, S. Li, P. Wu, N. Yi, S. Li, and X. Huang, ''Fine-grained sentiment analysis of Chinese reviews using LSTM network,'' *J. Eng. Sci. Technol. Rev.*, vol. 11, no. 1, pp. 174–179, 2018.

[42] S. Wen, H. Wei, Y. Yang, Z. Guo, Z. Zeng, T. Huang, and Y. Chen, ''Memristive LSTM network for sentiment analysis,'' *IEEE Trans. Syst., Man, Cybern. Syst.*, to be published.

[43] F. Abid, M. Alam, M. Yasir, and C. Li, ''Sentiment analysis through recurrent variants latterly on convolutional neural network of Twitter,'' *Future Gener. Comput. Syst.*, vol. 95, pp. 292–308, Jun. 2019.

[44] T. Chen, R. Xu, Y. He, and X. Wang, ''Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN,'' *Expert Syst. Appl.*, vol. 72, pp. 221–230, Apr. 2017.

[45] F. Hu, L. Li, Z.-L. Zhang, J.-Y. Wang, and X.-F. Xu, ''Emphasizing essential words for sentiment classification based on recurrent neural networks,'' *J. Comput. Sci. Technol.*, vol. 32, no. 4, pp. 785–795, Jul. 2017.

[46] X. Fu, W. Liu, Y. Xu, and L. Cui, ''Combine HowNet lexicon to train phrase recursive autoencoder for sentence-level sentiment analysis,'' *Neurocomputing*, vol. 241, pp. 18–27, Jun. 2017.

[47] X. Mao, S. Chang, J. Shi, F. Li, and R. Shi, ''Sentiment-aware word embedding for emotion classification,'' *Appl. Sci.*, vol. 9, no. 7, p. 1334, Mar. 2019.

[48] Y. Fang, H. Tan, and J. Zhang, ''Multi-strategy sentiment analysis of consumer reviews based on semantic fuzziness,'' *IEEE Access*, vol. 6, pp. 20625–20631, 2018.

[49] C. Fellbaum, ''WordNet: An electronic lexical resource,'' in *The Oxford Handbook of Cognitive Science*. London, U.K.: Routledge, 2017, pp. 301–314.

[50] M. Giatsoglou, M. G. Vozalis, K. Diamantaras, A. Vakali, G. Sarigiannidis, and K. C. Chatzisavvas, ''Sentiment analysis leveraging emotions and word embeddings,'' *Expert Syst. Appl.*, vol. 69, pp. 214–224, Mar. 2017.

[51] W. Li, L. Zhu, K. Guo, Y. Shi, and Y. Zheng, ''Build a tourism-specific sentiment lexicon via word2vec,'' *Ann. Data Sci.*, vol. 5, no. 1, pp. 1–7, Mar. 2018.

[52] Y. Li, Q. Pan, T. Yang, S. Wang, J. Tang, and E. Cambria, ''Learning word representations for sentiment analysis,'' *Cogn. Comput.*, vol. 9, no. 6, pp. 843–851, Dec. 2017.

[53] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, ''Deep contextualized word representations,'' Feb. 2018, *arXiv:1802.05365*. [Online]. Available: https://arxiv.org/abs/1802.05365

[54] C. Sun, L. Huang, and X. Qiu, ''Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence,'' Mar. 2019, *arXiv:1903.09588*. [Online]. Available: https://arxiv.org/abs/1903.09588

[55] Y. Chen, J. Wang, X. Chen, A. K. Sangaiah, K. Yang, and Z. Cao, ''Image super-resolution algorithm based on dual-channel convolutional neural networks,'' *Appl. Sci.*, vol. 9, no. 11, p. 2316, 2019.

[56] Y. Jung and J. Hu, ''Ak-fold averaging cross-validation procedure,'' *J. Nonparam. Statist.*, vol. 27, no. 2, pp. 167–179, 2015.

[57] L. Dora, S. Agrawal, R. Panda, and A. Abraham, ''Nested cross-validation based adaptive sparse representation algorithm and its application to pathological brain classification,'' *Expert Syst. Appl.*, vol. 114, pp. 313–321, Dec. 2018.

[58] L. Dey, S. Chakraborty, A. Biswas, B. Bose, and S. Tiwari, ''Sentiment analysis of review datasets using Naive Bayes and K-NN classifier,'' Oct. 2016, *arXiv:1610.09982*. [Online]. Available: https://arxiv.org/abs/1610.09982

[59] B. S. Lakshmi, P. S. Raj, and R. R. Vikram, ''Sentiment analysis using deep learning technique CNN with KMeans,'' *Int. J. Pure Appl. Math.*, vol. 114, no. 11, 2, pp. 47–57, 2017.

[60] B. Shin, T. Lee, and J. D. Choi, ''Lexicon integrated CNN models with attention for sentiment analysis,'' Oct. 2016, *arXiv:1610.06272*. [Online]. Available: https://arxiv.org/abs/1610.06272

[61] C. Chen, R. Zhuo, and J. Ren, ''Gated recurrent neural network with sentimental relations for sentiment classification,'' *Inf. Sci.*, vol. 502, pp. 268–278, Oct. 2019.

[62] L. Zhou and X. Bian, ''Improved text sentiment classification method based on BiGRU-attention,'' *J. Phys., Conf. Ser.*, vol. 1345, Nov. 2019, Art. no. 032097.