J Inf Process Syst, Vol.17, No.4, pp.818~833, August 2021
https://doi.org/10.3745/JIPS.04.0221

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Cross-Domain Text Sentiment Classification Method Based on the CNN-BiLSTM-TE Model

Yuyang Zeng*, Ruirui Zhang*, Liang Yang*, and Sujuan Song*

Abstract

To address the problems of low precision rate, insufficient feature extraction, and poor contextual ability in existing text sentiment analysis methods, a mixed model account of a CNN-BiLSTM-TE (convolutional neural network, bidirectional long short-term memory, and topic extraction) model was proposed. First, Chinese text data was converted into vectors through the method of transfer learning by Word2Vec. Second, local features were extracted by the CNN model. Then, contextual information was extracted by the BiLSTM neural network and the emotional tendency was obtained using softmax. Finally, topics were extracted by the term frequency-inverse document frequency and K-means. Compared with the CNN, BiLSTM, and gate recurrent unit (GRU) models, the CNN-BiLSTM-TE model's F1-score was higher than other models by 0.0147, 0.006, and 0.0052, respectively. Then compared with CNN-LSTM, LSTM-CNN, and BiLSTM-CNN models, the F1-score was higher by 0.0071, 0.0038, and 0.0049, respectively. Experimental results showed that the CNN-BiLSTM-TE model can effectively improve various indicators in application. Lastly, performed scalability verification through a takeaway dataset, which has great value in practical applications.

Keywords

Bidirectional Long Short-Term Memory, Convolutional Neural Network, Deep Learning, Sentiment Analysis, Topic Extraction

1. Introduction

According to the 45th China Internet Development Statistics Report released by the China Internet Network Information Center, the number of cyber netizens in China reached 896 million in March 2020. This represents an increase of 104 million netizens since the end of 2018, and accounts for 99.2% of the total number of netizens. People can easily complete various tasks in daily life using the Internet, such as invoice declarations and payment of phone bills. The most common of these daily life activities is online shopping. With the rise of e-commerce companies such as Taobao, people have gradually become used to shopping online. Generally speaking, online shopping is cheaper and more convenient than offline shopping. After consumers shop online, they are able to evaluate the products they have purchased, resulting in many product reviews that are visible to merchants. By reading these reviews, merchants can discover the advantages and disadvantages of commodities. Merchants can then strengthen advantages and correct the deficiencies of commodities to attract more consumers and obtain greater profits [1].

Manuscript received July 27, 2020; first revision September 24, 2020; accepted October 19, 2020.

Corresponding Author: Ruirui Zhang (zhangruiruisw@gmail.com)

* School of Business, Sichuan Agricultural University, Chengdu, China (lordsasori@163.com, zhangruiruisw@gmail.com, yl0119@outlook.com, songsujuan9695@163.com)

Other consumers can also read the reviews to evaluate whether goods are worth buying, with the aim of buying better and more satisfactory products [2]. However, with the increase of consumer sentiment reviews, it is difficult for merchants to obtain useful information by reading reviews manually. Thus, sentiment analysis using various methods has been put forward as a new field to address this issue.

Sentiment analysis is the process of analyzing, processing, summarizing, and making inferences about subjective texts with emotional trends. The Internet has enabled vast and valuable contributions from users in the form of reviews about elements such as characters, events, and products. These reviews demonstrate people's emotional trends, such as cheerfulness, anger, and sadness. By browsing subjective reviews, potential buyers are able to understand public opinion about a certain product or incident [3].

In terms of granularity of the processed text, sentiment analysis has been roughly divided into the three following levels: word, sentence, and chapter levels [4]. At present, researchers usually employ three main methods in sentiment analysis, including the sentiment dictionary method, machine learning, and deep learning.

The sentiment dictionary method employs a sentiment dictionary as a tool to estimate the emotional trend of a text [5]. The machine learning method uses mathematical classification algorithms, including support vector machine (SVM), naïve Bayes classifiers, and random forest, and techniques to extract data features and then classify the text [6]. However, in actual use, creating and updating a sentiment dictionary is a difficult task, and the effect of a sentiment dictionary method relies on the quality of the sentiment dictionary itself. The effect of machine learning methods depends on the choice of mathematical algorithm and model feature selection. However, due to their weak feature extraction ability and nonlinear fitting ability, the effect of machine learning methods is not good in practice.

Deep learning has become the most efficient method. Deep learning is a new branch of machine learning, which expresses information at multiple levels by studying the internal laws of data. Each presentation layer is composed of hidden layers containing multiple neurons. These hidden layers enable data to be more fully expressed in terms of their characteristics [7]. Traditional machine learning methods require artificial selection of features, which requires considerable human resources. However, the deep learning method automatically extracts complex and more information-rich features from the text without the need for additional manpower to select features, thereby improving the efficiency and accuracy of sentiment analysis.

In deep learning research, the convolutional neural network (CNN) is one of the most widely employed models. It has an outstanding performance in the domains of image recognition, question answering systems, and speech recognition, and is also used in the field of sentiment analysis, with the ability to extract features. However, the CNN does not consider the contextual information. Hence, the long short-term memory network (LSTM) has a time loop constructure that is specially designed to overcome this problem. In the LSTM model, information before the current time can be combined with information at the current time. Given that the LSTM model cannot obtain information after the current time, the bidirectional LSTM (BiLSTM) model was proposed; this model can combine the before and after information with the current time information to boost the result. This paper combined the CNN model, which is used to extract local features, and the BiLSTM model, which is used to extract contextual information. Then, topics were extracted using the term frequency-inverse document frequency (TF-IDF) and K-means. We processed the dataset using various models and compared the results. The CNN-BiLSTM-TE (convolutional neural network, bidirectional long short-term memory, and topic extraction) model achieved the best results and had a strong domain expansion. The extracted topics could help merchants to make improvement to products to meet the needs of consumers.

2. Related Research

Sentiment analysis is one of the most significant applications of natural language processing. Traditional research methods include sentiment dictionary and machine learning methods. With research advances, the excellent performance of deep learning in sentiment analysis has made this method increasingly popular.

The sentiment dictionary method requires certain linguistic knowledge. The quality of the sentiment dictionary method determines the effectiveness and accuracy of sentiment analysis. Commonly used emotional dictionaries include WordNet [8] and the National Taiwan University Sentiment Dictionary [9]. Turney and Pantel [10] have extended positive and negative sentiment dictionaries using the point mutual information method and polar semantic algorithms in sentiment analysis in 2010, which revealed that the accuracy in the general corpus dataset reached 74%. With the progress of society and advances in science and technology, people are becoming exposed to and familiar with a wider range of vocabularies relating to different fields. When performing sentiment analysis on texts in different fields, it is necessary to use an emotional dictionary that is specific to that field, the creation of which requires professionals with expert knowledge in that domain. However, a lack of these specialists has also become an obstacle to the development of emotional dictionaries.

Unlike the sentiment dictionary method, the machine learning method does not require a lot of resources to compile and maintain sentiment dictionaries, and there are fewer requirements for linguistic knowledge. The key step of this method is feature selection of the text data. Selecting appropriate features such as chi-squared tests and expected cross-entropy from the text data not only shortens the processing time, but also improves the performance of the classifier. Pang et al. [11] were the first researchers to use machine learning methods, which they first used in 2002 to conduct sentiment analysis trend estimates. They used three algorithms, including SVM, maximum entropy, and naïve Bayes, to extract emotional judgments from movie reviews. Tan and Zhang [12] studied four characteristic selection methods and five machine learning classifiers for Chinese text sentiment classification, and found that selecting features based on information gain and SVM classification reach a greater precision.

Methods based on deep learning construct multiple layers of neurons using a neural network that automatically extracts features to characterize and learn multi-layered characteristics. Kalchbrenner et al. [13] first proposed the application of CNN in natural language processing, and designed the dynamic CNN model to analyze texts of different lengths. Kim [14] first applied the CNN model to sentiment analysis in 2014, and found that the CNN model had a greater precision than machine learning methods. This laid the foundations for the wide application of CNN. In application, the CNN model has a demonstrated ability to extract features, but does not consider contextual information. For serialized data, it is impossible to extract contextual information, resulting in poor precision. Therefore, the recurrent neural network (RNN) model was proposed. The RNN model can extract contextual information, but as the sequence length and input information increase, a gradient explosion and gradient disappearance phenomenon occur, and the previously learned information is lost. This is also known as the long-term dependence of the RNN model.

To address this issue, Hochreiter and Schmidhuber [15] proposed the LSTM network, which has a time loop neural network structure that is specifically designed to overcome this problem. In the LSTM model, information before the current time can be obtained and combined with information at the current time. Liang et al. [16] advanced the accuracy by combining polarity transfer and the LSTM model, and used the LSTM to extract the semantic information from the emotional polarity shift model, which increases

the precision rate. However, the LSTM model cannot obtain information after the current time; to overcome this, Graves et al. [17] proposed the BiLSTM model, which can combine the before and after information with the current time information to improve the effectiveness of the model in terms of predictive ability. Zeng et al. [18] used part of speech features combined with word vectors to train the BiLSTM network to further enhance the accuracy of the model.

In summary, sentiment analysis has attracted vast attention from many researchers. Previously, researchers generally used sentiment dictionaries and machine learning methods to conduct research, but the limitations of these traditional methods mean that the results have been unsatisfactory. Thus, deep learning methods have been gradually adopted as an alternative, and neural network models such as CNN introduced into sentiment analysis tasks. In this paper, we propose the CNN-BiLSTM-TE model, which combines the CNN and BiLSTM model; the CNN model has a strong ability to extract local features from data, and the BiLSTM model can combine contextual information. We then used datasets to verify the effectiveness of our model. We also propose a method that combines TF-IDF and K-means to extract topics in product reviews, so that merchants can understand the specific feedback of consumers and make corresponding improvements to achieve a win-win situation for consumers and merchants. The main work of this paper includes the following: (1) we used the CNN to extract local features from data to reduce data size and improve processing efficiency; (2) the data processed by the CNN continue to be processed using the BiLSTM network, which allows the serialization features of the text data to be obtained through the BiLSTM network; we found that the model improves the accuracy of sentiment analysis; and (3) we used the takeaway dataset to verify the model's domain scalability, and the experimental results have reached a good level; and (4) we used the TF-IDF and K-means to extract topics from the takeaway reviews so that merchants can better understand the specific needs of consumers, which has potential practical applications.

3. The CNN-BiLSTM-TE Model

We used the CNN-BiLSTM-TE model for sentiment analysis of online reviews. The model structure is shown in Fig. 1.

The operational process included data collection, data representation, convolutional layer application, BiLSTM layer application, polarity discrimination, and keyword extraction. First, the review dataset was obtained, and reviews were preprocessed and converted into vectors; the vectorized data were then input into the convolutional layer, and after convolution and pooling operations, a new vector was obtained with local features extracted. Second, the vector was input into the BiLSTM layer to obtain contextual information. Then, the softmax layer was used to determine the polarity. Finally, the topics were extracted.

3.1 Data Processing

The online reviews of dataset were written by many consumers, had no fixed format, and contained a lot of noise. The reviews were therefore preprocessed to enhance the precision rate of the sentiment analysis. The preprocessing steps were as follows.

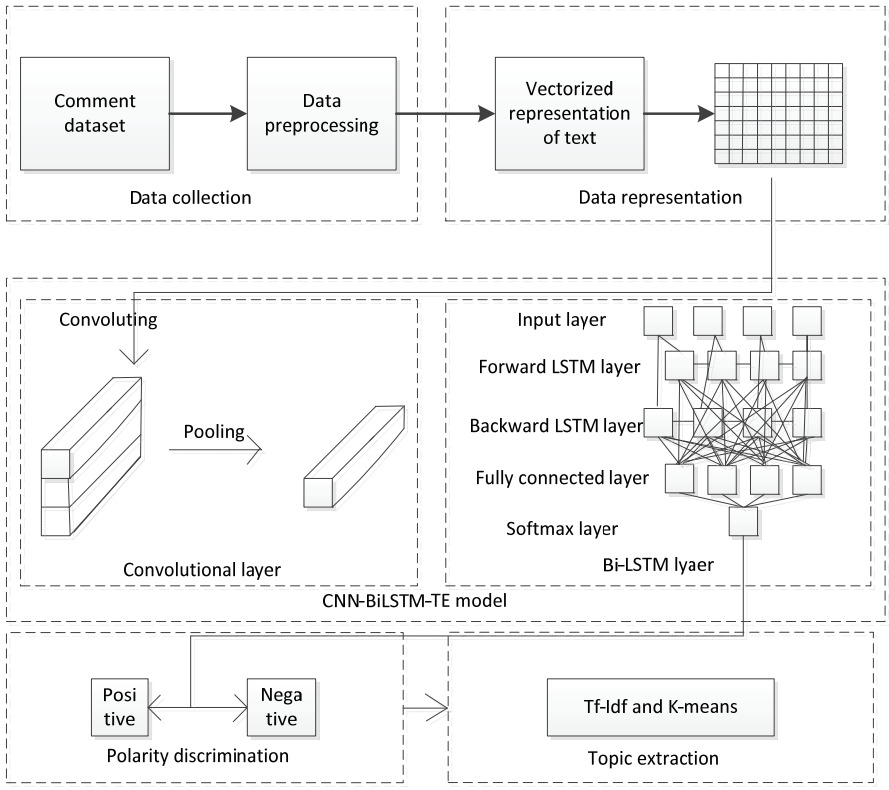

Fig. 1. Model structure diagram.

1) Text deduplication

Text deduplication refers to the removal of duplicate parts in a text dataset. Text datasets are generally obtained from reviews on various online platforms, some of which will automatically generate good reviews if users do not review within a specified time after purchasing an item. These positive reviews are meaningless for the sentiment analysis, so should be removed. Some users may also make the same reviews or copy others' reviews on different purchased products; these redundant reviews are not valuable for sentiment analysis and should also be removed.

2) Removal of stop words

The removal of stop words is the removal of connections that make writing and talking coherent in daily language. These conjunctions have no practical meaning, but frequently appear throughout the entire text data. Removing stop words not only improves processing efficiency, but also improves the accuracy of sentiment analysis.

3) Text segmentation

Text segmentation refers to the segmentation of continuous text into individual words. Word segmentation is a basic requirement of sentiment analysis. An excellent word segmentation algorithm can accurately separate words with unique meanings in the text to correctly understand the meaning of the entire text. If the text segmentation is wrong, the sentiment analysis results will be greatly biased. In the current research, we used the jieba.cut method to conduct Chinese word segmentation.

3.2 Word Vector Representation

In deep learning, word vectors are an important way to represent text. Words or phrases are mapped to real-number vectors. The words are converted into vectors of a certain dimension as the input to the neural network model. Hinton [19] first proposed the concept of distributed expression in 1986, and this concept provided the basis for the subsequent development of word vectors. Since a study by Bengio et al. [20] in 2003, the concept of distributed expression has been researched in more detail. In 2013, Google released the tool Word2Vec developed by Mikolov et al. [21] for training word vectors. They developed the neural network language model based on entire text data to efficiently transfer each single word into vectors under unsupervised learning training. The vector has tens to hundreds of dimensions, which overcomes the phenomenon of "data sparseness." The semantic distance between vectors can be calculated to find their similarity, and the "word meaning gap" problem is overcome. As an opening source tool for training word vectors, Word2Vec has continuous bag-of-words and Skip-gram structures. These two structures replace the hidden layer of the neural network model with the projection layer. This design reduces complicated calculations of the hidden layer that improve the efficiency of the model. Continuous bag-of-words and Skip-gram structures are similar, and both use the maximum likelihood function to train word vectors.

On the basis of these two structures, the word vector was updated through forward calculation of predicted probability and the backpropagation gradient. After multiplying optimization iterations, the final Word2Vec word vector was obtained.

3.3 Convolutional Neural Network

The CNN model was initially designed to process images, but also has a good performance when processing text data. The CNN construction contains the input layer, convolutional layer, pooling layer, and fully connected layer. When processing multi-dimensional inputs, the CNN decreases the number of arguments of the neural network across from the convolutional layer to the pooling layer, which prevents overfitting and reduces the complex rate of the model. Its characteristics include local connection and weight-sharing mechanisms, which can be used to extract local features from the data.

Fig. 2 illustrates the CNN model construction.

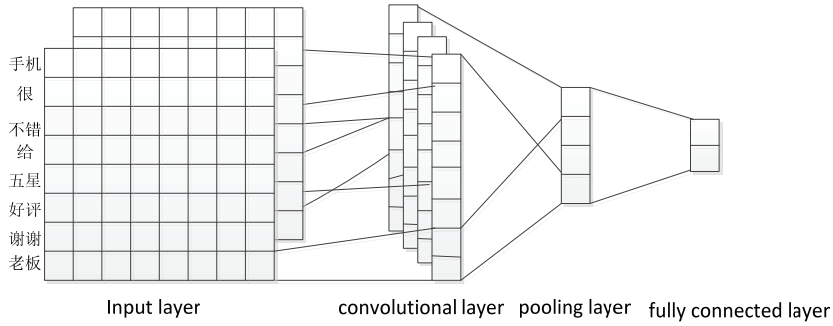

Fig. 2. Convolutional neural network.

The CNN's input is a matrix formed by converting the processed text into vectors using Word2Vec. For example, $Wn=\{w(1),w(2),\ldots\ldots,w(n\text{-}1),w(n)\}$ converts each word $w(i)$ of the review text into a word

vector $V(w(i))$, after which the converted word vector is spliced into a matrix S_{ij} using the following equation:

$$S_{ij} = \left\{V\left(w(1)\right), V\left(w(2)\right), \ldots V\left(w(n-1), V\left(w(n)\right)\right)\right\}, 1 \le i \le n \quad (1)$$

Next, the convolution layer employs a filter with a size of $r \times k$ to convolve S_{ij}. The word vector's dimension is the filter width, so that it has the ability to extract features of the input word vector using the following equation:

$$c_i = f(F \cdot V\left(W(i{:}\, i + r - 1)\right) + b) \quad (2)$$

Among these terms, f represents the ReLu activation function, $V(W(i{:}i+r\text{-}1))$ represents the r rows vector of i to $i+r$-1 in S_{ij}, F represents the filter of size $r \times k$, and c_i represents the local features gained from the convolution. The filter moves across the entire S_{ij} from top to bottom with the step size of 1. After this process, the local feature dataset C is finally created, as follows:

$$C = \{c_1, c_2 \ldots, c_{r-k+1}\} \quad (3)$$

After the convolution operation, the feature set C is still relatively large, so the local feature set C is input into the pooling layer. The pooling operation decreases the number of arguments while retaining useful information of the feature set. The pooling operations include average pooling and maximum pooling. Previous studies have shown that the maximum pooling effect is better than the average pooling effect [22]. Therefore, the maximum pooling method was used to perform the pooling operation to establish feature set C, as follows:

$$d_1 = \max C \quad (4)$$

Finally, the features after pooling were stitched in the fully connected layer to output vector D, as follows:

$$D = \{d_1, d_2 \ldots d_n\} \quad (5)$$

3.4 Bidirectional Long Short-term Memory Network

When using the RNN network for task processing, the advantage is that previous information can be used to strengthen the understanding of the current task. However, as time elapses and the current task's information length increases, and the ability to extract previous information gradually weakens, which is known as long-term dependence. The LSTM model was proposed to overcome the phenomenon of gradient explosion and disappearance in RNN. The RNN model has a simple structure and only contains an activation function. The construction of an LSTM model derived from an RNN model is relatively complex, and it results in an increased efficiency by introduction of the "gate" concept. The LSTM model has three kinds of gate: a forget gate, input gate, and output gate. Gates in the LSTM model solve the problem of long-term dependence by removing or increasing the memory function of information. In the meantime, the design of the gate can avoid gradient explosion and gradient disappearance to a certain extent, so that the LSTM can better deal with more data. The LSTM model structure is shown in Fig. 3.

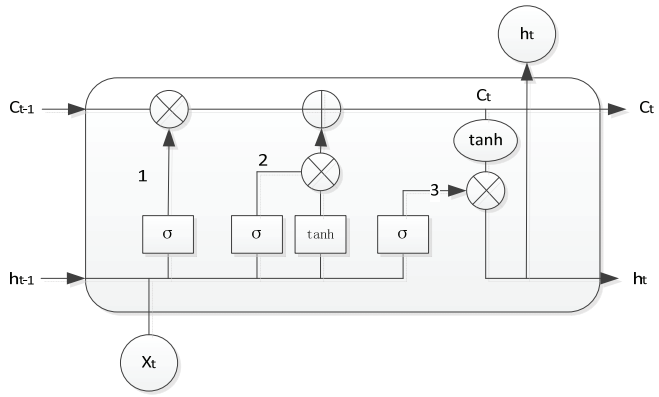

Fig. 3. Long short-term memory model.

The rectangle represents the neural network layer obtained by learning. The network layers 1, 2, and 3 represent the forget, input, and output gates, respectively. It can be seen from Fig. 3 that the hidden state h_t is obtained by x_t and h_{t-1}. On the one hand, h_t is used to count the loss of the current model. On the other hand, it is also used to calculate the h_{t+1} of the next layer. In addition to h_t, there is another hidden state that propagates forward at each time t, shown as the long horizontal line on the upper part of the neural network layer in Fig. 3, which is termed the cell state and is denoted as C_t.

The forget gate controls whether information is forgotten. At the forget gate, the previous sequence's hidden state h_{t-1} and the current sequence's data x_t are the inputs, the h_{t-1} and x_t are activated by the sigmoid function f_t, shown in the following equation:

$$f_t = \sigma(W_f \cdot h_{t-1} + U_f \cdot x_t + b_t) \tag{6}$$

where W_f and U_f are the coefficients. The biases are b_f, and σ represents the sigmoid activation function. The f_t value is between [0,1].

The input gate processes the current sequence's input and consists of two sections. The first is the input gate layer. The sigmoid activation function is used to determine what value will be updated and the resulting output. The second is the tanh layer, which creates a new candidate value C_t and adds it to the state using the following equation:

$$i_t = \sigma(W_i \cdot h_{t-1} + U_i \cdot x_t + b_i) \tag{7}$$

$$C_t = \tanh(W_c \cdot h_{t-1} + U_a \cdot x_t + b_c) \tag{8}$$

where W_i, U_i, and U_a are the coefficients, b_i and b_c are biases, and σ is the sigmoid activation function.

Then, the old cell state is updated. C_{t-1} is updated to C_t. At this time, the old state that has discarded part of the information is multiplied by f_t, the state is multiplied by the new state C_t, and the two are added to get the new candidate value, as follows:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot C_t \tag{9}$$

Finally, the final output value h_t is determined through the output gate. The part of the cell state that will be output is decided by the sigmoid layer, and then a tanh layer is used to process the cell state and multiply the output of the sigmoid layer to get the final output value, as shown in the following equations:

$$o_t = \sigma(W_O \cdot [h_{t-1}, x_t] + b_o) \tag{10}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{11}$$

However, the LSTM model has the problem of not being able to use future information. Therefore, the BiLSTM model was established. This model consists of forward LSTM and reverse LSTM, which have opposite timings and connect to the same output. The forward LSTM can obtain the previous information of the output sequence, and the reverse LSTM can obtain subsequent information of the input sequence. After stacking the BiLSTM layer, the final output is obtained [23]. The model structure is shown in Fig. 4.

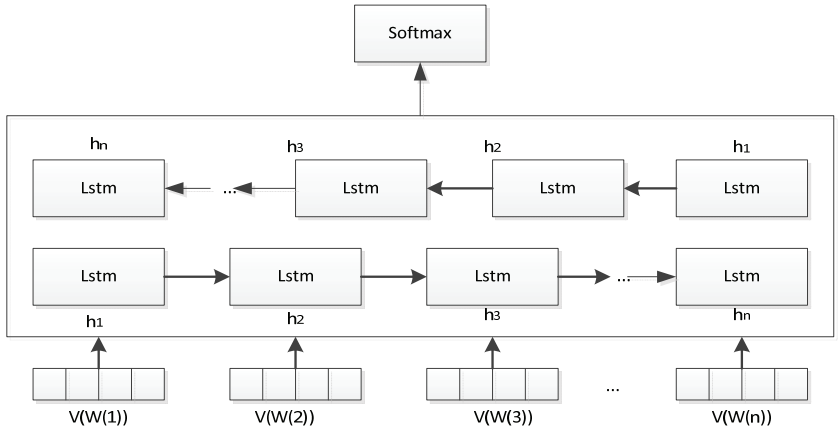

Fig. 4. Bidirectional long short-term memory model.

We input the final output into the softmax layer to obtain a value ranging from 0 to 1. When the value is 0.5 or more, the emotion trend is considered to be positive, otherwise the emotional trend is considered to be negative.

3.5 Topic Extraction

After judging the sentiment tendency of the product reviews, topic extraction of the reviews was performed using the TF-IDF and K-means, which confers application value to this research.

The TF-IDF can assess the significance of a term to a file in datasets. The more frequently a term appears in a text, and the fewer the occurrences over all documents, then the more representative the text is considered to be.

TF refers to the number of times a particular word appears in a text. This number is usually normalized so that it is not biased towards long documents (usually by using the frequency of words divided by the total number of words in the article).

The main idea of using reverse document frequency is that if there are few documents that contain the term *t*, the IDF will be larger, which means that the term has a greater ability to distinguish between categories. The IDF for a particular word can be obtained by dividing the total number of documents by the number of documents containing that word and taking the logarithm of the quotient.

High word frequency in a particular document and low word frequency throughout the document collection can result in a high-weight TF-IDF. Therefore, the TF-IDF tends to exclude common words and to retain important words.

After extracting keywords using TF-IDF, we used the K-means method to cluster the keywords. First, a K-value was randomly presented. Then, K keywords were chosen for each cluster. Finally, the Euclidean distance between every keyword was calculated with each cluster center keyword, and each keyword was appointed to the nearest cluster center keyword; each time a keyword was appointed, the cluster center was recalculated based on the existing objects in the cluster. This process is repeated until no keywords are reappointed to other classes, cluster centers are no longer changed, and the sum of squared errors is locally minimal. At this point, each category is summarized and the topic extraction work is completed.

4. Experimental Study

4.1 Experimental Data

In this experiment, the online_shopping_10_cats dataset of product reviews was used. This dataset contains consumer reviews on books, tablets, phones, fruits, shampoos, water heaters, milk, clothes, computers, and hotels. There are 62,774 items, including 31,728 positive samples and 31,046 negative samples.

The dataset was first preprocessed to remove stop words and perform word segmentation, and transfer learning was used to convert text reviews into vectors. Transfer learning referred to a pre-trained model being reused in another task to economize time and resources, and enabled the transfer of the acquired powerful skills to related problems using “Chinese-word-vectors” produced by researchers from the Chinese Information Processing Institute of Beijing Normal University and the Database & Intelligence Information Retrieval Lab of Renmin University of China [24]. The data were directly vectorized, which not only saves time and computing resources, but also makes vector features more effective. First, the words were converted into the corresponding index in “Chinese-word-vectors,” and then the vector was input into the model for vector conversion. The index used for words not included in “Chinese-word-vectors” was 0. Examples are shown in Table 1.

Table 1. Review examples and word segmentation results

Serial#	Category	Reviews	Reviews after word segmentation	Vector index
1	Book	作者逻辑严密，一气呵成 ° 没有一句废话，深入浅出，循循善诱，环环相扣	作者\逻辑\严密\一气呵成\没有\一句\废话\深入浅出\循循善诱\环环相扣	[944, 1020, 12069, 18996, 29, 0, 5696, 28233, 49202, 39218]
2	Phone	手机很不错，棒棒的，给五星好评，谢谢老板祝你生意兴隆	手机\很\不错\棒棒\的\给\五星\好评\谢谢\老板\祝\你\生意兴隆	[300, 34, 562, 11144, 1, 51, 14000, 7779, 1478, 751, 2908, 42, 102081]
3	Fruit	从北京发货的，包装不错，物流也快，吃起来味道挺好的	从\北京\发货\的\包装\不错\物流\也\快\吃\起来\味道\挺\好\的	[82, 497, 14713, 1, 3438, 562, 7521, 18, 488, 116, 207, 1451, 470, 72, 1]
4	Milk	蒙牛也有点儿太好看了，不冲这包装我肯定不带买的	蒙牛\也\有点儿\太\好看\了\不冲\这\包装\我\肯定\不\带\买\的	[26091, 18, 8520, 163, 772, 3, 0, 22, 3438, 6, 368, 0, 200, 1]
5	Hotel	这个酒店给人感觉不错，位置也比较好，硬件也不错，性价比较高	这个\酒店\给\人\感觉\不错\位置\也\比较\好\硬件\也\不错\性价比\较\高	[36, 1845, 51, 12, 176, 562, 949, 18, 169, 72, 4652, 18, 562, 4429, 3159, 236]

Next, the entire dataset, of which the training set accounted for 80%, was divided by the total of 50,219 items. The model set a 10-fold cross-validation, so used 45,197 reviews for training and the other 5,022 reviews for cross-validation. Then, the effect on a test dataset that included 12,555 reviews was measured. After obtaining the experimental results, the takeaway dataset was tested to verify the model's scalability. This dataset has a total of 8,000 reviews, including 4,000 positive reviews and 4,000 negative reviews.

4.2 Experimental Indicators

In this experiment, the indicators used to assess the effect of the classifiers were precision, recall, and the F1-score. As a visualization tool, the confusion matrix can be a good interpretation of these indicators, as shown in Table 2.

Table 2. Confusion matrix

		Forecast result	
		Positive	**Negative**
Real result	Positive	TP	FN
	Negative	FP	TN

If an item is a positive class and it is predicted to become a positive class, it is a true positive (TP). If an item is a negative class and it is predicted to be a negative class, it is a true negative class (TN). If an item is a negative class but it is predicted to be a positive class, it is a false positive class (FP). If an item is a positive class but it is predicted to be a negative class, it is a false negative class (FN).

The formula of precision, recall, and the F1-score can be expressed as follows:

$$precision = \frac{TP}{TP+FP} \tag{12}$$

$$recall = \frac{TP}{TP+FN} \tag{13}$$

$$F1 = \frac{2*\text{precision}*\text{recall}}{\text{precision}+\text{recall}} \tag{14}$$

4.3 Parameter Selection

First, analysis of datasets revealed that 90% of the text data was less than 72 words in length. To ensure the integrity of information and the efficiency of processing, the text length was set as 72. When the text length was less than 72 words, we used 0 to fill the void of word vectors. When the length of the text was greater than 72 words, we kept only the first 72 words. The setting of the experimental parameters has a great impact on the test results. After many comparison tests, the model effect was improved with the parameter settings shown in Table 3.

We introduced the early-stopping mechanism, and set the epoch period to 20 and the patience value to 3; that is, after 1 iteration, if the loss was not optimized, the learning rate was reduced and repeated iteratively. If the loss was not optimized after 3 iterations, the iteration was terminated, and the result of this iteration regarded as the final result; there were up to 20 iterations.

To prevent the occurrence of over-fitting, the Dropout mechanism [25] was introduced to delete a certain proportion of information in each iteration and replace it with 0, which gives the model a better

generalization ability. In the experiment, the Dropout layer was put on the CNN layer and BiLSTM layer. As shown in Fig. 5, different Dropout values have imparity impacts on the test consequences. When the Dropout value was 0.7, the F1-score was highest.

Table 3. Model parameter settings

Parameter	Value
Text length	72
Word vector dimension	300
Loss	Categorical_crossentropy
Optimizer	Adam
Cross-validation	0.1
Learning rate	0.001
CNN layer activation	Relu
Bi-LSTM layer activation	Sigmoid
Bi-LSTM layer Number of neurons	64
Batch size	64

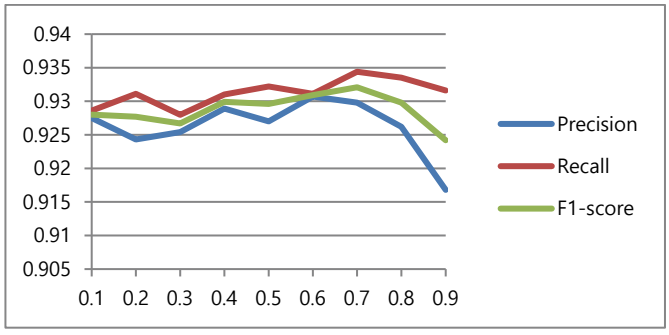

Fig. 5. Dropout curve.

The height of the filter is another factor that affects the model effect (Fig. 6). When the height is set too low, the features obtained by convolution are too few. When the height is set too high, it will lead to over-fitting. Different height values affect the test consequences. When the filter height is set to 3, the F1-score was highest.

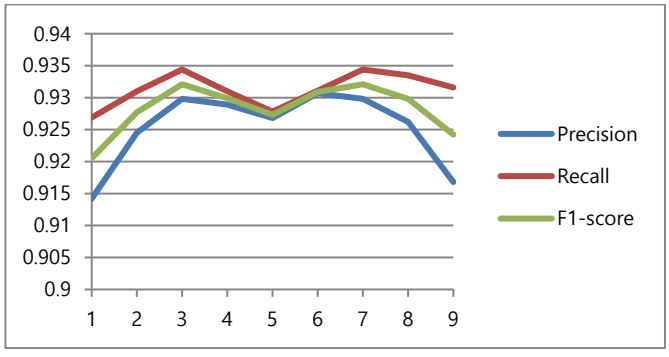

Fig. 6. Filter height curve.

4.4 Model Performance Comparison

To verify the effectiveness of the CNN-BiLSTM-TE model, single models such as the CNN, BiLSTM, and GRU [26] were constructed for comparison. The test results are shown in Table 4.

Table 4. Comparison of the proposed model and single models in performance

Model	Early-stopping	Precision (%)	Recall (%)	F1-score
CNN	6	92.80	91.38	0.9209
BiLSTM	7	93.37	92.55	0.9296
GRU	9	93.10	92.97	0.9304
CNN-BiLSTM-TE	4	93.73	93.39	0.9356

The CNN-BiLSTM-TE model achieved the highest values in all indicators. Compared with the CNN, BiLSTM, and GRU models, the precision rate was higher than other models by 0.93%, 0.36%, and 0.63%, respectively. The recall rate was higher than other models by 2.01%, 0.84%, and 0.42%, respectively. The F1-score was higher than other models by 0.0147, 0.006, and 0.0052, respectively. The early stopping periods of each model were 6, 7, 9, and 4, respectively. The CNN-BiLSTM-TE model completes the experiment in a short iterative period, which verifies its efficiency. The reason for this short iterative period is that the CNN model only extracts local features, and does not link these with contextual information. The BiLSTM model only extracts contextual information, but cannot extract local features as the CNN model does. As an improvement of the LSTM model, the GRU model optimizes the model structure, but can only extract the above information and cannot combine the following information. The CNN-BiLSTM-TE model combines the CNN model's forceful capacity to extract local features with the BiLSTM model ability to combine the characteristics of contextual information; the resulting effect is significantly more valid than the three single models independently.

To further verify the model's domain scalability, the takeaway dataset was used to test the model. Its precision, recall, and F1-scores reached 87.07%, 84.05%, and 0.8553, respectively. The TF-IDF and K-means were used to extract topics from the reviews. First, we used the TF-IDF to extract the first 30 keywords. After many experiments, we found that topic extraction was best when k was 3. The results of topic extraction are shown in Table 5.

Table 5. Topic extraction results

Category	Key words
1	小时(hour), 速度(speed), 太慢, 很快, 时间, 分钟, 下次
2	好吃(delicious), 味道, 不错, 难吃(unpalatable), 特别, 态度, 服务, 卷饼, 百度, 米饭, 辛苦, 超级, 感觉, 真的
3	送到, 送来, 外卖, 配送, 快递, 小哥(delivery man), 包装(packaging), 东西, 师傅

According to keywords such as hour and speed, the theme of category 1 was summarized as the delivery speed. According to keywords such as delicious, taste, and unpalatable, the theme of category 2 was summarized as food quality. According key words such as packaging, delivery man, the theme of category 3 was summarized as service quality. These results suggest that merchants should be most concerned with the delivery speed, food quality, and service quality of the take-out, and should optimize these three aspects to meet the consumers' needs.

In summary, firstly, the CNN-BiLSTM-TE model fully extracts the local features of the reviews without overfitting. Second, it combines contextual information to achieve better results. Finally, it extracts topics through the TF-IDF and K-means, which can help merchants save the cost of manual judgment, and encourage merchants to continuously improve service and product quality to bring consumers a better experience.

4.5 Comparison of Similar Related Work

In the same experimental environment, for comparison, we built the CNN-LSTM model, LSTM-CNN model, and BiLSTM-CNN serial feature fusion model, as follows [27]:

1) The CNN-LSTM model: First, the model abstracts the local features of the reviews through the CNN. Then, it uses the LSTM to abstract the above information. Finally, the softmax layer is applied for sentiment judgment.
2) The LSTM-CNN model: First, the LSTM model is used to extract the above information. Then the CNN is applied to abstract local features of the reviews. Finally, the softmax layer is entered for sentiment judgment.
3) The BiLSTM-CNN serial feature fusion model: First, the BiLSTM is used to abstract contextual information. Then, the CNN is used to extract local features of reviews, and finally enter the softmax layer for sentiment judgment.

These three models were used to process the online_shopping_10_cats dataset, and the results compared with those of the CNN-BiLSTM-TE model. The comparison results are shown in Table 6.

Table 6. Comparison of the proposed model and hybrid models in performance

Model	Precision (%)	Recall (%)	F1-score
LSTM-CNN	92.74	92.96	0.9285
CNN-LSTM	93.37	92.99	0.9318
BiLSTM-CNN	93.01	93.05	0.9307
CNN-BiLSTM-TE	93.73	93.39	0.9356

The CNN-BiLSTM-TE model had significantly better precision, recall, and F1-scores than the LSTM-CNN, CNN-LSTM, and BiLSTM-CNN models. Compared with these three hybrid models, the F1-score was higher than the other models by 0.0071, 0.0038, and 0.0049, respectively. This is because the LSTM-CNN and CNN-LSTM models ignore the following information, resulting in a poor model effect. Additionally, the CNN-BiLSTM-TE model has a greater capabilities than the BiLSTM-CNN model. Considering that the BiLSTM model is employed to abstract the context characteristics first, it not only takes a long time, but also deteriorates the effect of the CNN model. The CNN-BiLSTM-TE model first uses the CNN for local feature extraction, which saves time and resources, and also achieves better results.

5. Conclusion

This paper proposes the CNN-BiLSTM-TE model for text sentiment analysis. This model combines the CNN and BiLSTM models. First, the CNN is used to abstract the local features of the datasets. After that, the BiLSTM is combined with contextual information for sentiment judgment. Finally, the TF-IDF and K-means are used to extract text topics. The validity of the model was verified by training and testing on online review datasets containing 10 fields. The results demonstrate that the CNN-BiLSTM-TE model

achieves superior values in all indicators. Compared with the related models, and has a better performance in analysis tasks. Then, we used the takeaway dataset to verify its domain scalability. The F1-score reached 0.85, which indicates a good domain scalability, and extracts review topics through the TF-IDF and K-means, which could help to guide business decisions.

References

[1] J. Li and H. Li, "Research on product feature extraction and sentiment classification of short online review based on deep learning," *Chinese Journal of Information Studies: Theory & Application*, vol. 41, no. 2, pp. 143-148, 2018.

[2] J. Qiu, C. Liu, Y. Li, and Z. Lin, "Leveraging sentiment analysis at the aspects level to predict ratings of reviews," *Information Sciences*, vol. 451, pp. 295-309, 2018.

[3] H. Tang, S. Tan, and X. Cheng, "A survey on sentiment detection of reviews," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10760-10773, 2009.

[4] J. F. Xu, Y. Xu, Y. C. Xu, Y. Zhang, and Q. Liu, "Hybrid algorithm framework for sentiment classification of chinese based on semantic comprehension and machine learning," *Computer Science*, vol. 42, no. 6, pp. 61-66, 2015.

[5] A. Joshi, A. R. Balamurali, P. Bhattacharyya, and R. Mohanty, "C-Feel-It: a sentiment analyzer for micro-blogs," in *Proceedings of the ACL-HLT 2011 System Demonstrations*, Portland, OR, 2011, pp. 127-132.

[6] E. Boiy and M. F. Moens, "A machine learning approach to sentiment analysis in multilingual web texts," *Information Retrieval*, vol. 12, no. 5, pp. 526-558, 2009.

[7] Y. X. He, S. T. Sun, F. F. Niu, and F. Li, "A deep learning model enhanced with emotion semantics for microblog sentiment analysis," *Chinese Journal of Computers*, vol. 40, no. 4, pp. 773-790, 2017.

[8] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM*, vol. 38, no. 11, pp. 39-41, 1995.

[9] L. W. Ku and H. H. Chen, "Mining opinions from the web: beyond relevance retrieval," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 12, pp. 1838-1850, 2007.

[10] P. D. Turney and P. Pantel, "From frequency to meaning: vector space models of semantics," *Journal of Artificial Intelligence Research*, vol. 37, pp. 141-188, 2010.

[11] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Philadelphia, PA, 2002, pp. 79-86.

[12] S. Tan and J. Zhang, "An empirical study of sentiment analysis for Chinese documents," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2622-2629, 2008.

[13] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL)*, Baltimore, MD, 2014, pp. 655-665.

[14] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, 2014, pp. 1746-1751.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[16] J. Liang, Y. Chai, H. Yuan, M. L. Gao, and H. Zan, "Polarity shifting and LSTM based recursive networks for sentiment analysis," *Journal of Chinese Information Processing*, vol. 29, no. 5, pp. 152-159, 2015.

[17] A. Graves, N. Jaitly, and A. R. Mohamed, "Hybrid speech recognition with deep bidirectional LSTM," in *Proceedings of 2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouc, Czech Republic, 2013, pp. 273-278.

[18] S. F. Zeng, X. Y. Zhang, X. F. Du, and T. B. Lu, "New method of text representation model based on neural network," *Journal on Communications*, vol. 38, no. 4, pp. 86-98, 2017.
[19] G. E. Hinton, "Learning distributed representations of concepts," in *Proceedings of the 8th Annual Conference of the Cognitive Science Society*, Amherst, MA, 1986.
[20] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *The Journal of Machine Learning Research*, vol. 3, pp. 1137-1155, 2003.
[21] T. Mikolov, K Chen, G Corrado, and J. Dean. "Efficient estimation of word representations in vector space," 2012 [Online]. Available: https://arxiv.org/abs/1301.3781.
[22] X. D. Guo, N. Zhao, and S. Z. Cui, "Consumer reviews sentiment analysis based on CNN-BiLSTM," *Systems Engineering-Theory & Practice*, vol. 40, pp. 653-663, 2020.
[23] S. P. Zhai and Y. Y. Yang, "Bilingual text sentiment analysis based on attention mechanism Bi-LSTM," *Computer Applications and Software*, vol. 36, no. 12, pp. 251-255, 2019.
[24] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14-23, 2015.
[25] G. E. Hinton, N. Srivastava, and A. Krizhevsky, "Improving neural networks by preventing co-adaption of feature detectors," 2012 [Online]. Available: https://arxiv.org/abs/1207.0580.
[26] Y. Zhang, Y. Jiang, and Y. Tong, "Study of sentiment classification for Chinese microblog based on recurrent neural network," *Chinese Journal of Electronics*, vol. 25, no. 4, pp. 601-607, 2016.
[27] H. Zhao, L. Wang, and W. Wang, "Text sentiment analysis based on serial hybrid model of bi-directional long short-term memory and convolutional neural network," *Journal of Computer Applications*, vol. 40, no. 1, pp. 16-20, 2020.

Yuyang Zeng **https://orcid.org/0000-0002-6460-9363**

He received B.S. degrees in Department of Information Management from Chengdu Neusoft University in 2019. He is currently a M.S. candidate in Business School from Sichuan Agricultural University, Chengdu, China. His current research interests include natural language processing and deep learning.

Ruirui Zhang **https://orcid.org/0000-0003-1898-1487**

She received B.S., M.S., and Ph.D. degrees in School of Computer Science from Sichuan University in 2004, 2007, and 2012, respectively. Her current research interests contain artificial immune systems, intrusion detection and wireless sensor networks.

Liang Yang **https://orcid.org/0000-0003-0250-4707**

He received B.S. degrees in School of Business from Sichuan Agricultural University in 2019. He is currently a M.S. candidate in Business School, Sichuan Agricultural University, Chengdu, China. His current research interests include business intelligence and data mining systems.

Sujuan Song **https://orcid.org/0000-0003-3364-3005**

She received B.S. degrees in School of mathematics and Information Science from Henan Normal University in 2019. She is currently a M.S. candidate in Business School, Sichuan Agricultural University, Chengdu, China. Her research fields include business intelligence and data mining.