Research on sentiment classification of Chinese short text

ZHICHAO ZHU1*, ZUI ZHU2, WENJUN ZHU3

1College of Computer Beijing University Of Technology Beijing, China
2College of Computer Beijing University Of Technology Beijing, China
3College of Computer Beijing University Of Technology Beijing, China

* Corresponding author: Zhuzc@emails.bjut.edu.cn

Abstract—Sentiment classification can be widely used in various fields because of the value it has gained, and has received extensive attention in recent years. No matter the traditional methods based on language rules, machine statistical learning, or the current popular deep learning have achieved great success in this respect, but up to now, because of the particularity of Chinese, there are still few methods for Chinese text sentiment classification. Moreover, the traditional method is time-consuming and labor-consuming, with poor versatility. The neural network method saves resources and is generally applicable, but the application of language knowledge is not sufficient. For this reason, we propose a method combining traditional method with neural network, which combines sentiment lexicon, attention mechanism and stacked BILSTM. Word vectors are generated using word2vec, and the context semantics are captured multiple times by stacked BILSTM to get richer semantic information, for sentiment classification, sentiment words often have more influence on the result of classification, so we use sentiment lexicon which contains rich sentiment word to monitor the attention mechanism, and make the attention model pay more attention to sentiment words. Combining the advantages of traditional methods and neural networks, we use the proposed method to do the sentiment binary classification experiments on two data sets. The experimental results show that our proposed method is superior to the previous method.

1. INTRODUCTION

In recent years, with the rapid development of Internet and social network, the network not only facilitates people's life, but also has a profound impact on various industries. More and more people are beginning to express their opinions on the website, and big data of user reviews emerges spontaneously, for example, Taobao, Jingdong, hotel reviews and so on. Sentiment classification is one of the directions of text classification, developed from text classification. The explosion of big data, text data occupies the vast majority. Sentiment classification refers to the extraction of text sentiment tendencies from a large amount of unprocessed text data. Various companies and enterprises try to analyze users' behavior through a large number of comment text data generated by users, mining deeper knowledge hidden under the text data to make decisions, find business value, and provide more convenient and humanized products.

Sentiment classification is one of the basic tasks of natural language processing (NLP). Its purpose is to divide a sentence into a certain emotional polarity (positive, negative or more fine-grained category). In e-commerce, government affairs, information prediction and other fields, it can understand users' interest preferences, familiar with the public sentiment and public opinion, and predict
the future trend of the financial and stock markets. Therefore, in practical applications, sentiment classification plays an essential role and has great research value.

1.1. Motivation
In recent years, rules and lexicons based, machine learning and neural network based methods have achieved good results in sentiment classification, especially neural network model, it breaks the limitations of the traditional rule-based and dictionary based methods, and has stronger versatility. Compared with machine learning methods, it has less dependence on corpus. Therefore, the neural network model has made outstanding progress in the research of sentiment classification. But at present, for text sentiment classification, there may be more researches on English texts, and Chinese research is relatively weak, and due to the particularity of Chinese, the sentiment classification methods applicable to English may not be applicable to Chinese. Therefore, there is still a lot of research space for sentiment classification of Chinese text, and although neural network model has made outstanding achievements in sentiment classification. But there are still many deficiencies, mainly reflected in two aspects. On the one hand, it is mainly the current training method of neural networks. Most of them are to train the optimized model by directly inputting the training data and let the neural network learn autonomously. Although the attention mechanism appeared later to allow the model to pay attention to certain important words as much as possible, the words that the attention model noticed may not be the most critical. For sentiment classification, sentiment words are obviously essential. As an important reference for sentiment analysis in traditional methods, it is obviously reasonable to introduce sentiment dictionary into deep learning model as the extra information assistant. On the other hand, simply looking at sentimental words to determine the overall emotional tendency of a text is obviously not enough, a text may have sentimental words with completely opposite polarities. For example, a comment: “物流速度虽然让人失望，但是老板的服务态度还是非常让人满意的!”(English translation: “Although the logistics speed is disappointing, the boss's service attitude is still very satisfying!”) The main sentimental vocabulary includes “失望” (English translation: “disappointing”) and “满意”(English translation: “satisfying”). If only referring to the emotional vocabulary, “失望”(English translation: “disappointing”) and “满意”(English translation: “satisfying”) is completely opposite. It is difficult for us to distinguish emotional tendencies. Therefore, it is very important to analyze the overall sentimental tendency of the whole text by referring to the context. As far as our human brain distinguishes it, it may not be able to distinguish the true tendency of sentiment once. It takes repeated thinking to truly appreciate the true expression of sentiment. Similarly, for deep learning models, simply capturing the context once may not be enough, so it is obviously reasonable to increase the number of context feature extractions.

1.2. Contribution
In order to solve this problem, we propose a model based on the combination of sentiment lexicon and neural network. The sentiment lexicon is jointly stacked with a bidirectional long short-term memory model and attention model, which aims to classify Chinese texts into two categories (positive or negative) and further improve the accuracy of sentiment classification. Our main contributions in this article are as follows:

- As far as we know, we are the first to combine a compound model (stacked bidirectional long short-term memory model) with attention model and sentiment lexicon to perform Chinese sentiment classification task. In addition, we achieved good results on two Chinese sentiment classification data sets.
- We proved that the composite model (stacked bidirectional long short-term memory model) can further extract rich semantic features to improve the accuracy of the model, but it also has limits.
- For sentiment classification, using sentiment lexicon to supervise attention model can significantly improve the accuracy.
- Through experiments, we found that our model is superior to other baseline models.
2. RELATED WORK
At present, in the field of sentiment classification, the main research methods at home and abroad can be divided into three categories: rules and lexicon-based methods, statistical machine learning-based methods and neural network-based methods.

2.1. Traditional Method
Traditional methods can be divided into two categories: rules-based and lexicon-based methods and methods based on statistical machine learning.

The method based on the sentiment dictionary mainly uses a sentiment dictionary with sentimental polarity and intensity as an important reference for judging the emotional orientation of words. In order to improve the effectiveness of the sentiment dictionary-based method, a large amount of sentimental vocabulary needs to be collected to expand the sentiment dictionary. In particular, the work in [1] built a microblog-specific Chinese sentiment lexicon based on existing lexicons for subjectivity detection and sentiment polarity classification. Zhang et al. [2] combined degree adverb dictionary, negative word dictionary, network word dictionary and other related dictionaries to further extend the sentiment dictionary.

The method based on rules and lexicons is the earliest method used in sentiment classification. They mainly use sentiment lexicons, and many rules are hand-made. Each rule is given different weight, it is a sentiment classification system that depends on rules [1]-[11]. In case of rule conflict, the rule with the highest weight is used to identify the emotion category. In general, when the extracted rules can accurately classify emotion, the performance of rule-based method is better than that of statistical learning method. However, these rules have great limitations. They usually rely on a fixed language, domain, and text style. It is difficult to apply to all language phenomena, error-prone and poor portability. For different styles of texts, linguistic experts need to rewrite the rules. In a word, the disadvantages of rule-based and lexicon based methods are poor portability and the need to establish knowledge base of different languages, fields and text styles to help improve the system recognition ability.

Methods based on statistical machine learning are also widely used in sentiment classification, mainly including: Logistic Regression (LR [12]), Support Vector Machine (SVM [13][14]), Naive Bayes (NB [15][16]), etc. Take SVM for example, SVM is a statistical model used for outlier detection, text classification or regression. SVM is a non-probabilistic linear classifier. Suppose the goal is to classify some given data points in one of two classes. SVM separates the data points on the high-dimensional space by finding hyper planes or hyper plane sets, so that the optimal hyper plane represents the larger margin between the two classes. Researchers have widely used this classifier in the field of sentiment analysis. Although machine learning methods have a good classification effect in sentiment classification, it has a great dependence on corpus, and needs a lot of external resources or manually configured features.

2.2. Neural Network Method
In recent years, with the development of hardware capabilities and the emergence of word distributed representation, neural network has become a model that can effectively handle many NLP tasks. At present, many neural networks have been proposed for sentiment classification.

For example, Convolutional Neural Network (CNN [17][18]) is widely used for sentiment classification. Recurrent neural networks (RNN [19]), because of their ability to model context before and after, are also very common when learning sentence-level expressions. Since RNN will encounter the problem of gradient disappearance, some scholars have proposed a long-short-term memory model (LSTM [20]), which solves the problem of RNN gradient disappearance through various gating units. LSTM can be applied to sequence data, tree structure data [21], and its extended bidirectional LSTM (BiLSTM [22]). LSTM can only capture context information from the forward direction, whereas BiLSTM can capture both forward and back directions at the same time, providing richer semantic information. Stacked BiLSTM was proposed by Zhou et al. in 2019. They believe that feature
extraction of high-dimensional word vectors in sequences may be quite complex, and one extraction may not be enough to complete. It has been proved that the accuracy of two extractions is better than that of single extraction BILSTM. Many researchers have combined CNN and LSTM into a model, which has achieved good classification results [23][24]. Attention mechanism methods are also widely used in sentiment classification. The earliest introduction of attention models into the field of natural language processing was Bahdanau et al. [25]. Attention mechanism was first widely used in image recognition and other fields. Attention mechanism can be regarded as simulating human brain, which will automatically screen out important information from the data and focus attention on these important information. The main idea is also applicable to text classification. Different words have different influences on sentence classification. Words with greater impact on the classification result should be paid more attention to, increase the weight of these words. By assigning attention weights to improve the accuracy of classification, Yang et al. [26] achieved a good text classification effect by adding an attention mechanism, and the principle of sentiment classification is the same.

The traditional sentiment classification method and the currently popular neural network sentiment classification method have their own advantages and disadvantages. The traditional sentiment classification method and the currently popular neural network sentiment classification method have their own advantages and disadvantages. In contrast, the neural network method reduces human resources and is more versatile, but the application of language knowledge is insufficient. Therefore, we propose a method that uses sentiment lexicons, stacked BILSTM, and attention models to perform sentiment analysis on Chinese text. The advantage of our model is to perfect the defects of traditional methods and neural network methods, and combine their respective advantages. The application of language knowledge is comprehensive, independent of human resources, universal and applicable.

3. METHODOLOGY

In this section, we will analyze and introduce our proposed method in detail. Fig. 1 shows a method for sentiment classification of Chinese text.

In our model, we combine the traditional emotion lexicon with the popular attention mechanism and stacked BILSTM. The advantages of this method are that the neural network model makes up for the time-consuming and laborious shortage of the traditional emotion classification method, and the traditional emotion lexicon method satisfies the deficiency of the neural network in the application of language knowledge, and for single model, the multi-model fusion method combines the advantages of different models, classification accuracy is higher, more universal.

In our model, it's divided into three parts. First, for a data set that contains many sentences, we perform word segmentation processing on each sentence to get the corresponding word list. Secondly,
the obtained word list is sent to the trained word2vec model to generate word vectors, and stacked BILSTM performs two semantic captures of the word vectors in the positive and negative directions to obtain a hidden layer output containing rich semantic information. The attention model adds different attention to the hidden output according to the importance of the word. Under the supervision of the sentiment dictionary, the attention model focuses on the sentiment words as much as possible to get the weighted output. Finally, the weighted output is put into the softmax classifier to obtain the final sentiment classification result.

3.1. Word Preparation
Since there is no clear delimiter in the Chinese text to split the sentence into words, we used a popular Chinese word segmentation tool named Jieba to split Chinese comments, it is a library of Python. Table I shows two examples after the segmentation of the Jieba library.

| Processed Text                                                                 | Segmented Text                                                                 |
|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------|
| 餐厅很差，菜的种类和水准都不行！ (English translation: “The restaurant is bad, the variety and standard of dishes are terrible!”) | 餐厅/很/差/，/菜/的/种类/和/水准/都/不行/！ |
| 房间很旧，有一股糟糕的味道！ (English translation: “The room is very old and has a bad smell!”) | 房间/很/旧/，/有/一股/糟糕/的/味道/！ |

3.2. Embedding Layer
We use the word2vec model to obtain the word vector representation. The word2vec model includes the CBOW model and the Skip-gram model. Both the CBOW model and the Skip-gram model include an input layer, an inference layer, and an output layer. The core idea of the CBOW model is to predict the target word according to the context distribution, whereas the Skip-gram model is to predict the context based on the target word. In this experiment, we used the CBOW model. In order to train the model, we set the window size to 5 words, and obtain the context of the current word from the segmented comments within the 5-word limit, and then send the extracted context to the CBOW model, and the current word becomes the feature to be predicted by the model. After training, we set the word vector dimension to 300.
3.3. Text Representation

The text representation model obtains text features based on contextual information. We use a two-layer stacked BILSTM model to obtain the textual representation of each comment. Stacked BILSTM has the same principle as BILSTM, rich content information can be obtained in the past and the future. But different from BILSTM, BILSTM has only one hidden layer extraction in each direction, whereas stacked BILSTM has more layers in each direction for further extraction, and each direction will continue to extract further in the hidden layer state after the first extraction, the advantage of this is that it can obtain richer context information, and the effect improvement is more obvious than the single-layer BILSTM.

The principle of stacked BILSTM is shown in Fig. 2 (2 hidden layers). The initial sequence \([x_1, x_2, x_3, ..., x_t, ..., x_k]\) generated from word2vec pass through the first layer of positive and negative LSTM to obtain two sequences \([a_1, a_2, a_3, ..., a_t, ..., a_k]\) and \([c_1, c_2, c_3, ..., c_t, ..., c_k]\), these two sequences are then obtained \([b_1, b_2, b_3, ..., b_t, ..., b_k]\) and \([d_1, d_2, d_3, ..., d_t, ..., d_k]\) respectively through the second layer of positive and negative LSTM, then \([b_1, b_2, b_3, ..., b_t, ..., b_k]\) and \([d_1, d_2, d_3, ..., d_t, ..., d_k]\) are spliced at the corresponding positions to generate the final hidden state output \([O_1, O_2, O_3, ..., O_t, ..., O_k]\) of the stacked BILSTM.

In Fig. 2, each hidden layer node (such as \(a_t\)) represents a LSTM cell, which contains a “forgetting gate” \(f_t\), an “input gate” \(i_t\), and an “output gate” \(o_t\), a “temporary cell state” \(C_t\). The working principle of LSTM is to first decide what information is obtained before discarding, and it is completed by the “forgetting door”. Its input is the output of the last time and the input of the current time. The output is a number between 0 and 1, 0 represents all abandonment, 1 represents all reservations; then decides what kind of information should be stored. This process is mainly divided into two steps. First, the sigmoid function of the “input gate” determines which values we need to update, and it is recorded as \(i_t\). Then, the tanh function generates a new candidate vector , which can be added to the state \(C_t\). Next, we can update the state of neurons, multiply the old state with \(f_t\), forget what we want to forget before, and then add \(C_t\) to get the new state \(C_t\). Finally, we need to determine the output content. First, use the sigmoid function to determine which part of the neuron output is recorded as \(o_t\), then use the tanh
function to keep the output value between -1 and 1, and then multiply it by the sigmoid output value to get the final output results $h_t$.

For the first forward layer, the calculation process of hidden state $a_t$ is as follows:

$$f_t = \sigma(W_f^a \cdot [a_{t-1}, x_t] + b_f^a),$$  \hspace{1cm} (1)

$$i_t = \sigma(W_i^a \cdot [a_{t-1}, x_t] + b_i^a),$$  \hspace{1cm} (2)

$$\tilde{C}_t = \tanh(W_c^a \cdot [a_{t-1}, x_t] + b_c^a),$$  \hspace{1cm} (3)

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t,$$  \hspace{1cm} (4)

$$O_t = \sigma(W_o^a \cdot [a_{t-1}, x_t] + b_o^a),$$  \hspace{1cm} (5)

$$a_t = O_t \odot \tanh(C_t^a).$$  \hspace{1cm} (6)

For the second forward layer, the calculation process of hidden state $b_t$ is as follows:

$$f_t = \sigma(W_f^b \cdot [b_{t-1}, a_t] + b_f^b),$$  \hspace{1cm} (7)

$$i_t = \sigma(W_i^b \cdot [b_{t-1}, a_t] + b_i^b),$$  \hspace{1cm} (8)

$$\tilde{C}_t = \tanh(W_c^b \cdot [b_{t-1}, a_t] + b_c^b),$$  \hspace{1cm} (9)

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t,$$  \hspace{1cm} (10)

$$O_t = \sigma(W_o^b \cdot [b_{t-1}, a_t] + b_o^b),$$  \hspace{1cm} (11)

$$b_t = O_t \odot \tanh(C_t^b).$$  \hspace{1cm} (12)

For the first backward layer, the calculation process of hidden state $c_t$ is as follows:

$$f_t = \sigma(W_f^c \cdot [c_{t+1}, x_t] + b_f^c),$$  \hspace{1cm} (13)

$$i_t = \sigma(W_i^c \cdot [c_{t+1}, x_t] + b_i^c),$$  \hspace{1cm} (14)

$$\tilde{C}_t = \tanh(W_c^c \cdot [c_{t+1}, x_t] + b_c^c),$$  \hspace{1cm} (15)

$$C_t = f_t \odot C_{t+1} + i_t \odot \tilde{C}_t,$$  \hspace{1cm} (16)

$$O_t = \sigma(W_o^c \cdot [c_{t+1}, x_t] + b_o^c),$$  \hspace{1cm} (17)

$$c_t = O_t \odot \tanh(C_t^c).$$  \hspace{1cm} (18)

For the second backward layer, the calculation process of hidden state $d_t$ is as follows:

$$f_t = \sigma(W_f^d \cdot [d_{t-1}, c_t] + b_f^d),$$  \hspace{1cm} (19)

$$i_t = \sigma(W_i^d \cdot [d_{t-1}, c_t] + b_i^d),$$  \hspace{1cm} (20)

$$\tilde{C}_t = \tanh(W_c^d \cdot [d_{t-1}, c_t] + b_c^d),$$  \hspace{1cm} (21)

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t,$$  \hspace{1cm} (22)

$$O_t = \sigma(W_o^d \cdot [d_{t-1}, c_t] + b_o^d),$$  \hspace{1cm} (23)

$$d_t = O_t \odot \tanh(C_t^d).$$  \hspace{1cm} (24)

For each time step $t$, the output $O_t$ is generated by the combination of the hidden vectors of the second forward layer $b_t$ and the second backward layer $d_t$, we have:

$$O_t = U^{(o)}_t b_t + W^{(o)}_t d_t + b.$$

Where $U^{(o)}$, $W^{(o)}$, and $b$ are the parameters to be learned in the stacked BILSTM model.

3.4. Attention Mechanism

3.4.1. Basic attention mechanism
Different words have different importance for sentiment classification. The attention mechanism simulates the human brain, focusing on key information and filtering out irrelevant information. Such as a sentence “我今天心情不好,” (English translation: “I am in a bad mood today!”) contact context, “不好” (English translation: “bad”) is obviously more important than “心情,” (English translation: “mood”) because it can understand the meaning of the sentence. The attention mechanism is designed to focus on these key words and assign different weights to words of different importance and construct a more informative text representation. The process of attention adding is shown in Fig. 3.

The text $S_i$ contains $n$ words, let $S_i = [w_{i1}, w_{i2}, ..., w_{ik}, ..., w_{in}]$, then the attention weight $\alpha_{ik}$ of the k-th word $w_{ik}$ in the sentence $S_i$ can be expressed as:

$$\alpha_{ik} = \frac{\exp(u_{ik}^Tu_w)}{\sum_k \exp(u_{ik}^Tu_w)}.$$  \hfill (27)

Where $O_{ik}$ is the hidden layer output of stacked BILSTM, and $W_w, b_w, u_w$ are the parameters learned by the attention model. The final expression of sentence $S'_i$ is the sum of the hidden states of words weighted by attention weights, the formula is as follows:

$$S'_i = \sum_{k=1}^{n} \alpha_{ik} \times O_{ik}.$$ \hfill (28)

Where $O_{ik}$ is the hidden layer output of the k-th word learned by the stacked BILSTM layer, and $\alpha_{ik}$ is the attention weight of this word in this sentence.

Finally, perform emotion prediction, take the sentence representative $S'_i$ as input, and output the score of each emotion:

$$\hat{y} = \text{soft max}(W_{S'_i} \times S'_i + b_{S'_i}).$$ \hfill (29)

Where $W_{S'_i}$ and $b_{S'_i}$ are parameters of emotion prediction.

In the model training phase, cross entropy is used as the loss function, and the overall objective function is:

$$L_x = -\sum_{i=1}^{M} \sum_{j=1}^{U} y_{ij} \log(\hat{y}_{ij}).$$ \hfill (30)

Where $U$ is the number of sentiment categories (we are doing sentiment binary classification, i.e. $U = 2$), $M$ is the number of labeled sentences for training, and $y_{ij}$ and $\hat{y}_{ij}$ are sentiment labels and prediction scores for the i-th sample of the j-th sentiment category.

3.4.2. Sentiment awareness attention mechanism

In existing sentiment classification methods, the attention mechanism is usually learned jointly with a sentiment classification model based on labeled data and back propagation. Word embedding is usually
pre-trained on a large corpus in an unsupervised manner, and then fine-tune the labeled data through back propagation. However, pre-trained words embedded in an unsupervised manner usually do not contain sentiment information. For sentiment classification, sentiment words are important clues for the expression of sentiment and have important information value for inferring the sentiment tendency of text, such as a hotel review: “餐厅很差, 菜的种类和水准都不行!”, (English translation: “The restaurant is bad, the variety and standard of dishes are terrible!”) the word “很差” (English translation: “bad”) and “不行” (English translation: “terrible”) are very important to understand the sentiment orientation of this review. Therefore, if the attention model can recognize and highlight these emotion words in different contexts, then we can build richer information text features to classify emotions. At present, many high-quality sentiment lexicons have been summarized by many predecessors, they contain a large number of sentiment words, so we suggest using sentiment lexicons to enhance the learning of attention model in sentiment classification.

For incorporate useful knowledge of emotional semantics into the attention model, we propose a method of emotional attention supervision. As shown in Figure 4, different from the original model, we have replaced BILSTM with a stacked BILSTM, record the sentiment words in the sentences that have been divided into words and mark their positions through the sentiment lexicon, and generate a one-hot vector. For the words appearing in the sentiment lexicon, the record is 1, otherwise it is 0, such as a hotel review, after the word segmentation through the Jieba library, then we get: “餐厅/很/差/菜/的/种类/和/水准/都/不行!” (English translation: “The restaurant is bad, the variety and standard of dishes are not good!”) (’’ stands for space), the one-hot vector is \( \mathbf{P}_i = [0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0] \), through the vector \( \mathbf{P}_i \) to monitor the attention model, then through the model back propagation, update the model parameters, let the attention to the emotional words as much as possible.

For the data \( D = [S_1, S_2, ..., S_i, ..., S_M] \), where \( S_i \) represents the \( i \)-th sentence and \( M \) represents the total number of sentences. Correspondingly, compared with the emotional word lexicon, and the sentiment word position marking matrix \( \mathbf{P}_D \) of each sentence corresponding to the data \( D \) is generated \( \mathbf{P}_D = [\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_i, ..., \mathbf{P}_M] \). The attention model generates the weight matrix \( \mathbf{W}_D = [\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_i, ..., \mathbf{W}_M] \) of each sentence corresponding to the data \( D \) through the attention mechanism.

![Figure 4. Sentiment awareness attention method](image)"

In the model training phase, attention monitoring uses cross entropy as the loss function, and the overall objective function is as follows:

\[
L_{ATT} = \sum_{i=1}^{M} \mu(P_i \ast \log(\text{soft max}(W_i))).
\]  

(31)
Where $M$ represents the number of sentences, $\mu$ stands calculate mean value. The overall loss function is:

$$L = (1 - \lambda)L_S + \lambda L_{ATT}.$$  \hspace{1cm} (32)

Where $\lambda \in [0, 1]$ is the coefficient of relative importance for controlling the loss of attention monitoring.

For model training, our goal is to minimize the cross entropy between the basic facts of all sentences and the predetermined results. At the same time, in order to avoid over fitting, in each training case, we use dropout strategy to randomly miss some parameters.

4. EXPERIMENT

In this section, we describe the experimental setup and report the experimental results.

4.1. Dataset and Configuration

In order to evaluate the effectiveness of our proposed method and compare it with previous methods, we conducted experiments on two Chinese data sets, one is Professor Tan Songbo’s collection and integration of 4,000 balanced hotel reviews public dataset, and the other is our own collection and integration of 12,000 balance data set. Table II shows the descriptive statistics of the dataset used in our experiment.

During the experiment, we collected and integrated the Chinese sentiment lexicon such as HowNet lexicon, Taiwan University NTUSD lexicon, Tsinghua University Li Jun lexicon and some of the current popular words of the Internet and spoken words, such as “秀.” (English translation: “brilliant”) We put them together into a sentiment lexicon that contains 23715 sentiment words. Table III summarizes the hyper parameters used in our experiments. These hyper parameters were selected based on the performance of the validation set. We applied a dropout strategy at each layer to reduce over fitting.

| Class | Positive | Negative | Total | Positive | Negative | Total |
|-------|----------|----------|-------|----------|----------|-------|
| Train | 1167     | 1161     | 3334  | 5000     | 5000     | 10000 |
| Dev   | 208      | 208      | 416   | 624      | 624      | 2000  |
| Test  | 125      | 125      | 250   | 376      | 376      | 752   |

| Hyperparameters                   | Value |
|-----------------------------------|-------|
| Word embedding dimension          | 300   |
| Stacked BILSTM hidden layer dimensions | 128   |
| Number of stacked BILSTM hidden layers | 2     |
| Attention monitoring weight $\lambda$ | 0.05  |
| Dropout rate                      | 0.3   |

4.2. Performance Evaluation

In this section, we evaluate the performance of our method by comparing with several baseline methods, including:

- CBOW+SVM: CBOW model of word2vec and support vector machine.
- CBOW+LR: CBOW model of word2vec and logistic regression.
- CNN: Convolutional neural network.
- LSTM: Long short-term memory network.
- BILSTM: Bi-directional long short-term memory network.
- Stacked BILSTM: Stacked bi-directional long short-term memory network.
- Att_LSTM: LSTM with attention mechanism.
• Att_BILSTM: BILSTM with attention mechanism.
• BERT: Bi-directional Encoder Representation from Transformers.

We used the above model and our own model (we call it Att_lex_Stacked_BILSTM) to conduct experiments on the two data sets, and the results are shown in Table IV.

From the table, we can see that our proposed model has higher accuracy than the previous models in terms of 4000 or 12000 data sets. Moreover, the accuracy of 12,000 data is higher than that of 4000 data. Comprehensive consideration should be that the model training is more sufficient, so the effect is improved obviously.

4.3. Effect of Sentiment Lexicon
In order to prove that the combination of sentiment lexicon does have a positive effect on the model, we designed a comparison model, a model without sentiment lexicon (we call it Att_nolex_Stacked_BILSTM). The experimental results are shown in Table V.

It can be seen from the Table V that the addition of sentiment lexicons has a positive impact on the model in the two data sets of 4000 data and 12000 data. The accuracy of both data sets has been improved, the accuracy of 4000 data has reached 4.4%, and the accuracy of 12000 data has been improved by 0.9%.

4.4. Effect of Stacked BILSTM Hidden Layers
Facts show that the model effect of the stacked BILSTM with 2 hidden layers is better than the BILSTM model with a single hidden layer. More upper layer extraction can enrich the semantic information and improve the accuracy. So, can we consider setting more BILSTM hidden layers, the better the model is? To prove the conjecture, we also designed a comparative experiment. Based on the supervision of sentiment lexicon, taking the 12000 data as an example, we set up different hidden layers respectively. The experimental results are shown in Table VI.

From the table, it is not that the more hidden layers, the higher the accuracy of the model. More hidden layers are set, but the result is not ideal, and the accuracy gradually decreases. Two layers have reached the highest accuracy, so more upper layer extraction can further enrich semantic information, there are limits.

| Table IV. | 4000 AND 12000 DATA |
|-----------|--------------------|
| Model     | Accuracy           |
|           | 4000   | 10000 |
| CBOB+SVM  | 65.2%  | 80.9% |
| CBOB+LR   | 73.2%  | 82.5% |
| CNN       | 68.0%  | 86.7% |
| LSTM      | 71.2%  | 85.2% |
| BILSTM    | 72.0%  | 87.2% |
| Stacked BILSTM | 73.2% | 87.5% |
| ATT_LSTM  | 78.8%  | 96.3% |
| ATT_BILSTM| 80.0%  | 96.4% |
| BERT      | 86.4%  | 95.6% |
| Att_lex_Stacked_BILSTM(ours) | 87.6% | 97.7% |

| Table V. | EFFECT OF SENTIMENT LEXICON ON DATA ACCURACY |
|-----------|---------------------------------------------|
| Model     | Accuracy          |
|           | 4000   | 10000 |
| Att_nolex_Stacked_BILSTM | 83.2% | 96.8% |
| Att_lex_Stacked_BILSTM    | 87.6% | 97.7% |
TABLE VI. **EFFECT OF HIDDEN LAYERS ON THE ACCURACY OF 12000 DATA**

| Hidden layers | Accuracy |
|---------------|----------|
| 1             | 97.1%    |
| 2             | 97.7%    |
| 3             | 96.4%    |
| 4             | 96.0%    |

5. CONCLUSIONS

In this paper, we propose a sentiment classification method that combines the composite model stacked BILSTM with attention mechanism and sentiment lexicon. We describe and discuss the effect of the number of hidden layers of stacked BILSTM and sentiment lexicon on the performance of the model. In the experiment, we proved that our model is significantly better than the previous model, and the classification accuracy has been greatly improved. In the future, we can consider whether turning words have further influence on sentiment classification. For sentences containing turning words, the clauses before and after the transitional words may have completely opposite emotional polarities. Accurately obtaining the subjective sentiment expression clauses and classifying sentiment may further improve the accuracy of the model's sentiment classification. In future work, it may be worth studying to explore the effect of turning words on the accuracy of the sentiment classification.

REFERENCES

[1] F. Wu, Y. Huang, Y. Song, and S. Liu, “Towards building a high-quality microblog-specific chinese sentiment lexicon,” Decis. Support Syst., vol. 87, pp. 39–49, Jul. 2016.
[2] 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.
[3] W. Du, S. Tan, X. Cheng, et al., “Adapting information bottleneck method for automatic construction of domain-oriented sentiment lexicon”, ACM Int. Conf. Web Search Data Min. (2010) 111-120. Available: https://dl.acm.org/doi/10.1145/1718487.1718502.
[4] Elissa, “Title of paper if known,” unpublished.
[5] J.M. Chen, H.F. Lin, Z.H. Yang, “Automatic acquisition of emotional vocabulary based on syntax”, CAAI Trans. Intell. Syst. 4 (2) (2009) 100-106.
[6] A.P. Li, P. Di, G.L. Duan, “Document sentiment orientation analysis based on sentence weighted algorithm”, J. Chin. Comput. Syst. 36 (10) (2015) 2252–2256.
[7] Y. Lu, M. Castellanos, U. Dayal, “Automatic construction of a context-aware sentiment lexicon: An optimization approach”. International Conference on World Wide Web, 2011, pp. 347-356. Available: https://dl.acm.org/doi/10.1145/1963405.1963456.
[8] F. Wu, Y. Huang, Y. Song, and S. Liu, “Towards building a high quality microblog-specific chinese sentiment lexicon,” Decis. Support Syst., vol. 87, pp. 39–49, Jul. 2016.
[9] 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.
[10] W. Li, Y. Li, and Y. Wang, “Chinese microblog sentiment analysis based on sentiment features”, in Proc. Asia-Pacific Web Conf. Suzhou, China: Springer, 2016, pp. 385-388.
[11] Breiman, L.: “Random forests. Machine learning” 45(1), 5-32 (2001)
[12] B. Chen, Z. Hao, R. Cai, W. Wen, and S. Du, “Sentiment target extraction based on CRFs with multi-features for chinese microblog,” in Proc. Asia-Pacific Web Conf. Suzhou, China: Springer, 2016, pp. 29-41.
[13] L. Zheng, H. Wang, and S. Gao, “Sentimental feature selection for sentiment analysis of chinese online reviews,” Int. J. Mach. Learn. Cybern., vol. 9, no. 1, pp. 75-84, 2018.
[13] Adankon, M.M., Cheriet, M.: “Support vector machine”. In: Encyclopedia of Biometrics, pp. 1303–1308. Springer, Boston (2009)

[14] D. Zhang, H. Xu, Z. Su, and Y. Xu, “Chinese comments sentiment classification based on word2vec and SVM perf,” Expert Syst. Appl., vol. 42, no. 4, pp. 1857–1863, Mar. 2015.

[15] Rish, I.“An empirical study of the naive Bayes classifier”. In: IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence, vol. 3(22), pp. 41–46. IBM, August 2001

[16] P. Ficamos, Y. Liu, and W. Chen, “A naive Bayes and maximum entropy approach to sentiment analysis: Capturing domain-specific data in Weibo”, in Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp), Feb. 2017, pp. 336–339.

[17] Yoon Kim. 2014. “Convolutional neural networks for sentence classification”. In EMNLP. pages 1746–1751.

[18] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. “A convolutional neural network for modelling sentences”. In ACL. pages 655-665.

[19] Lai, S., Xu, L., Liu, K., and Zhao, J. 2015. “Recurrent convolutional neural networks for text classification”. In AAAI, vol. 333. vol. 333, 2267-2273

[20] Sepp Hochreiter and Jürgen Schmidhuber. 1997. “Long short-term memory”. Neural Computation 9(8):1735–1780.

[21] Qiao Qian, Bo Tian, Minlie Huang, Yang Liu, Xuan Zhu, and Xiaoyan Zhu. 2015. “Learning tag embeddings and tag-specific composition functions in recursive neural network”. In ACL. volume 1, pages 1365–1374.

[22] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” IEEE Trans. Signal Process., vol. 45, no. 11, pp. 2673–2681, Nov. 1997.

[23] Chiu, J. and Nichols, E., “Named entity recognition with bidirectional lstm-cnns”. Transactions of the Association for Computational Linguistics 4 (2016), 357-370.

[24] Wang, J., Yu, L.-C., Lai, K. R., and Zhang, X. 2016. “Dimensional sentiment analysis using a regional cnn-lstm model”. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). Association for Computational Linguistics, 225-230. Available: https://www.semanticscholar.org/paper/Dimensional-Sentiment-Analysis-Using-a-Regional-Wang-Yu/6f72ca2f58e4e7b0a4e218f9ee2f65f6662ec5b2

[25] Bahdanau D, Cho K, Bengio Y. “Neural Machine Translation by Jointly Learning to Align and Translate” [C]//Proceedings of the ICLR. 2015.

[26] Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; and Hovy, E. 2016. “Hierarchical attention networks for document classification”. In NAACL-HLT, 1480–1489.