

Article

Emotion-Semantic-Enhanced Bidirectional LSTM with Multi-Head Attention Mechanism for Microblog Sentiment Analysis

Shaoxiu Wang, Yonghua Zhu *, Wenjing Gao, Meng Cao and Mengyao Li

Shanghai Film Academy, Shanghai University, Shanghai 200072, China; iamalion@shu.edu.cn (S.W.); gaowj715@shu.edu.cn (W.G.); cicimeng@shu.edu.cn (M.C.); limengyaoyao@shu.edu.cn (M.L.)

* Correspondence: zyh@shu.edu.cn

Received: 26 April 2020; Accepted: 20 May 2020; Published: 22 May 2020



Abstract: The sentiment analysis of microblog text has always been a challenging research field due to the limited and complex contextual information. However, most of the existing sentiment analysis methods for microblogs focus on classifying the polarity of emotional keywords while ignoring the transition or progressive impact of words in different positions in the Chinese syntactic structure on global sentiment, as well as the utilization of emojis. To this end, we propose the emotion-semantic-enhanced bidirectional long short-term memory (BiLSTM) network with the multi-head attention mechanism model (EBiLSTM-MH) for sentiment analysis. This model uses BiLSTM to learn feature representation of input texts, given the word embedding. Subsequently, the attention mechanism is used to assign the attentive weights of each words to the sentiment analysis based on the impact of emojis. The attentive weights can be combined with the output of the hidden layer to obtain the feature representation of posts. Finally, the sentiment polarity of microblog can be obtained through the dense connection layer. The experimental results show the feasibility of our proposed model on microblog sentiment analysis when compared with other baseline models.

Keywords: sentiment analysis; Chinese microblog; BiLSTM; multi-head attention; emoticons

1. Introduction

With the rapid development of the Internet and social networks, an increasing amount of users begin to freely share their own opinions and comments on the web to express their personal emotional opinions on some issues or events. The massive data reveal the public's emotion or attitudes, which is of great value to the applications in political, financial, entertainment, and news industries. From a technical point of view, the development of artificial intelligence (AI) has been rather rapid in recent years. As a major branch of AI, natural language processing (NLP) has attracted considerable attention in both research and industrial fields [1]. Sentiment analysis, one of the hottest topics in NLP, is of great significance for the timely understanding of online public opinion, market research, online public opinion monitoring, and early warning, so more and more researchers are committed to the field of sentiment analysis.

Sentiment analysis, which is also known as opinion mining, refers to the process of analysis, processing, induction, and reasoning of subjective text with emotion [2]. Sentiment analysis is a text classification technology that involves research fields, such as NLP, machine learning, data mining, and information retrieval. Its main data source can be the textual contents publicly released on the popular social networking PLATFORMS. Sina Weibo, as a social network platform for sharing, disseminating, and acquiring user relationship information, has become an extremely important channel for obtaining the public's opinions or emotions on specific events.

Sentiment analysis of microblog texts has extremely high application value due to the huge user base and rich information generated by users. However, although sentiment analysis in English language is well studied, research in Chinese sentiment analysis is substantially less developed [1]. Especially, sentiment analysis on microblog is more challenging under the circumstances of lacking contextual information, serious colloquialism, a large number of symbols, the emergence of new words, etc. Therefore, how to mine the discriminative features effectively from the massive, unstructured microblog is a key for microblog sentiment analysis. The previous microblog sentiment analysis methods mainly focused on the content of words, especially the role of nouns, adjectives, verbs, and adverbs, while ignoring the influence of the association structure and emoticons on the sentiment of the sentence, resulting in deviations in the final sentiment analysis result.

We propose the EBILSTM-MH for sentiment analysis on microblog to tackle these issues. Specifically, the Word2Vec model is adopted to represent the input texts. Subsequently, with the help of the BiLSTM, the abstract characteristics of microblog are extracted. Besides, we introduce the attention mechanism for calculating the attentive score of each word based on emoticons. The weighted sum of the weight value and output of hidden layer is the final representation of the microblog content. Finally, a classifier is trained to realize the sentiment classification of microblog content. The main contributions of this article are:

- We collect and sort out the correlation structures that have a turning or progressive effect on the global sentiment of microblog in the Chinese grammatical structure. The special correlation structures are maintained in the pre-processed corpus to avoid the model wrongly judging posts' sentiment polarity.
- We collect and organize the new words appearing on Weibo in the past ten years, and then add them to the user-defined dictionary of jieba word segmentation toolkit to avoid the loss of important semantic information and word segmentation errors, meanwhile, indirectly expand the vocabulary of Word2Vec model.
- We sort out the common emoticons in Sina Weibo and regard them as an important basis for sentiment analysis. The multi-head attention mechanism is used to calculate the contribution of words to global sentiment analysis, and the emotional semantic enhancement of emoticons is exerted.

The remaining parts of the paper are organized, as follows: Section 2 reviews related works on sentiment analysis. Section 3 presents the model that we proposed and this section also gives a modular mathematical elaboration on the structure of model. Section 4 concerns the experiments that are based on the real-world dataset, where the experiment configuration, the experiment results, and corresponding analysis are given in detail. Finally, we conclude our work and discuss future research direction in Section 5.

2. Related Works

The text sentiment analysis has attracted wide attention and become a hot topic in the field of NLP. Various kinds of approaches have been proposed for sentiment analysis.

Sentiment analysis can be divided into three levels, including sentiment judgment at the word, sentence, and chapter levels [3–5]. Sentiment analysis can be divided into aspect-based sentiment analysis and traditional polarity classification, according to the division of basic tasks [6]. Aspect-based sentiment analysis aims at inferring the sentiment polarity (e.g., positive, negative, neutral) of a sentence expressed toward a target, which is the aspect of one specific entity [7]. Many researchers have contributed to this field. Zhou et al. [8] developed an unsupervised label propagation algorithm for opinion target extraction and clustered the opinion targets into several groups. Subsequently, a co-ranking algorithm was proposed to rank both the opinion targets and microblog sentences simultaneously. It was the first study in which aspect-based opinion summarization was performed on Chinese microblog texts. Ma et al. [9] modeled attention as a stepping model of coding objectives

and full sentences and proposed an extension of LSTM units to more effectively combine emotional common-sense knowledge when encoding sequences as vectors. The model designed by them can effectively filter information that conflicts with the background knowledge. Peng et al. [10] modeled the aspect target and conduct sentiment classification directly at the aspect target level via three granularities, which is radical, character, and word. Moreover, they studied two fusion methods to model aspect target sequence in the task of Chinese aspect-level sentiment analysis and achieved outstanding experimental results on three Chinese review datasets. Based on the number of categories of emotion, sentiment classification can be divided into binary sentiment classification [11], ternary sentiment classification [12], and multi-sentiment classification [13–15]. Among them, the binary sentiment classification is to divide the emotion tendency of the research object into two categories: positive and negative, while the ternary emotion classification is to divide the emotion tendency into three categories: positive, neutral, and negative, while the multi sentiment classification is more complex. Human emotion can be divided into a variety of basic sentiments, according to the different emotion theories referred by the researchers.

At present, the methods of sentiment analysis are mainly divided into three categories: the method based on sentiment lexicon [16,17], the method based on machine learning [18,19], and the method based on deep learning [20,21]. Next, we mainly discuss the related work of sentiment analysis in detail according to the basic method.

The lexicon-based approaches focus on recognizing the characters with obvious emotion tendency in the texts under the help of external databases, and predicting the sentiment based on the word-level sentiment value and specific calculation rules [14]. This kind of heuristic method based on external knowledge does not require training samples and it is relatively easy to implement. Almeida et al. [15] established sentiment lexicon through the extended affective tendency point-wise information algorithm. Dong et al. [22] first obtained the embedding of seed emotion words and localized the words in the documents by LSTM, autofocus mechanism, and L1 rules. Subsequently, a logistic regression was trained to judge the polarity of the document words, so as to expand the lexicon. Pu et al. [23] constructed semantic association graph of sentiment dictionary based on the similarity between seed words and candidate sentiment words, and finally the polarity of sentiment words is calculated through the label propagation algorithm. Zhang et al. [24] established a lexicon based on several dictionaries including a dictionary of basic emotions, a dictionary of degree adverbs, a dictionary of negative words, a dictionary of Internet words, an emoji dictionary, and a dictionary of relational conjunctions, and proved that emoticons and relational conjunctions can effectively improve the accuracy of sentiment analysis. Based on this observation, the influence of correlation structures and emoticons on microblog sentiment is also comprehensively considered in the method that is proposed in this article. Specifically, we screen and retain the special correlation structures in posts and use emoticons to enhance the emotion semantics of blog.

Supervised learning is the mainstream for the machine learning based sentiment analysis. Classifiers, such as Naive Bayes (NB) [25], support vector machine (SVM), Maximum Entropy, and k-nearest neighbor (KNN), have been widely used in short text sentiment analysis, such as twitter and microblog in combination with text features, and have significantly improved the performance. Bo et al. [26] used SVM, NB, maximum entropy, and other methods to judge the sentiment polarity of the film reviews, but they failed to consider the fact that users with different personalities have different review styles. Liu et al. [27] extracted TF-IDF features from Chinese microblog and sent them into a random forest for emotion classification. Hatzivassiloglou et al. [2] used a variety of machine learning classification algorithms to conduct sentiment analysis of Twitter data, and the experiment verified that the NB algorithm is the best one for emotion classification of Twitter data when the data sets were small.

In recent years, an increasing amount of researchers have introduced deep learning into sentiment analysis and achieved considerable experimental results. Deep learning models enable more robust word embedding than the traditional hand-crafted ones. Specifically, deep neural networks can encode

semantic and grammatical features in the text and provide relatively accurate information represented by the document [11], such as Word2Vec [28] and Doc2Vec [29]. Besides, some Chinese researchers presented methods to leverage radical for learning Chinese character embedding. Sun et al. [30] presented a method to leverage radical for learning Chinese character embedding. They introduced a dedicated neural architecture with a hybrid loss function to effectively integrate radical information through the softmax layer. This is the first work that utilized the radical information for learning Chinese character embedding. Peng et al. [31] established semantic radical embeddings and sentic radical embedding by using the skip-gram model, which incorporated not only semantics at radical and character level, but also sentiment information. Wang et al. [32] built a sentiment analysis model fusing tweet-text information and user-relationship information through a three-layer neural structure, which is a word embedding layer, a user embedding layer, and an attentional graph layer. When compared with several other common methods, the performance of this model is improved by more than 5%. Irsoy et al. [33] used the recurrent neural network model (RNN) based on time series information to obtain sentence representation, which can also further improve the accuracy of emotion classification. However, RNN model is confronted with the problem of gradient disappearance when dealing with long distance dependency. The introduction of the LSTM model and its variants, like BiLSTM, stacked BiLSTM, tackle this problem and they are widely utilized to model the long text. Due to this, we use the BiLSTM network to extract the semantic information in the text sequences.

3. EBILSTM-MH Structure

Figure 1 shows the overall framework of EBILSTM-MH, which is mainly divided into two parts, namely data preprocessing and classifier design. The first stage is to perform a series of necessary natural language processing on the microblog texts, such as data cleaning, word segmentation, and stop words removal to obtain keyword phrases, emoticons, and important related structures. In the second stage, the Word2Vec model is first trained on the microblog corpus to obtain the feature representation of text and emoticons. Subsequently, the joint embedding of text and emoticons is sent into the BiLSTM layer to extract abstract features. We propose a multi-head attention layer to determine the contribution of each word to the sentiment conveyance that is based on the emoticons. Finally, the sentiment classification of microblog is obtained after normalization in the dense connection layer.

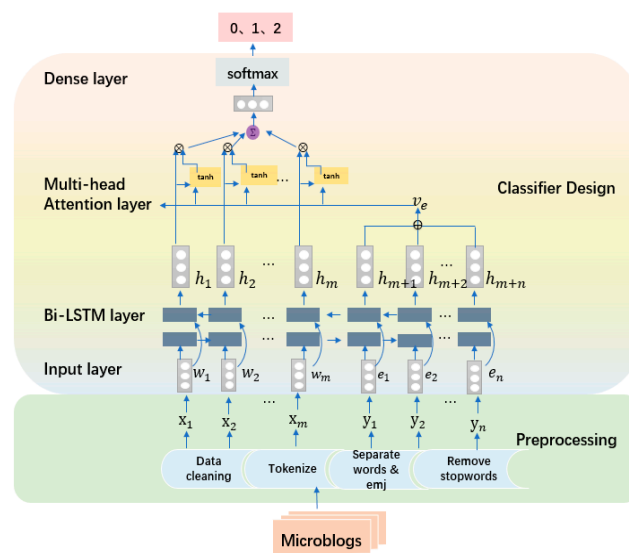


Figure 1. The framework of proposed model.

3.1. Data Preprocessing













Most of the existing methods for sentiment analysis mainly focus on the extraction of key words, such as nouns, adjectives, verbs, and adverbs, while ignoring the influence of correlation structures in Chinese grammar. For example, “尽管这个小小的地下室阴暗潮湿，一家三口住在这儿也显得着实拥挤，但是对于洋洋来说能和家人住在一起，即使环境再恶劣也不是什么多大的问题 (although this small basement is dark and humid, the family of three living here is also very crowded, for Yangyang to be able to live with his family, even if the environment is bad, it is not a big problem)”. In this sentence, it is easy to judge it as a negative emotion in traditional word-level sentiment analysis. However, we find that this sentence belongs to a positive emotion by analyzing the related turning structure, “尽管... 但是... (although... but...)”, “即使... 也不... (even if... not at all)”. Based on this observation, eight kinds of association relations in Chinese grammar are collected and sorted out, which are sum up to 115 association structures. Among these, we extract three relationships involving 39 relational structures when considering their turning or progressive effect on the emotional expression. We separate the keywords related to these association structures from the stop words list to prevent the information from being discarded. Table 1 shows some examples of the extracted relationships and correlation structures.

Table 1. Part of relations and correlation structures.

Relation	Correlation Structures
Progressive relation	不但不... 反而... (not only... not... ,but also...) 尚且... 何况... (... not to mention...) 甚至... (... even...)
Selection relation	宁可... 也不... (rather... not...) 与其... 不如... (It is not as good as...) ... 还是... (... or...)
Twist relation	虽然... 但是... (However, although...) 尽管... 可是... (... despite...) 然而... (... yet...)

Linguists have found that emoticons are one of the most important features of network language [14], which not only enhance the emotional expression of the subjective text, but also add subjective feelings to the objective text [34]. Microblog platform provides a lot of emoticons for users to express their emotions. There are more than 1500 commonly used emoticons, according to the statistics [14]. In this paper, we collect about 170 commonly used emoticons on microblog platform for enhancing the microblog sentiment analysis. Table 2 shows some emoticons that we utilized. In addition, some recently popular neologisms on microblog are also taken into consideration in this paper. Those words are produced by Chinese homophonic, such as “酱紫”, which mean “like this way”, or from some online celebrities, such as “猴赛雷”, which means “awesome”. We add them to the user defined dictionary in the jieba toolkit to avoid the loss of the word vector of the network neologisms and the semantic loss of the microblog text due to the word segmentation errors.

Table 2. Emoticons and corresponding meaning.

Emoticon	Meanings	Emoticon	Meaning	Emoticons	Meaning
	[Haha]		[startle]		[contempt]
	[love you]		[handclap]		[shy]
	[suffer beating]		[ok]		[No]
	[good]		[sad]		[hum]

3.2. Classifier Design

One of the novelties of our method lies in the utilization of the BiLSTM network and multi-head attention mechanism combined with emoticons for enhancing the microblog sentiment analysis. Particularly, we divide the design part of the classifier into four layers, namely the input layer, BiLSTM layer, multi-head attention layer, and dense connection layer.

Input layer: the input layer generates the embedding of words and emojis. After the above-mentioned preprocessing, the input microblog data are expressed as: $\{x_1, \dots, x_i \dots x_m; y_1, \dots, y_j, \dots, y_n\}$, where x_i is the i th word and y_j is the j th emoticon in microblog. After the words and emoticons are mapped to the corresponding embedding, the input layer is represented as $\{w_1, \dots, w_i, \dots, w_m; e_1, \dots, e_j, \dots, e_n\} \in R^{(m+n) \times d}$, where $(m+n)$ represents the total number of words and emoticons in microblog, and d is the feature dimension.

BiLSTM layer: LSTM is a variant of RNN, which can solve the problem of gradient disappearance by introducing input gate i , output gate o , forget gate f , and memory cell state. LSTM can improve the memory mechanism of neural network to receive input information and training data, which is very suitable for modeling time series data, like texts, due to the characteristics of its design. BiLSTM is a combination of forward LSTM and backward LSTM. The former deals with the sequence from left to right, while the latter deals with the sequence from right to left. The biggest advantage of this structure is that the sequence context information is fully considered. Next, we introduce the procedures of LSTM in detail.

An LSTM unit consists of three controlling gates, including an input gate i_t , a forget gate f_t , and an output gate o_t , as well as a memory cell state c_t , all of which affect the unit's ability to store and update information. The input gate output a value between 0–1 based on the input h_{t-1} and w_t (see Equation (1)). When the output is 1, it means that the cell state information is completely retained, and when the output is 0, it is completely abandoned. Next, the input gate layer decides which values need update, and the tanh layer creates a new candidate value vector \tilde{c}_t , which can be added to the cell state. Subsequently, the two will be combined to update the cell state c_t (see Equations (2)–(4)); finally, the output gate will determine the output value based on the cell state (See Equations (5) and (6)). Among them, $W_f, U_f, b_f, W_i, U_i, b_i, W_c, U_c, b_c$, and W_o, U_o, b_o are the internal training parameters in the LSTM, $\sigma(\cdot)$ is sigmoid activation function, and \odot denotes dot multiplication.

$$f_t = \sigma(W_f w_t + U_f h_{t-1} + b_f) \quad (1)$$

$$i_t = \sigma(W_i w_t + U_i h_{t-1} + b_i) \quad (2)$$

$$\tilde{c}_t = \tanh(W_c w_t + U_c h_{t-1} + b_c) \quad (3)$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} \quad (4)$$

$$o_t = \sigma(W_o w_t + U_o h_{t-1} + b_o) \quad (5)$$

$$h_t = o_t \tanh \odot (c_t) \quad (6)$$

The above is the calculation process of LSTM. As we said earlier, BiLSTM includes forward LSTM and backward LSTM. \overrightarrow{LSTM} in BiLSTM, read the input from w_1 to e_n to generate $\overrightarrow{h_t}$ and another \overleftarrow{LSTM} read the input from e_n to w_1 in order to generate $\overleftarrow{h_{t-1}}$:

$$\overrightarrow{h_t} = \overrightarrow{LSTM}(w_t, \overrightarrow{h_{t-1}}, c_{t-1}), t \in [1, m + n] \tag{7}$$

$$\overleftarrow{h_t} = \overleftarrow{LSTM}(w_t, \overleftarrow{h_{t-1}}, c_{t-1}), t \in [m + n, 1] \tag{8}$$

The forward and reverse context representations generated by $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ are connected into a long vector, and the combined output is the representation of the current time to the input:

$$h_t = \overrightarrow{h_t} \oplus \overleftarrow{h_t} \tag{9}$$

Finally, the output $[h_1, \dots, h_i, \dots, h_m, l_1, \dots, l_j, \dots, l_n]$ of the whole sentence is obtained, where h_i and l_j are utilized to represent the output of words and emoticons, respectively, in the hidden layer. In addition, we set all of the intermediate layers in BiLSTM to return the complete output sequence, thereby ensuring that the output of each hidden layer retains the long-distance information as much as possible.

Multi-head attention layer: attention is a mechanism for improving the effect of RNN-based models, and the calculation of attention is mainly divided into three steps. The first step is to use the attention function F to score query and key to get s_i . The two most common attention functions are additive attention and dot-product attention [35]; in this article, we use the former. The second step is to use softmax function to normalize the scoring result s_i , so as to obtain the weight a_i . The third step is to calculate attention, which is the weighted average of all values and weights a_i . Figure 2 shows the flow chart of the attention mechanism.

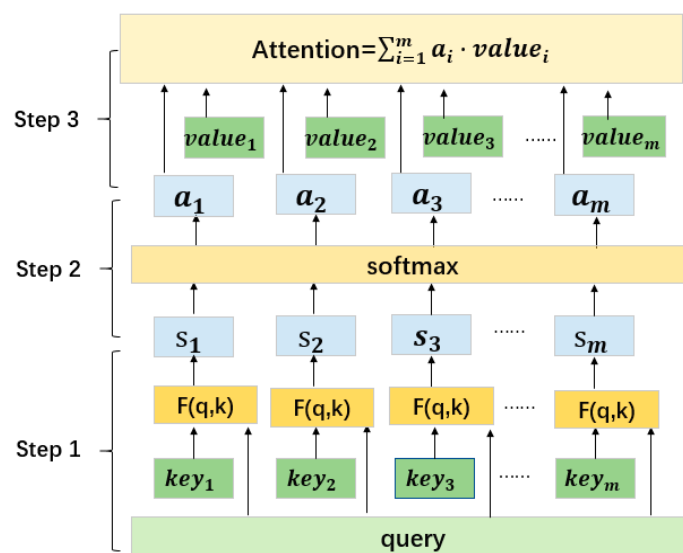


Figure 2. The flowchart of attention mechanism.

Multi-head attention has improved the traditional attention mechanism, so that each head can extract the features of query and key in different subspaces. To be precise, these features come from Q and K, which are the projection of query and key in the subspace. The operation that is shown in Figure 2 is performed once in each head, and a total of h times need to be performed. It should be noted that in the multi-head attention mechanism, the attention function could be the scaled dot-product function, which is the same as the traditional attention mechanism, except for the regulating scaling

factor. In the experiment, h needs to be continuously debugged to determine the most suitable value for the task. Finally, the results that are returned in each head are concatenated and linearly converted to obtain multi-head attention. Figure 3 shows the main idea. Next, we will combine the tasks in this article to explain, in detail, how we use the multi-head attention mechanism.

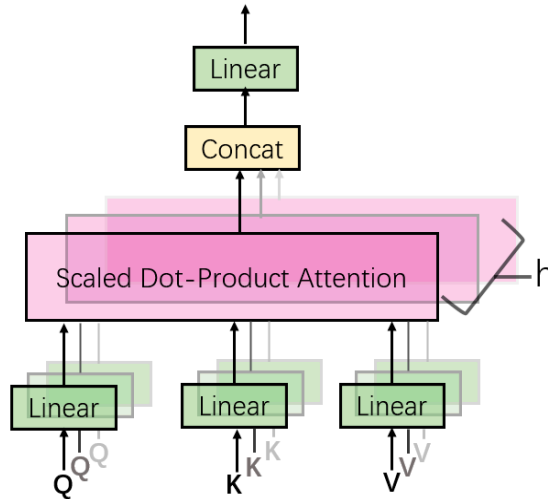


Figure 3. The structure of multi-head attention.

As we all know, each word contributes to the sentiment conveyance differently, and the effect of combining words with emoticons is also different. Therefore, in this paper, we propose scoring emoticons and words, and then weight the importance of each word in determining the emotional polarity of microblog.

First, w_i and e_j are the word embeddings corresponding to the words and emoticons in microblog, respectively, where $w_i, e_j \in R^d$ and d is the dimension of the word vector. The BiLSTM generates more abstract representations of word and emoji, namely $h_i \in R^k$ and $l_j \in R^k$, given the word embedding of words and emoticons as input. Observing that many users will post multiple emojis in the same microblog or multiple identical emojis in a row, but only emoticons that are not repeated in blog post are extracted in our method, and the number of different emoticons is limited to no more than 5. All emojis are combined to obtain the emoticon representation:

$$q_v = l_1 \oplus l_2 \oplus l_3 \oplus l_4 \oplus l_5 \tag{10}$$

The attention function that is used in this paper is additive attention [36], which performs better in higher dimensions. We regard the output $[h_1, h_2, \dots, h_m]$ of words and q_v of emoticons in the BiLSTM hidden layers as the query and key in the attention mechanism to calculate the attentive scores, and perform a softmax normalization operation in order to obtain the weight α_t , which represents the importance of the t -th word combined with emoticons for sentiment analysis:

$$f(h_i, v_e) = v^T \tanh(W_h h_t + W_e q_v + b) \tag{11}$$

$$\alpha_t = \text{softmax}(f(h_i, v_e)) = \frac{\exp(f(h_i, v_e))}{\sum_{i=1}^T \exp(f(h_i, v_e))} \tag{12}$$

Among them, W_h, W_e are the weight matrices; b is the bias; v is the weight vector; and, v^T is the transpose of v . Accordingly, output of each head is:

$$\text{head}_s = \sum_{t=1}^m \alpha_t h_t \tag{13}$$

When compared with the attention mechanism with only one head, multi-head attention allows for the model to jointly attend to information in different feature subspaces. It performs the attention function in parallel. Subsequently, we concatenate the results of every head in a linear manner, resulting in the final output of this layer.

$$output = concatenate(head_{s_1}, head_{s_2}, \dots head_{s_m}) W_o \tag{14}$$

$$head_{s_i} = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{15}$$

where W_i^Q, W_i^K, W_i^V are the parameter matrices that project $Q, K,$ and V to different representation subspaces, and $Q = q_v, K = V = [h_1, h_2, \dots h_m]$.

Dense connection layer: finally, we send the vectors from the previous layer to the densely connected layer. We use the ReLu function as the activation function to complete the nonlinear conversion. At the last densely connected layer, we perform Softmax operation on the output of the previous layer, and finally obtain the sentiment classification of microblog.

4. Experiment

4.1. Experiment Environment

The experiments in this article were conducted on AMD A4-6210 APU with AMD Radeon R3 Graphics, 4G memory. The development environment is the Python 3.7 programming language. Python’s keras library and scikit library are used to build the models and comparative experiments that are proposed in this article.

4.2. Dataset Construction

The dataset that was used in this article comes from the NLPCC2013 and NLPCC2014 Chinese microblog sentiment analysis competition. The original data were divided into eight categories, namely “anger”, “sadness”, “surprise”, “like”, “happy”, “fear”, “disappointment”, and “none”. In the task of sentiment analysis, we change the corresponding text labels of “anger”, “sadness”, “fear”, and “disappointment” into “negative” labels, while “happy”, “surprise”, and “like” are divided into “positive” categories, “none” is corresponding to “neutral”. Finally, the data from the NLPCC2013 and NLPCC2014 datasets are adopted to form the required datasets for this article. The specific dataset is constructed, as shown in Table 3.

Table 3. The statistics of dataset.

Sentiment Polarity	Negative		Positive		Neutral	
NLPCC emotion type	Anger	1717	Happiness	2000	None	5000
	Fear	415	Like	4000		
	Sadness	1003	Surprise	700		
	Disgust	1565				
Total	4700		6700		17100	

4.3. Experimental Preprocessing

Text preprocessing: in this paper, a special pattern is used to extract blog posts and sentiment tags from NLPCC raw data. The Chinese word segmentation task is completed by jieba, a third-party library of Python. We partially modify the toolkit, so that it can accurately separate the pre-defined symbols in the text.

Word embedding: in this experiment, we used genism, a python natural language processing toolkit, in order to train the Word2Vec model on the microblog corpus. The skip-gram algorithm,

sensitive to low-frequency words, is used during our training process. Table 4 shows the specific parameters used in the training process of Word2Vec model.

Table 4. Training parameters for Word2Vec.

Sg	Window	Sample	Min_count	Negative	Hs	Workers
1	5	0.001	10	1	1	4

Parameters setting: in order to balance the proportion of different types of microblog in the training data and the test data, stratified sampling is adopted to determine the number of microblog posts of different emotion categories in the training set and test set, and the dataset in the experiment is divided into training set and test set according to the ratio of 8:2. In the training process, we calculated the length of the microblogs after word segmentation and stop words removal, which would hardly exceed 80 characters, i.e., only seven microblog posts are longer than 80 characters. Figure 4 shows the statistical results. The length of the pre-processed microblog is limited to 80 based on this observation. Table 5 shows the parameters in EBILSTM-MHA and other baselines in this paper, and the Appendix A at the end of this paper describes more details.

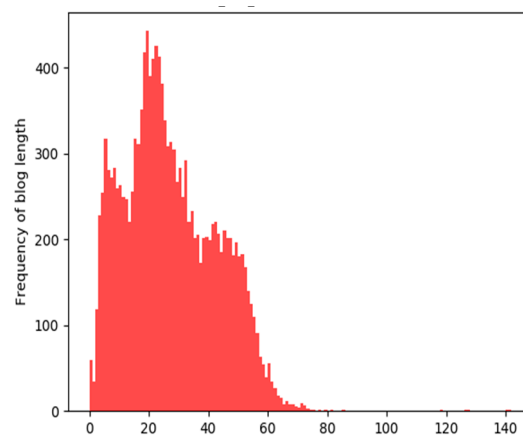


Figure 4. The frequency of different blog length.

Table 5. Parameters in our experiment.

Parameters	Dimension(d)	Maximum Sentence Length	Dropout Rate	Heads	Batch_Size
value	200	80	0.4	8	32
Parameters	Patience	Epochs	Conv1D	Kernel size	Pool size
value	7	30	256	3, 4, 5	28, 27, 26

Evaluation method: in this paper, we use comprehensive measures, including precision, accuracy, recall rate, and F1 value, to evaluate the performance of the model: the definition of different measures is as follows, and the relevant parameters are shown in Table 6.

$$accuracy = \frac{TP + TN}{TN + FP + FN + TP} \quad (16)$$

$$Precision = \frac{TP}{TP + FP} \quad (17)$$

$$Recall = \frac{TP}{TP + FN} \quad (18)$$

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (19)$$

Table 6. The relevant parameters.

	Label	Prediction	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

4.4. Comparative Experiment

We compare EBILSTM-MH with the following baselines in order to confirm the effectiveness of introducing the emoticons to microblog sentiment analysis by multi-head attention mechanism:

BiLSTM + multi-head_attention (text only): this method only pays attention to the text information in microblog, and does not consider the influence of emoticons on the semantics of words.

BiLSTM + emj_attention: this method considers the enhancement effect of emoticons on the sentiment judgment of microblog, but uses the attention mechanism with single head to calculate the attentive weights of each word. Besides, the additive attention function is adopted in this model.

SVM: SVM is one of the commonly used baselines for emotion classification. It is a multi-class SVM classification that is based on structural SVM. We use an SVM with a linear kernel, making the penalty coefficient (C) equal to 1.

CNN: CNN is also one of the commonly used models for sentiment classification, which could show advantages in learning complex data. We combine the CNN model with the pretrained embedding to achieve sentiment classification. This model also considers the effect of text and emoticons on microblog sentiment analysis.

4.5. Experimental Results and Analysis

Result: the evaluation metrics that are used in this paper include accuracy, precision, recall, and F1 values. Table 7 shows the final evaluation results of all the baselines and our proposed model.

Table 7. The Result of proposed method and other models.

Method	Positive			Negative			Neutral			Accuracy (%)
	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	P(%)	R(%)	F(%)	
EBILSTM-MH	76.77	63.62	69.58	76.04	73.94	74.97	65.04	77.20	70.60	71.70
BiLSTM + emj_att	73.93	62.13	67.51	74.32	72.66	73.49	62.86	73.60	67.80	69.55
BiLSTM + multi-head_att(text_only)	68.27	64.79	66.48	71.72	74.47	73.07	63.64	64.40	64.02	67.81
SVM	63.17	67.87	65.44	73.23	70.40	71.80	64.59	62.40	63.48	66.81
CNN	60.07	70.10	64.70	66.97	61.91	64.34	59.41	54.30	56.74	63.19

It can be seen from the Table 7 that SVM, as a classical classifier for sentiment analysis, achieves fairly good results when considering both text and emoticons, with an accuracy of about 66.81%. As for another traditional model, CNN, which has the worst performance in this task, the accuracy of this model only reaches about 63%. The main reason might be that SVM does not require a high sample size, while the sample size limits the performance of other models. Even so, our model, EBILSTM-MH, can still achieve much better results with small data than those models. In addition, the accuracy score of the BiLSTM + multi-head_att (text only) method is lower than that of the EBILSTM-MH. This is because emoticons are an important reference condition for microblog sentiment analysis. Therefore, the method that ignores emoticons and only analyzes plain text gets worse results in accuracy and F1 evaluation function. When compared with the emj_att method, our model also achieves better results in accuracy and F1 values. The reason lies in that the multi-head_attention method repeatedly calculates the relationship between words and emoticons and extracts more features between them. More intuitively, the accuracy of EBILSTM-MH is improved by 8.51%, 4.89%, 3.89%, and 2.15% than that of the CNN, SVM, BiLSTM + multi-head_att (text only), and BiLSTM + emj_att models, respectively. From the perspective of categories, EBILSTM-MH achieves the best results in all three categories.

The impact of the head number: here, we mainly discuss the influence of numbers of head on the experimental results. Figure 5 shows the model's performance on the test set using different head-num in the attention mechanism and the value of head_num ranges from 7 to 12. With the increase of head_num, the performance of the model was continuously optimized at the beginning, but with the increase of abscissa value, the performance of the model began to be relatively stable, as can be seen from Figure 5 The continued increase of head_num would not benefit the experimental results much. Since the best result is reached when head_num is valued at 8, the model finally determines head_num to be 8 in our model.

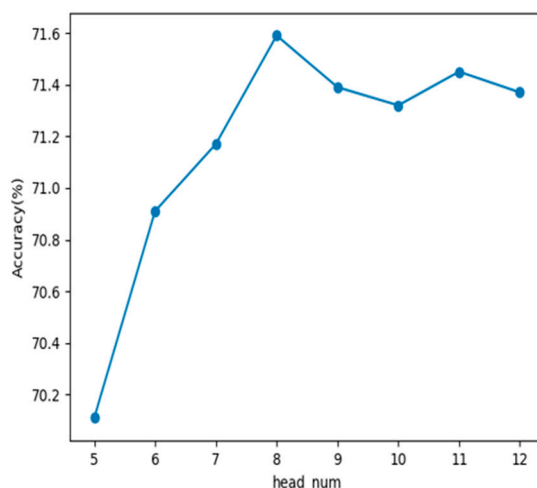


Figure 5. The accuracy score with different head_num.

5. Conclusions

In this paper, we introduced a BiLSTM and multi-head attention mechanism combined model, which realizes more accurate sentiment analysis for microblog with the help of the semantic enhancement of emoticons. We cleaned the original data, separated the emojis from the text, segmented words, preserved the associated structures and removed the stop words, and used the Word2Vec model to convert the text into a vector representation. Subsequently, we passed the word embedding to BiLSTM layer to extract the advanced features. In addition, we cited the multi-head attention mechanism to calculate the effect of combining emoticons and words in microblog. Through this mechanism, the weight of the importance of words to the sentiment analysis was calculated, and finally through the dense connection layer and softmax operation, the emotional polarity of microblog was obtained.

The experimental results show that, when compared with the simple sentiment analysis of the words in microblog, the multi-head attention mechanism can effectively point out the words that contribute more to sentiment analysis after the addition of emoticons. The multi-head attention mechanism can further improve the performance of the model with approximately the same computational time as using a single head attention mechanism. Additionally, in this task, EBILSTM-MH is better than the traditional deep learning model SVM and CNN. However, the number of emojis in the dataset also limits the effect of the method in this paper. If there are more emojis in the training set and test set, the final comparison result will be more prominent. In future work, we will continue to collect more real-time microblog texts for fine-grained sentiment analysis, and mine more features in texts in order to improve the performance of the model for sentiment analysis.

Author Contributions: Conceptualization, S.W. and W.G.; Formal analysis, W.G.; Funding acquisition, Y.Z.; Investigation, W.G., M.C. and M.L.; Project administration, Y.Z.; Resources, M.C. and M.L.; Software, S.W.; Supervision, Y.Z. and W.G.; Validation, S.W.; Writing—original draft, S.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Research and Development Plan of China, grant number 2017YFD0400101.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Parameters of the Baseline Methods

All microblogs in the experiment are represented by 200-dimensional word vectors with the maximum length of each sentence limited to 80 characters and the maximum number of emojis in each sentence limited to 5.

For the baseline model BiLSTM + emj_att, 128 and 64 hidden units are set in the LSTM layer and the attention layer, respectively, and the dropout is set to 0.4. After every 9 epochs, the learning rate will be updated to 40% of the previous learning rate. The same setting as that for BiLSTM + emj_att is used in LSTM layer and attention layer of BiLSTM + multi-head_att (text only), but the dropout rate of this baseline is set to 0.3, and every 6 epochs, the learning rate will be updated to 10% of the previous learning rate. We use a standard Adam optimizer to train these two deep neural network models with a batch size of 32, and the number of epochs as 30. Models get trained on the training dataset and its performance gets evaluated on the validation dataset after every epoch using the accuracy. If the performance of the model does not improve after 7 consecutive epochs, we will stop early and save the best model.

We use SVM model with linear kernel as one of the baselines and set the penalty coefficient (C) to 1.

We also used stacked Conv1D and MaxPooling1D to construct the baseline model CNN. The output channels of each convolutional layer are 256, with the kernel size of 3, 4, 5, respectively, strides of 1, and the activation function of ReLu. In the three pooling layers, the pooling sizes are 28, 27, and 26, respectively. In this baseline model, the number of epochs is set to 30 and the batch size is equal to 512. After every 5 epochs, the learning rate of the model will be updated to 50% of the previous learning rate. If the performance of the model does not improve after 7 consecutive epochs, we will stop early and save the best model.

References

1. Peng, H.; Cambria, E. *CSenticNet: A Concept-Level Resource for Sentiment Analysis in Chinese Language*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10762 LNCS; Springer: Berlin/Heidelberg, Germany, 2018; Volume 10762, pp. 90–104.
2. Hatzivassiloglou, V.; Mckeown, K.R. Predicting the Semantic Orientation of Adjectives. In Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, Madrid, Spain, 7–12 July 1997; pp. 174–181.
3. Liu, K.; Xu, L.; Zhao, J. Co-extracting opinion targets and opinion words from online reviews based on the word alignment model. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 636–650. [[CrossRef](#)]
4. Hao, Z.; Cai, R.; Yang, Y.; Wen, W.; Liang, L. A Dynamic Conditional Random Field Based Framework for Sentence-level Sentiment Analysis of Chinese Microblog. In Proceedings of the 20th IEEE International Conference on Computational Science and Engineering and 15th IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, CSE and EUC 2017, Guangzhou, China, 21–24 July 2017.
5. Rehman, Z.U.; Bajwa, I.S. Lexicon-based Sentiment Analysis for Urdu Language. In Proceedings of the 2016 Sixth International Conference on Innovative Computing Technology (INTECH 2016), Dublin, Ireland, 24–26 August 2016; pp. 497–501.
6. Che, W.; Zhao, Y.; Guo, H.; Su, Z.; Liu, T. Sentence Compression for Aspect-Based Sentiment Analysis. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 2111–2124. [[CrossRef](#)]
7. Yang, C.; Zhang, H.; Jiang, B.; Li, K. Aspect-based sentiment analysis with alternating coattention networks. *Inf. Process. Manag.* **2019**, *56*, 463–478. [[CrossRef](#)]
8. Zhou, X.; Wan, X.; Xiao, J. CMiner: Opinion Extraction and Summarization for Chinese Microblogs. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1650–1663. [[CrossRef](#)]

9. Ma, Y.; Peng, H.; Cambria, E. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive LSTM. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, LA, USA, 2–7 February 2018; pp. 5876–5883.
10. Peng, H.; Ma, Y.; Li, Y.; Cambria, E. Learning multi-grained aspect target sequence for Chinese sentiment analysis. *Knowl.-Based Syst.* **2018**, *148*, 167–176. [[CrossRef](#)]
11. Lu, Y.; Zhou, J.; Dai, H.-N.; Wangt, H.; Xiao, H. Sentiment Analysis of Chinese Microblog Based on Stacked Bidirectional LSTM. *IEEE Access* **2019**, *7*, 38856–38866.
12. Tai, Y.-J.; Kao, H.-Y. Automatic Domain-Specific Sentiment Lexicon Generation with Label. In Proceedings of the IJWAS'13: Proceedings of International Conference on Information Integration and Web-based Applications & Services, Vienna, Austria, 2–4 December 2013; pp. 53–62.
13. Su, M.-H.; Wu, C.-H.; Huang, K.-Y.; Hong, Q.-B. LSTM-based Text Emotion Recognition using semantic and emotional word vector. In Proceedings of the 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia), Beijing, China, 20–22 May 2018.
14. Yang, G.; He, H.; Chen, Q. Emotion-Semantic-Enhanced Neural Network. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 531–543. [[CrossRef](#)]
15. De Almeida, A.M.G.; Barbon, S.; Paraiso, E.C. Multi-class Emotions Classification by Sentic Levels as Features in Sentiment Analysis. In Proceedings of the 5th Brazilian Conference on Intelligent Systems (BRACIS), Pernambuco, Brazil, 2 February 2016.
16. Wang, J.; Yu, L.-C.; Lai, K.R.; Zhang, X. Community-Based Weighted Graph Model for Valence-Arousal Prediction of Affective Words. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2016**, *24*, 1957–1968. [[CrossRef](#)]
17. Huang, S.; Niu, Z.; Shi, C. Automatic Construction of Domain-specific Sentiment Lexicon Based on Constrained Label Propagation. *Knowl.-Based Syst.* **2014**, *56*, 191–200. [[CrossRef](#)]
18. Teng, Z.; Ren, F.; Kuroiwa, S. Recognition of emotion with SVMs. In Proceedings of the International Conference on Intelligent Computing, ICIC 2006, Kunming, China, 16–19 August 2006; pp. 701–710.
19. Xia, R.; Zong, C.; Li, S. Ensemble of feature sets and classification algorithms for sentiment-classification. *Inf. Sci.* **2011**, *181*, 1138–1152. [[CrossRef](#)]
20. Chen, T.; Xu, R.; He, Y.; Wang, X. Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Syst. Appl.* **2017**, *72*, 221–230. [[CrossRef](#)]
21. Meisheri, H.; Ranjan, K.; Dey, L. Sentiment Extraction from Consumer-generated Noisy Short Texts. In Proceedings of the 2017 IEEE International Conference on Data Mining Workshops (ICDMW), New Orleans, LA, USA, 18–21 November 2017.
22. Deng, D.; Jing, L.; Yu, J.; Sun, S. Sparse Self-Attention LSTM for Sentiment Lexicon Construction. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2019**, *27*, 1777–1790. [[CrossRef](#)]
23. Zhang, P.; Wang, J.; Wang, Y. Sentiment Lexicon Construction Method Based on Label Propagation. *Comput. Eng.* **2018**, *44*, 168–173.
24. Zhang, S.; Wei, Z.; Wang, Y.; Liao, T. Sentiment analysis of Chinese microblog text based on extended sentiment dictionary. *Future Gener. Comput. Syst.* **2018**, *81*, 395–403. [[CrossRef](#)]
25. Chen, J.; Huang, H.; Tian, S. Feature selection for text classification with Naïve Bayes. *Expert Syst. Appl.* **2009**, *36*, 5432–5435. [[CrossRef](#)]
26. Pang, B.; Lillian, L.; Vaithyanathan, S. Thumbs up? Sentiment Classification using Machine Learning Techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Philadelphia, PA, USA, 6–7 July 2002; pp. 79–86.
27. Liu, X.; Wu, Q.; Pan, W. Sentiment classification of microblog comments based on Random forest algorithm. *Concurr. Comput.-Pract. Exp.* **2019**, *31*, e4746. [[CrossRef](#)]
28. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. In Proceedings of the 1st International Conference on Learning Representations, ICLR 2013, Scottsdale, AZ, USA, 2–4 May 2013.
29. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.; Dean, J. Distributed Representations of Words and Phrases and their Compositionality. *arXiv* **2013**, arXiv:1310.4546.
30. Sun, Y.; Lin, L.; Yang, N.; Ji, Z.; Wang, X. *Radical-Enhanced Chinese Character Embedding*; Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer: Berlin/Heidelberg, Germany, 2014; Volume 8835, pp. 279–286.

31. Peng, H.; Cambria, E.; Zou, X. Radical-based hierarchical embeddings for Chinese sentiment analysis at sentence level. In Proceedings of the FLAIRS 2017—Proceedings of the 30th International Florida Artificial Intelligence Research Society Conference, Sarasota, FL, USA, 19–22 May 2019; pp. 347–352.
32. Wang, M.; Hu, G. A novel Method for Twitter Sentiment Analysis Based on Attention-Graph Neural Network. *Information* **2020**, *11*, 92. [[CrossRef](#)]
33. Irsoy, O.; Cardie, C. Opinion Mining with Deep Recurrent Neural Networks. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 720–728.
34. Schlichtkrull, M.S. Learning affective projections for emoticons on Twitter. In Proceedings of the 6th IEEE Conference on Cognitive Infocommunications, CogInfoCom 2015, Szechenyi Istvan University, Gyor, Hungary, 19–21 October 2015; pp. 539–543.
35. Vaswani, A.; Shazeer, N.; Parmar, N. Attention Is All You Need. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems, NIPS 2017, Long Beach, FL, USA, 4–9 December 2017.
36. Bahdanau, D.; Cho, K.; Bengio, Y. Neural Machine Translation by Jointly Learning to Align and Translate. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, FL, USA, 7–9 May 2015.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).