

Received October 24, 2019, accepted November 27, 2019, date of publication December 4, 2019, date of current version December 23, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2957510

# Self-Attention-Based BiLSTM Model for Short Text Fine-Grained Sentiment Classification

JUN XIE<sup>1</sup>, BO CHEN<sup>1</sup>, XINGLONG GU<sup>1</sup>, FENGMEI LIANG<sup>1</sup>, AND XINYING XU<sup>2</sup>

<sup>1</sup>College of Information and Computer, Taiyuan University of Technology, Jinzhong 030600, China

<sup>2</sup>College of Electrical and Power Engineering, Taiyuan University of Technology, Taiyuan 030024, China

Corresponding author: Xinying Xu (xuxinying@tyut.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61503271 and Grant 61603267, and in part by the Natural Science Foundation of Shanxi Province under Grant 201801D121144 and Grant 201801D221190.

**ABSTRACT** Fine-grained sentiment polarity classification for short texts has been an important and challenging task in natural language processing until these years. The short texts may contain multiple aspect-terms, opinion terms expressing different sentiments for different aspect-terms. The polarity of the whole sentence is highly correlated with the aspect-terms and opinion terms. Besides, there are two challenges, which are how to effectively use the contextual information and the semantic features, and how to model the correlations between aspect-terms and context words including opinion terms. To solve these problems, a Self-Attention-Based BiLSTM model with aspect-term information is proposed for the fine-grained sentiment polarity classification for short texts. The proposed model can effectively use contextual information and semantic features, and especially model the correlations between aspect-terms and context words. The model mainly consists of a word-encode layer, a BiLSTM layer, a self-attention layer and a softmax layer. Among them, the BiLSTM layer sums up the information from two opposite directions of a sentence through two independent LSTMs. The self-attention layer captures the more important parts of a sentence when different aspect-terms are input. Between the BiLSTM layer and the self-attention layer, the hidden vector and the aspect-term vector are fused by adding, which reduces the computational complexity caused by the vector splicing directly. The experiments on public Restaurant and Laptop corpus from the SemEval 2014 Task 4, and Twitter corpus from the ACL 14. The Friedman and Nemenyi tests are used in the comparison study. Compared with existing methods, experimental results demonstrate that the proposed model is feasible and efficient.

**INDEX TERMS** Aspect-term, bidirectional LSTM (BiLSTM), fine-grained, self-attention.

## I. INTRODUCTION

The task of sentiment polarity classification is also regarded as opinion mining [1]. Fine-grained sentiment polarity classification for short texts is an important task in the sentiment analysis, which has obtained much attention for these years. It can get more precise and in-depth sentiment polarity for each aspect-term in the sentence. It is known that the sentiment polarity depends on the context and the aspect-term, for example, the sentiment polarity of “*The cake is pretty delicious, but staffs are not friendly.*” will be positive when considering the aspect-term “*cake*”, however, if the aspect-term is “*staffs*”, the polarity is negative. Through this example it can be intuitively found that the sentiment polarity could be different if different aspect-terms are considered.

The associate editor coordinating the review of this manuscript and approving it for publication was Fatos Xhafa<sup>1</sup>.

Otherwise, when considering the large-scale texts, a few of aspect-terms can be regarded as important memes which can be found in the text [2]. There exists Matthew effect in the aspect-terms which can be interpreted as the textual sentiment extension [3].

Neural networks are widely used in many natural language processing tasks, such as machine translation [4], paraphrase identification [5], question answering [6], and sentence summarization [7]. In these tasks, neural networks have achieved a state-of-the-art performance. However, neural networks are still in infancy to solve sentiment analysis tasks. Binary sentiment polarity classification can be benefited from using recurrent neural networks [8], however, this model is only suitable for shorter sentences. The emergence of LSTM has solved this problem well [9]. In some works, the target information is considered for the target dependent sentiment polarity classification, for example, Target-Dependent LSTM

model and Target-Connection LSTM model [10]. However, those models only capture the historical information about sentences and cannot make full use of the contextual information, so that each word cannot also achieve the more precise semantic information, and cannot find the highlight words that have the greater influence on the sentiment polarity in sentences. Besides, these models do not enough use aspect-term information, which is vital to the short text fine-grained sentiment polarity classification, and the more completely the aspect-term information transfers, the more greatly the overall level of the semantic relation is promoted [11].

The attention mechanism is firstly applied in image recognition domain [12], [13], and has achieved pretty good results. Now the attention mechanism is used widely in natural language processing tasks, such as machine translation [14], text summarization [15]. Attention is regarded as an effective mechanism, and has been used in the fine-grained sentiment polarity classification tasks. Tang. built an attention-based deep memory network [16]. Wang. proposed attention-based LSTM and attention-based LSTM with aspect embedding [17]. Ashish Vaswani. designed self-attention which was different from the encoder-decoder network based on the attention mechanism in the domain of machine translation [18]. The self-attention mechanism can focus on the important parts in the sentence itself, which is a way that the more relevant the semantic relation between words and sentiment polarity is, the greater the weight of the connection between them is. These connected nodes acquiring more critical information assist the model to determine more accurate sentiment polarity [19].

In the article, the sentiment polarity of each aspect-term is identified in the short texts. For this task, although some effective models have been proposed, there still exist some problems as how to effectively use contextual information and semantic features and how to model the correlations between aspect-terms and context words. To settle the problem mentioned, the main works of the article can be summarized as follows:

1. Self-Attention-Based BiLSTM model with aspect-term information is proposed for the fine-grained sentiment polarity classification. The model can not only make full use of contextual information, but also can capture important parts of the sentence itself through a simple and effective self-attention layer. Experiments prove that the proposed model is effective, seen from FIGURE 7 to FIGURE 11.

2. Since the short text may contain multiple aspect-terms, which are vital to the short text sentiment polarity classification, the hidden vector is combined with aspect-term information to model the correlations between aspect-terms and context words, and make the fusion vector as an input to self-attention layer, computing the attention weights, seen in FIGURE 6.

3. The dimension of the word vector affects the results of the experiment, where the dimension is set as {50, 100, 200, 300} respectively. The best suitable dimension was found as from TABLE 3 to TABLE 5.

The rest parts of the article are arranged as follows: Section 2 described the related work about the sentiment polarity classification. Section 3 proposed the details of Self-Attention-Based BiLSTM model and Self-Attention-Based BiLSTM with aspect-term information. Section 4 did the experiment setting and analyzed experimental results. The conclusion came to the Section 5.

## II. RELATED WORK

Fine-grained sentiment polarity classification is a branch of sentiment analysis tasks, which can obtain more precise sentiment polarity for each aspect-term [20]. Deep learning models have achieved pretty good performance in sentiment classification [21]–[24]. Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) are two main neural network models used in the sentiment polarity classification, where the word embedding is applied into the model to complete sentence classification [25], [26]. RNN model can solve the problem of long-distance dependence of sentences [27], [28]. Because of the gradient exploring and gradient vanishing of RNN, LSTM model is used to replace the RNN model for remembering the sentences [29].

Fine-grained sentiment polarity classification task has also attracted much attention over these years. A lot of work in this task has been conducted. Previous work always brings about a huge number of labor work and extra lexicon to exact features, which could be an enormous and complicated project. Some machine learning methods, such as Support Vector Machine, Naïve Bayes, which have been used in the text sentiment polarity classification field [21]. These methods usually capture sentiment features in the sentence to establish the model, then the model predicts classes of the test text. Machine learning methods have gotten many better results in the text sentiment polarity classification. However, machine learning methods also have obvious shortcomings that regard the text as a bag of words, and cannot consider which relation between word and word. Sometimes the sentiment polarity classification is determined by the relations. Besides, machine learning methods require a lot of feature annotations on the text, and need a quantity of manual pre-processing, which prevent them from being more efficient or precise.

Deep learning has achieved pretty good performance in many natural language processing tasks. RNN treats the sentence as a word sequence, and considers the context when modeling the sentence [22]. Dong. proposed a self-adaptive recursive neural network, which can learn the sentiment relationships between word and word by using the sentiment structure and context information [23]. But RNN did not work well when modeling long sentences. LSTM model was proposed to solve this problem [24], which had ability to preserve the sequence information, and obtained strong results on some sequence modeling tasks. Tai. proposed tree-LSTM model, which was a generalization of LSTM to tree-structured network topologies. Tree-LSTM outperformed all existing LSTM baselines in sentiment

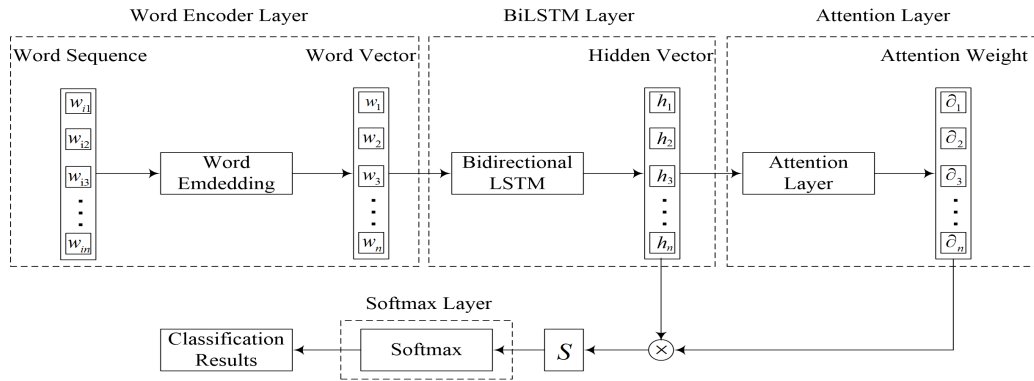


FIGURE 1. The basic framework of the Self-Attention-Based BiLSTM neural network model.

analysis [25]. To get in-depth results in sentiment analysis. Tang. proposed Target-Dependent LSTM model, which considers the target information for the target sentiment analysis and got state-of-the-art performance [10].

For recent years, attention mechanisms have gradually been used in the field of natural language processing, and have achieved good results in fine-grained sentiment analysis tasks [7], [35], [36]. Tang. built an attention-based deep memory network [16]. Wang. proposed attention-based LSTM and attention-based LSTM with aspect embedding [17]. Ashish Vaswani. designed self-attention which was different from the encoder-decoder network based on the attention mechanism in the domain of machine translation [18].

In the article, a self-attention mechanism is introduced into the BiLSTM neural network model to improve the ability to capture the key information for each aspect-term. A Self-Attention-Based BiLSTM model with aspect-term information is also proposed to solve the short text contains multiple aspect-terms.

### III. SELF-ATTENTION-BASED BiLSTM NEURAL NETWORKS MODEL

In this section, the Self-Attention-based BiLSTM neural networks model is proposed for the sentiment polarity classification for short texts. The proposed model consists of a word encoder layer, a BiLSTM layer, an attention layer and a softmax layer. The basic framework of Self-Attention-Based BiLSTM model is shown in FIGURE 1. The details of each component of the model are depicted as follows.

#### A. WORD ENCODER LAYER

The sentence consists of a sequence of tokens  $s_i = \{w_{i1}, w_{i2}, \dots, w_{ik}, \dots, w_{in}\}$ , where  $w_{ik}$  denotes the  $k$ th word in the  $i$ th sentence and  $n$  represents the length of the sentence. Each word in the sentence is converted into a  $d$ -dimension vector, which is called as the word embedding [26]. After all words in a sentence are represented by a  $d$ -dimension vector, the word embedding matrix  $S^{n \times d}$  is constructed, where  $n$  denotes the length of sentence and  $d$  represents the embedding size. The word embedding matrix is regarded as

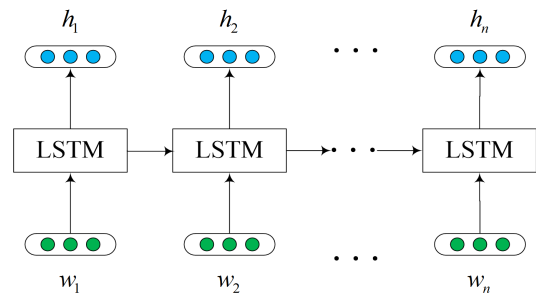


FIGURE 2. The architecture of a basic LSTM.

parameters of the neural network model. After that, the word embedding matrix is regarded as the input for the BiLSTM neural network model aiming to encode a sentence.

#### B. BiLSTM LAYER

LSTM is an extension of RNN, which is good at solving the gradient vanishing or exploding problems in standard RNN. The LSTM neural networks contain three gates and a cell memory state. FIGURE 2 shows the basic structure of a standard LSTM, where  $\{w_1, w_2, \dots, w_n\}$  denotes the word vector in a sentence,  $\{h_1, h_2, \dots, h_n\}$  represents the hidden vector.

The key to LSTM is the cell state, as for a single LSTM cell, it can be computed as follows:

$$X = \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} \tag{1}$$

$$f_t = \sigma(W_f \cdot X + b_f) \tag{2}$$

$$i_t = \sigma(W_i \cdot X + b_i) \tag{3}$$

$$o_t = \sigma(W_o \cdot X + b_o) \tag{4}$$

$$c_t = f_t * c_{t-1} + i_t * \tanh(W_c \cdot X + b_c) \tag{5}$$

$$h_t = o_t * \tanh(c_t) \tag{6}$$

where  $W_f, W_i, W_o$  are the weight matrices and  $b_f, b_i, b_o$  are the bias of LSTM cell during training, which are regarded as parameters of the input gates, forget gates and output gates respectively.  $\sigma$  denotes the sigmoid function and

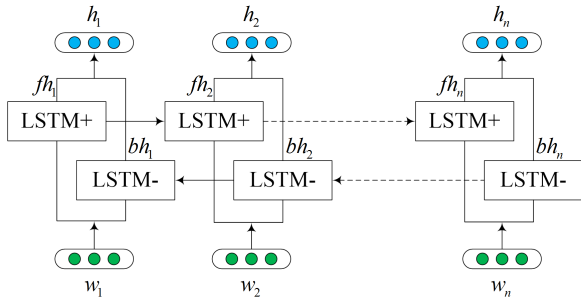


FIGURE 3. The architecture of a basic BiLSTM.

\* represents the element-wise multiplication.  $x_t$  stands for the word embedding as input unit to LSTM,  $h_t$  is the hidden vector, so  $h_n$  can denote a sentence.

In the proposed approach, the BiLSTM is constituted of two independent LSTMs, which can sum up information from forward and backward direction of a sentence, and then can merge the information coming from the two directions. Especially, at each time  $t$ , the forward LSTM computes the hidden vector  $fh_t$  based on the previous hidden vector  $fh_{t-1}$  and the input word embedding  $x_t$ , and the backward LSTM computes the hidden vector  $bh_t$  based on the opposite previous hidden vector  $bh_{t-1}$  and the input word embedding  $x_t$ . Afterwards, the forward hidden vector  $fh_t$  and the backward hidden vector  $bh_t$  are merged into the final hidden vector of the BiLSTM model. In BiLSTM model, the parameters of two opposite directions are independent, but they share the same word embedding of a sentence. FIGURE 3 illustrates the basic structure of the BiLSTM model, where  $\{w_1, w_2, \dots, w_n\}$  denotes the word vector,  $n$  is the length of a sentence.  $\{fh_1, fh_2, \dots, fh_n\}$  and  $\{bh_1, bh_2, \dots, bh_n\}$  represent the forward hidden vector and the backward hidden vector respectively.  $h_n$  stands for the vector connected by  $fh_n$  and  $bh_n$ .

The final hidden vector  $h_t$  of the BiLSTM is shown as following equation:

$$h_t = [fh_t, bh_t] \quad (7)$$

### C. ATTENTION LAYER

The sentiment polarity of a sentence is not only related to the contextual information, but also has a highly correlation with the opinion terms and the aspect-terms. But given a sentence, not all context words have the equal contribution to the semantics of a sentence. To address this issue, the self-attention mechanism is used to extract these more important words by giving them a higher weight to increase their importance.

Specifically, the BiLSTM neural networks will produce a hidden vector  $h_t$ . Firstly, the hidden vector  $h_t$  as an input is fed into a simple Multi-Layer Perceptron to get a new hidden representation  $u_t$ . Then a weight value that symbolizes the importance of words is calculated for  $h_t$  given  $u_t$  and a word-level context vector  $u_w$  [27]. The context vector  $u_w$

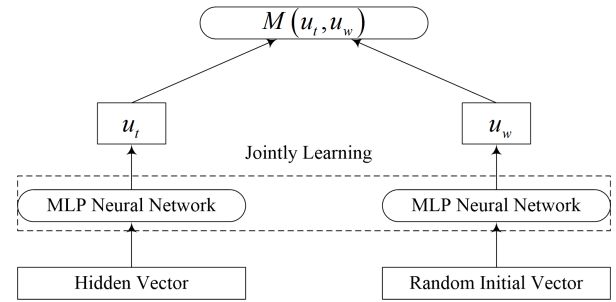


FIGURE 4. The basic process of self-attention mechanism.

is considered as a high dimensional representation to judge the importance of different words in the sentence, which is randomly initialized and jointly learned in the process of training. Finally, the weighted mean of the hidden vector  $h_t$  is computed by a softmax function. FIGURE 4 shows the basic process of self-attention mechanism, where  $M$  represents a function that computes the similarity between  $u_t$  and  $u_w$ . And for each step, these equations can be shown as follows:

$$u_t = \tanh(W_w h_t + b_w) \quad (8)$$

$$\partial_t = \frac{\exp(u_t^T u_w)}{\sum_t \exp(u_t^T u_w)} \quad (9)$$

$$s = \sum_t \partial_t h_t \quad (10)$$

### D. SOFTMAX LAYER

In the proposed model, the softmax layer is used as a classifier. The model produces a high-level representation for a sentence. The hidden vector of each word multiplies its corresponding weight to get the vector  $s$ , and the vector  $s$  is regarded as the sentiment feature for the sentiment polarity classification:

$$\tilde{y} = \text{soft max}(W_s s + b_s) \quad (11)$$

where  $\tilde{y}$  is the predicted result through the model,  $W_s$  is the weighted matrix, and  $b_s$  is the bias.

## IV. SELF-ATTENTION-BASED BILSTM MODEL FOR SHORT TEXT SENTIMENT CLASSIFICATION

The LSTM neural network has a good performance for representing semantic composition of a sentence, which can better capture longer distance dependencies. However, LSTM neural networks can only capture the forward parts of a sentence. Sometimes the semantics of a word in a sentence cannot be correctly represented only by the historical information of the sentence. For example, “Not only was the food outstanding, but the little ‘perks’ were great.”, “not” is a negative word. But by analyzing the second part of the sentence, it can be found that the word does not represent negation. Therefore, the bidirectional LSTM formed by two

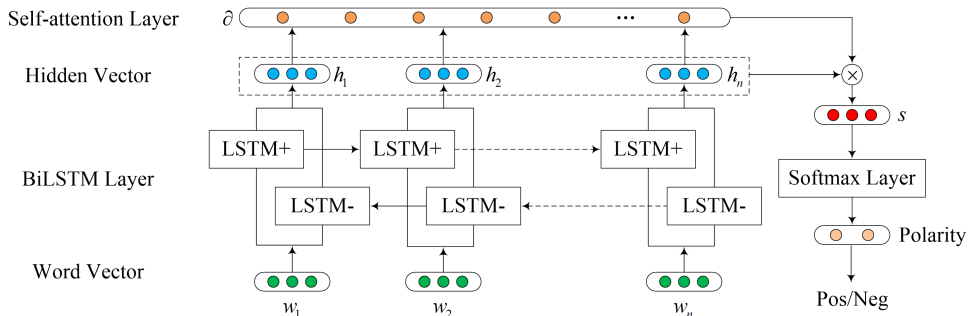


FIGURE 5. The architecture of a Self-Attention-Based BiLSTM neural network model.

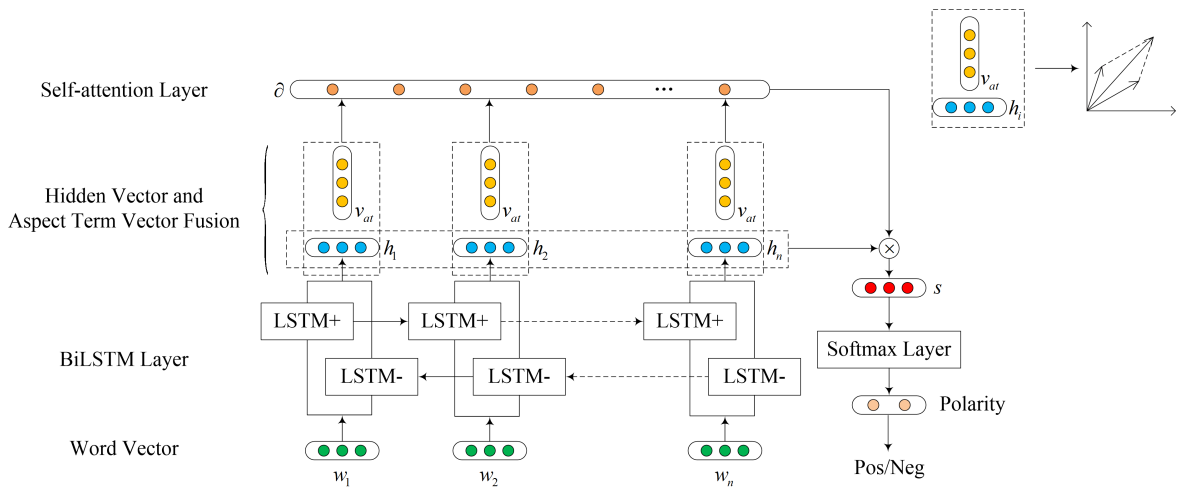


FIGURE 6. The architecture of a Self-Attention-Based BiLSTM model with aspect-term information.

independent LSTMs is applied, which can capture information from two opposite directions of a sentence to make full use of the contextual information. In this case, the semantic of the currently input word could be denoted more accurately.

Mentioned above, the sentiment polarity of a sentence has highly correlated with the aspect-terms and the opinion terms in the sentence. Therefore, how to concentrate on these aspect-terms is very important in the sentiment analysis task. However, the standard BiLSTM cannot recognize which is the more important part for the sentiment analysis. To solve this problem, a simple and effective self-attention mechanism is introduced that can capture the important part of a sentence. FIGURE 5 shows the architecture of the Self-Attention-Based BiLSTM neural network model, where the  $\{w_1, w_2, \dots, w_n\}$  denotes the word embedding of a sentence, and the length is  $n$ .  $\{h_1, h_2, \dots, h_n\}$  is a hidden vector.  $\delta$  is an attention weight.  $s$  denotes a sentence vector which can be regarded as the sentiment feature with attention weight.

Aspect-term information is critical for the fine-grained sentiment polarity classification of the short text. If different aspect-terms are given, the different sentiment polarity can be obtained. For example, “The cake tastes deliciously, the

service of restaurant is terrible.”. For aspect-term “cake” and “service”, the corresponding sentiment polarity is opposite completely. To make full use of the aspect-term information, an embedding vector is trained for each given aspect-term. If the aspect-term consists of multiple words, such as “restaurant service”, the aspect-term is represented by an average of its consisting word vectors. Suppose the aspect-term is composed of  $m$  words  $\{e_1, e_2, \dots, e_m\}$ , the aspect-term vector is represented in the formula (12). And then each hidden vector  $h_i$  is fused with the aspect-term vector  $v_{at}$  of each given aspect-terms as an input to the attention layer to learn an attention weight. The fusion vector  $f_i$  is computed by the formula (13). FIGURE 6 shows the basic structure of the Self-Attention-Based BiLSTM model with aspect-term information, where the  $\{w_1, w_2, \dots, w_n\}$  denotes the word embedding of a sentence, and the length is  $n$ .  $\{h_1, h_2, \dots, h_n\}$  is a hidden vector.  $v_{at}$  denotes the aspect-term vector.  $\delta$  is an attention weight.  $s$  denotes a sentence vector which can be regarded as the sentiment features with attention weight.

$$v_{at} = \frac{1}{m} \sum_{i=1}^m e_i \tag{12}$$

$$f_i = h_i + v_{at} \tag{13}$$

**Algorithm 1** Self-Attention-Based BiLSTM Algorithm

- Step1:** Through Glove word vector, converting word sequence  $\{w_{i1}, w_{i2}, \dots, w_{in}\}$  into corresponding word vectors  $\{w_1, w_2, \dots, w_n\}$ .
- Step2:** Modeling sentences using two separate and opposite LSTM models, according to formulas (1) ~ (6), and learning the hidden vectors  $\{h_1, h_2, \dots, h_n\}$  for each word.
- Step3:** Obtaining the attention weight. 1) Feeding the hidden vector and random initial vector into a simple Multi-Layer Perceptron for jointly learning, getting a new hidden representation  $u_t$  and a context vector  $u_w$ , through the formula (9) computing the attention weight  $\partial_t$  of each word. 2) Fusing the hidden vector and aspect-term vector, similar to 1), feeding the fusion vector and random initial vector into Multi-Layer Perceptron, getting another hidden representation  $u'_t = \tanh [W'_w (h_t + v_{at}) + b'_w]$  and context vector  $u'_w$ , then using  $u'_t$  and  $u'_w$  to calculate the attention weight  $\partial'_t$  for each word by the formula (9).
- Step4:** Constructing a high-level sentence vector. 1) Without fusing aspect-term vector, the sentence vector is calculated through weighted sum of attention weight and corresponding hidden vector. 2) Fusing aspect-term vector, as 1), the sentence vector is computed by the formula  $s' = \sum_t \partial'_t h_t$ .
- Step5:** Adopting the sentence vector  $s, s'$  obtained in step4 as the sentiment feature for sentiment polarity classification, then the softmax layer output the value  $\tilde{y}$ , which is the predicted value by the proposed model.

**A. MODEL TRAINING**

In the process of the model training, it was used back propagation method [28], by calculating the error term value for each neuron in reverse. Like the recurrent neural network, the backpropagation of the BiLSTM error term includes two directions: one is the back propagation along time, that is, from the current  $t$  time, the error term at each moment is calculated, the other is to move the error term up one layer spread. The gradient of each weight is calculated based on the corresponding error term. Finally, the parameters are updated by the stochastic gradient descent algorithm. The cross entropy error is applied as the object function (loss function):

$$E = - \sum_i y_i \log \tilde{y}_i \quad (14)$$

where  $y_i$  is the basic sentiment polarity label and  $\tilde{y}_i$  is the result label predicting from the model. Like the standard LSTM model,  $\{W_i, b_i, W_f, b_f, W_o, b_o, W_c, b_c, W_s, b_s\}$  is used as a parameter set. At the same time, word embedding are also the parameters.

**V. EXPERIMENT AND ANALYSIS****A. EXPERIMENT SETTING**

In the article, the BiLSTM model is implemented through Pytorch, which is a pretty popular deep learning framework.

In the experiments, all word vectors are initialized by Glove<sup>1</sup> (Pennington et al., 2014). In order to find a suitable word vector dimension, therefore, in the process of training the word vector, the dimension of the word vector is set to  $\{50, 100, 200, 300\}$  respectively.

As for the BiLSTM neural network model, its parameters are vital to an effective result. The size of hidden layer is 50. In order to prevent overfitting problems, the dropout rate is set to 0.1. The batch size is 16 and the learning rate is 0.01. All

<sup>1</sup>Pre-trained word vectors of Glove can be obtained from <https://nlp.stanford.edu/projects/glove/>

**TABLE 1.** The distribution of aspects under each sentiment polarity classification on different datasets.

Dataset	Positive		Negative		Neural	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	807	196	637	196
Laptop	994	341	870	128	464	169
Twitter	1561	173	1560	173	3127	346

models were trained for 15 epochs in the experiments. The other parameters are initialized through the uniform distribution  $[-0.1, 0.1]$ . The dimensions of aspect embedding need to be consistent with the dimensions of each hidden vector. The reason is that the aspect-term vector and each hidden vector are combined as an input to attention layer to get attention weight. Finally, the attention dimension is set to 100, and the attention weights are obtained with their length as the same as the length of sentence. In order to better measure the performance of our model, the 10-fold cross-validation method was used to train the model, and finally take the average score of ten times cross-validation to obtain a more stable evaluation result.

**B. DATASET**

The experiments were conducted on three datasets: SemEval 2014 Task 4<sup>2</sup> (Pontiki et al., 2014) datasets composed of Restaurant corpus and Laptop corpus, and ACL 14 Twitter datasets. Every review includes a list of aspect-terms and is labeled with three sentiment polarities: *positive*, *neural* and *negative*. The statistic of datasets is shown in TABLE 1, and the balanced test datasets with equal class is shown in TABLE 2, the duplicate data mechanism was applied to make the classes equal.

<sup>2</sup>The introduction about SemEval 2014 can be obtained from <http://alt.qcri.org/semeval2014/>

**TABLE 2.** The balanced training datasets with equal classes, duplicated datasets are in bold.

Dataset	Positive		Negative		Neural	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	<b>1614</b>	196	<b>1911</b>	196
Laptop	994	341	870	128	<b>928</b>	169
Twitter	1561	173	1560	173	<b>1564</b>	346

**TABLE 3.** Accuracy on Restaurant. The sentiment polarity includes 3 classes {Positive, Negative, Neural}. Top scores are in bold.

Models	Dimensions			
	50	100	200	300
Model1	0.6793	0.6858	0.7084	0.7143
Model2	0.6964	0.7034	0.7123	0.7230
Model3	0.7021	0.7113	0.7281	0.7388
Model4	0.7172	<b>0.7294</b>	0.7360	0.7236
<b>Model5</b>	<b>0.7219</b>	0.7241	<b>0.7453</b>	<b>0.7446</b>

**C. EVALUATION METRIC**

The accuracy is applied as the evaluation metric, which is defined as follow:

$$Accuracy = \frac{T}{N} \tag{15}$$

where  $T$  is the correct number of predicted samples,  $N$  is the total number of tested samples.

And another evaluation metric is the confusion matrix, which is a table that summarizes the prediction results of the classification model. The records are summarized in a matrix form according to the real and predicted category, where the rows of the matrix represent the true values and the columns of the matrix represent the predicted values.

**D. COMPARISON WITH DIFFERENT METHODS**

The Glove word2vec converts each word of a sentence to a word vector, which is regarded as an input to the BiLSTM model. After that, the hidden vector is obtained for each word vector of a sentence, the hidden vectors contain semantic information about the word in the sentence, and through the BiLSTM model, it makes full use of the context information. Secondly, the given aspect-terms in every review are converted into the vectors, the hidden vector and aspect-term vector are synthesized as the input of the attention layer, and the corresponding attention weights are obtained. This is more beneficial to identify the sentiment polarity of the corresponding aspect-terms.

The proposed Self-Attention-Based BiLSTM with aspect-term information (Model5) is compared with LSTM (Model1), BiLSTM (Model2), Target-Dependent LSTM (Model3) and Self-Attention-Based BiLSTM (Model4). The results of experiment are shown in following tables. TABLE 3 to TABLE 5 show the accuracy with different word vectors on different datasets. Owing to the best performance on three classes of sentiment polarity task is almost at 300-D.

**TABLE 4.** Accuracy on Laptop. The sentiment polarity includes 3 classes {Positive, Negative, Neural}. Top scores are in bold.

Models	Dimensions			
	50	100	200	300
Model1	0.5470	0.5436	0.5693	0.5824
Model2	0.5508	0.5572	0.5840	0.5966
Model3	0.5705	0.5963	0.6361	0.6236
Model4	0.6022	0.6215	0.6367	0.6557
<b>Model5</b>	<b>0.6317</b>	<b>0.6441</b>	<b>0.6616</b>	<b>0.6818</b>

**TABLE 5.** Accuracy on Twitter. The sentiment polarity includes 3 classes {Positive, Negative, Neural}. Top scores are in bold.

Models	Dimensions			
	50	100	200	300
Model1	0.6098	0.6103	0.6251	0.6468
Model2	0.6174	0.6241	0.6462	0.6596
Model3	0.6012	0.6233	0.6694	0.6715
Model4	0.6195	<b>0.6407</b>	<b>0.6839</b>	0.6696
<b>Model5</b>	<b>0.6388</b>	0.6393	0.6708	<b>0.6952</b>

**TABLE 6.** Accuracy on different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Positive, Negative}. Top scores are in bold.

Models	Datasets		
	Restaurant	Laptop	Twitter
Model1	0.8017	0.8133	0.8136
Model2	0.8079	0.8393	0.8121
Model3	0.8287	0.8465	0.8280
Model4	0.8333	0.8516	<b>0.8396</b>
<b>Model5</b>	<b>0.8369</b>	<b>0.8689</b>	0.8379

**TABLE 7.** Accuracy on different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Positive, Neural}. Top scores are in bold.

Models	Datasets		
	Restaurant	Laptop	Twitter
Model1	0.8220	0.8016	0.7064
Model2	0.8297	0.8000	0.7167
Model3	0.8398	0.8078	0.7253
Model4	0.8555	0.8118	0.7437
<b>Model5</b>	<b>0.8624</b>	<b>0.8375</b>	<b>0.7754</b>

Therefore, the same 300-D word vectors are set in the experiments of two classes of sentiment polarity task on different datasets. The accuracy results are shown from TABLE 6 to TABLE 8.

**E. COMPARISON WITH EQUAL CLASSES**

The imbalance between different classes is an important factor that constrains the accuracy of many classification algorithms. Many classifiers tend to classify the corpus into majority class, thus reducing the accuracy of classification.

**TABLE 8.** Accuracy on different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Negative, Neural}. Top scores are in bold.

Models	Datasets		
	Restaurant	Laptop	Twitter
Model1	0.7673	0.6465	0.8035
Model2	0.7749	0.6531	0.8073
Model3	0.7726	0.6653	<b>0.8173</b>
Model4	0.7877	0.6731	0.8035
<b>Model5</b>	<b>0.7914</b>	<b>0.6768</b>	0.8155

**TABLE 9.** Accuracy on balanced datasets with 300-D word vector. The sentiment polarity includes 3 classes {Positive, Negative, Neural}. Top scores are in bold.

Models	Datasets		
	Restaurant	Laptop	Twitter
Model1	0.7252	0.6245	0.6551
Model2	0.7230	0.6717	0.6637
Model3	0.7429	0.6472	0.6717
Model4	0.7361	0.6464	0.6868
<b>Model5</b>	<b>0.7514</b>	<b>0.6655</b>	<b>0.6979</b>

**TABLE 10.** The different Model take time under different word embedding dimensions on raw datasets. (Unit: second).

Models	Dimensions			
	50	100	200	300
Model1	34	43	49	65
Model2	41	48	55	73
Model3	49	52	59	76
Model4	49	59	67	83
<b>Model5</b>	53	62	75	89

The corpus was duplicated directly with minority class by oversampling. The results are shown in TABLE 9.

**F. EFFECTS OF WORD EMBEDDINGS DIMENSION**

The dimension affects the quality of word embedding, and a good word embedding is vital to producing a powerful text representation at higher level. Therefore, it is necessary to study the influence of different dimensions of the word embedding on LSTM (Model1), BiLSTM (Model2), Target-Dependent LSTM (Model3), Self-Attention-Based BiLSTM (Model4) and Self-Attention-Based BiLSTM with aspect-term information (Model5). In the experiment, the dimension is set to {50, 100, 200, 300} respectively, and run them on the same GPU.<sup>3</sup> The data in TABLE 10 shows the time spent by different models in different dimensions.

The experimental results show that different dimensions {50, 100, 200, 300} have different results, and 100-dimension word vectors have a better performance than 50-dimension

<sup>3</sup>NVIDIA TITAN Xp

**TABLE 11.** Ranks of 5 models.

Dataset	Model1	Model2	Model3	Model4	<b>Model5</b>
Restaurant	4.5	4.5	3	2	1
Laptop	5	3.5	3.5	2	1
Twitter	5	4	3.5	2	1
<b>Mean Rank</b>	<b>4.8</b>	<b>4</b>	<b>3.2</b>	<b>2</b>	<b>1</b>

word vectors, while 300-dimension word vectors achieve the best performance than other dimensional word vectors. And as the dimension of the word vector increases, the cost time of the model increases too.

**G. STATISTICAL TESTS IN DIFFERENT METHODS**

In order to compare the performance differences between different models, statistical tests were performed on five models with three datasets. First, 10-fold cross-validation was used to get the accuracy of each model on 3 datasets, and then rank the accuracy. If the accuracy of the model is the same, the rank is evenly divided. The model comparison rank value is shown in TABLE 11. Then, the Friedman test was used to determine if the performance of these models is the same. The assumption was made that “all models have the same performance”, and the statistic  $\tau_F$  obeys the F distribution with degrees of freedom  $(k - 1)$  and  $(k - 1)(N - 1)$ .

$$\tau_F = \frac{(N - 1)\tau_{\chi^2}}{N(k - 1) - \tau_{\chi^2}} \tag{16}$$

$$\tau_{\chi^2} = \frac{12N}{k(k + 1)} \left( \sum_{i=1}^k r_i^2 - \frac{k(k + 1)^2}{4} \right) \tag{17}$$

Through checking the critical table of the F test,<sup>4</sup> the assumption that “all models have the same performance” is rejected, which means that all models have significantly different performance. Then we use the Nemenyi test<sup>5</sup> to further distinguish the models. The critical value of Nemenyi test is defined as follow:

$$CD = q_{\alpha} \sqrt{\frac{k(k + 1)}{6N}} \tag{18}$$

The comparison results of Friedman and Nemenyi test can be visually represented as FIGURE 7. In FIGURE 7, the vertical axis shows each model, and the horizontal axis is the mean rank value. For each model, the center point represents its mean rank value, and the length of horizontal line represents the size of the critical value range. It can be observed from FIGURE 7 if two horizontal line are not overlapping, it means that there is a significant difference between the two models, otherwise there is no significant difference.

**H. ANALYSIS**

TABLE 3 to TABLE 5 clearly show the experimental results of three polarity classification, TABLE 6 to TABLE 8 show

<sup>4</sup> $F_{\alpha=0.05} = 3.8379(k = 5, N = 3)$

<sup>5</sup> $q_{\alpha=0.05} = 2.728(k = 5)$



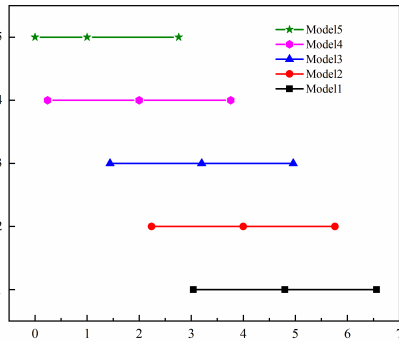


FIGURE 7. Friedman test on 5 models.

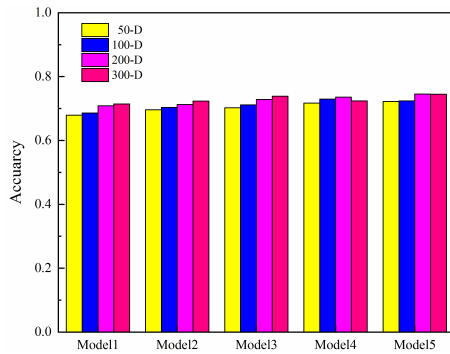


FIGURE 8. Accuracy on the different dimensions of word vector on Restaurant with 10-fold cross-validation. The sentiment polarity includes 3 classes {Positive, Negative, Neural}.

the results of two polarity classification and TABLE 9 shows the results on balanced datasets with 300-D word vector. All the highest accuracy is shown in bold. In order to show the results more intuitively, we draw the data in TABLE 3 into a histogram, and the data in TABLE 6 to TABLE 8 into a line chart. FIGURE 8 shows the performance of different models in different word vector dimensions under three sentiment polarity classification. FIGURE 9 to FIGURE 11 show the best performance of different models in 300-D word vector under different two sentiment polarity classification. FIGURE 12 shows the performance of the unequal and equal classes of different datasets with 300-D word vector.

As is clearly shown in the FIGURE 8, the result is best when the word vector dimension is set to 200 or 300. Besides, when the dimension is given, the Self-Attention-Based BiLSTM model with aspect-term information (Model5) is better than the other four models. Only considering the accuracy rate, when we use the proposed Model5 and the dimension is set to 200 can achieve the best performance, seen in TABLE 3. In order to verify the generalization performance of our model, it was evaluated on two other datasets (Laptop and Twitter). The specific results are shown from TABLE 4 to TABLE 5.

As is shown obviously from FIGURE 9 to FIGURE 11, the accuracy of two classes of sentiment polarity task on different datasets with 300-D word vector is generally

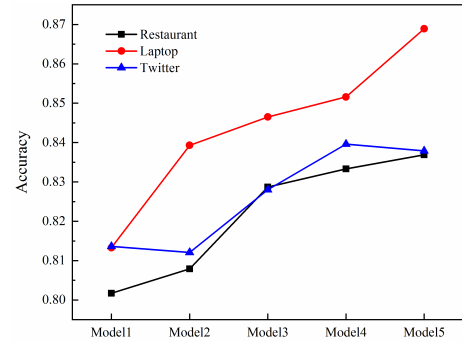


FIGURE 9. Accuracy on the different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Positive, Negative}.

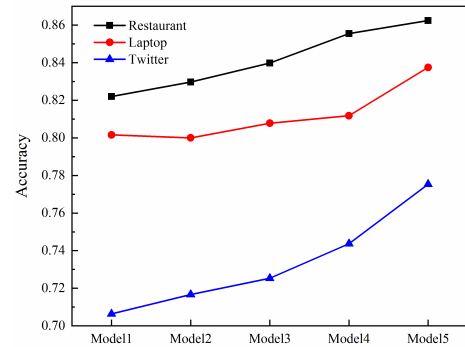


FIGURE 10. Accuracy on the different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Positive, Neural}.

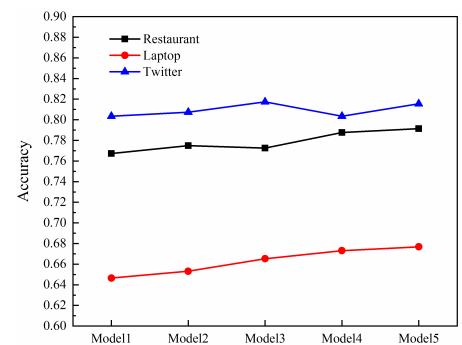


FIGURE 11. Accuracy on the different datasets with 300-D word vector. The sentiment polarity includes 2 classes {Negative, Neural}.

higher than the three classification, but for {Negative, Neural} classification of Laptop, the accuracy has a significant decline. It is enlightening to analyze what kind of misclassification leads to a decline in accuracy. FIGURE 13 and FIGURE 14 shows the normalized confusion matrix on Laptop with 300-D word vector. We can clearly notice the misclassification of Neural and Negative are 0.37 and 0.18 for two sentiment polarity of {Neural, Negative}, and the wrong classification rates are 0.41 and 0.2 respectively.

If the accuracy is considered together with the efficiency, it is more suitable when we choose the Self-Attention-Based

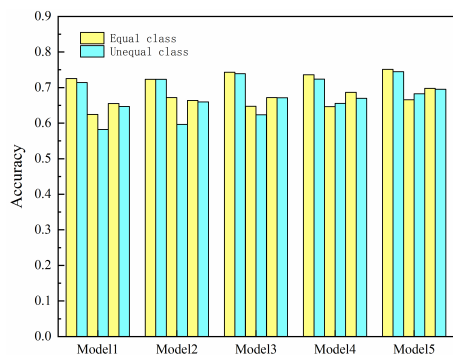


FIGURE 12. Accuracy on the equal and unequal classes with 300-D word vector on different datasets.

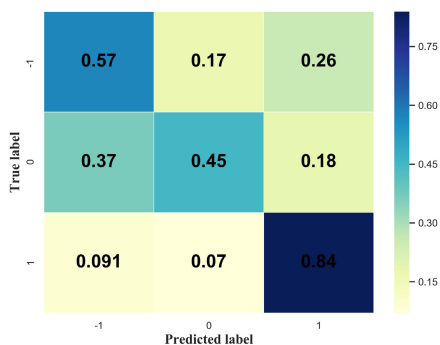


FIGURE 13. Normalized Confusion Matrix on Laptop with 300-D word vector. The sentiment polarity includes 3 classes (Positive, Negative, Neural).

BiLSTM model with aspect-term information and set the word vector dimension to 100.

To further improve the accuracy of the model, the existing corpus were augmented with some data enhancement methods, where the corpus was duplicated to make the classes equal approximately, and finally it was sent to the model for classification. The result is shown that it is better to use the duplicate data mechanism in the FIGURE 12.

The LSTM model has the worst performance than other models, because the standard LSTM cannot effectively capture information about the aspect-terms in the sentence. It can only use the forward information of the target word. That is, it cannot make full use of the semantic information.

The BiLSTM model contains a forward LSTM neural network model and a backward LSTM neural network model, which can capture more information from two directions of the target word, so that BiLSTM can get more semantic information for the target word. Compared with the standard LSTM model, the BiLSTM model improves the accuracy of judging sentiment polarity. However, the BiLSTM model still cannot capture information about aspect-terms in the sentence.

Target-Dependent LSTM model proves that the target information (aspect-term information) is good for sentiment

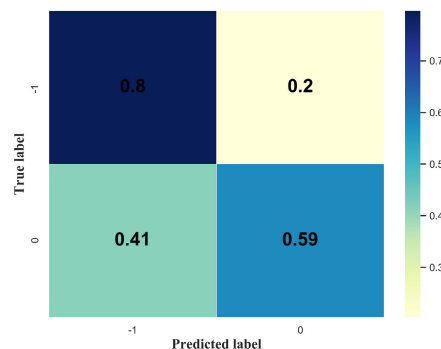


FIGURE 14. Normalized Confusion Matrix on Laptop with 300-D word vector. The sentiment polarity includes 2 classes (Negative, Neural).

polarity classification, improving the accuracy of sentiment polarity classification in a degree. But Target-Dependent LSTM model does not introduce attention mechanism, so the aspect-term corresponding to the sentiment polarity will be affected by irrelevant words in the sentence.

Comparing with models mentioned above, the Self-Attention-Based BiLSTM model attends the self-attention mechanism on the basic BiLSTM model. According to the context, the rich semantic information about the target can be obtained. Afterwards, the key information is captured by the attention layer, which can give different weights to each word based on their importance in the sentence. So, identifying the sentiment polarity corresponding to the aspect-term, the Self-Attention-Based BiLSTM model with aspect-term information has a significant improvement in accuracy.

To construct the Self-Attention-Based BiLSTM model with aspect-term information, the given aspect-term information is fused with the hidden vector. The fusion vector is used as input to attention layer. The Self-Attention-Based BiLSTM model with aspect-term information not only can reduce the impact of other aspect-terms on the weight assignment of the current aspect-term, but also can capture the most important information about the aspect-term corresponding to the opinion terms. In the sentence “I think they have the cleanest living environment in the surrounding city.”, it only includes an aspect-term “living environment”. After the attention layer is added, some words that are more closely related to the aspect-terms are assigned higher weight. Besides, there are more than one aspect-term included in the sentence, like to the sentence “the pizza we eat was rare and distasteful and the concentrated juice was way pretty sugary.”, which contains “pizza” and “concentrated juice”. When “pizza” is considered, the weight distribution of each word in the sentence is shown in FIGURE 15(b). The words “pizza”, “rare”, “distasteful” have achieved higher weight. However, if “concentrated juice” is considered, the weight distribution of each word in the sentence is illustrated in FIGURE 15(c), and the words “concentrated”, “juice”, “pretty”, “sugary” have gotten higher weight. So, the Self-Attention-Based BiLSTM model with aspect-term information has the best performance.

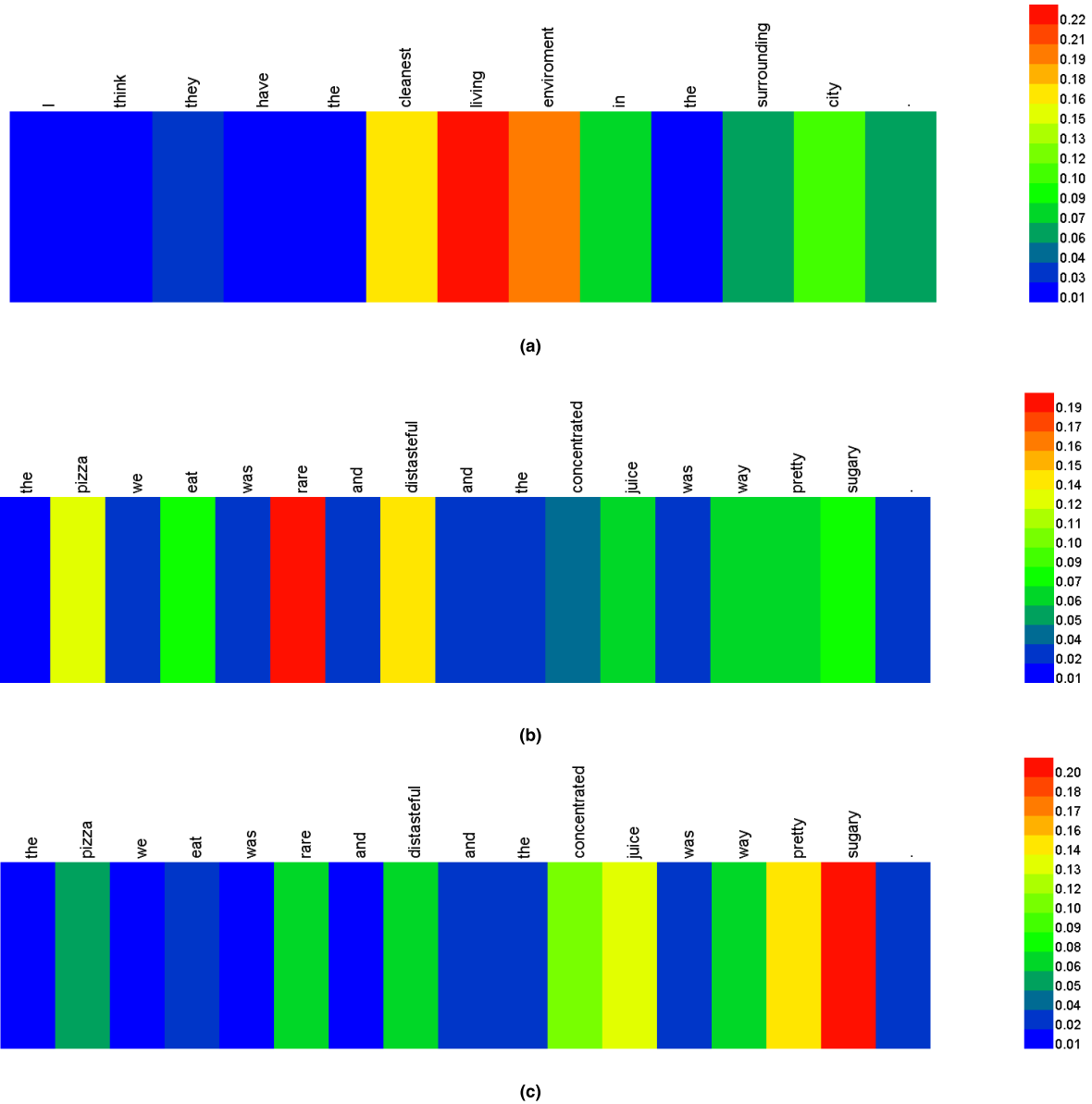


FIGURE 15. Self-Attention Visualizations. The aspect-term of (a), (b) and (c) are “living environment”, “pizza”, “concentrated juice”, respectively.

VI. CONCLUSION AND FUTURE WORK

In this article, a Self-Attention-Based BiLSTM model with aspect-term information is designed. The main work as follows: the bidirectional LSTM model is employed to make full use of the context information, so that more semantic information can be captured. And then by incorporating self-attention mechanism it can capture the important information about aspect-term in the sentence. Finally, after the hidden vector and the aspect-term vector are fused by adding, the composite vector is used as the input to the attention layer to calculate the attention weight, which reduce the computational complexity caused by the vector splicing directly.

Comparing with LSTM and the BiLSTM model, the Self-Attention-Based BiLSTM model is shown to improve the

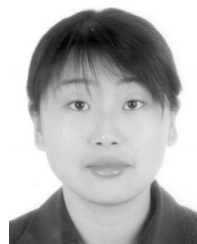
accuracy for sentiment polarity classification. To reduce the effect of among different aspect-terms in the sentence, and to model the correlations between aspect-term and context words, the Self-Attention-Based BiLSTM model with aspect-term information is proposed. Through experiments, the proposed Model5 is illustrated to have a better performance for sentiment polarity classification. In the future, how to get more accurately semantic information should be considered. Besides, finding a more suitable way to merge the hidden vectors with the given aspect-term information can be further improve the accuracy of sentiment polarity classification.

Through the current methods have performed very well for aspect-level sentiment analysis, they do not perform well on some categories where the boundaries are not obvious, such

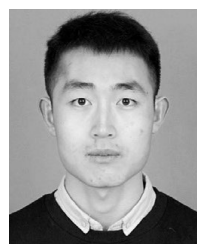
as the Negative and Neural polarity mentioned in our work. As future work, the possible directions of learning better features would be modeled with the attention mechanism and be researched on the Matthew effect in the aspect-terms.

## REFERENCES

- [1] B. Liu, "Sentiment analysis: A multi-faceted problem," *IEEE Intell. Syst.*, vol. 25, no. 3, pp. 76–80, May 2010.
- [2] T. Kuhn, M. Perc, and D. Helbing, "Inheritance patterns in citation networks reveal scientific memes," *Phys. Rev. X*, vol. 4, no. 4, Nov. 2014, Art. no. 041036.
- [3] M. Perc, "Self-organization of progress across the century of physics," *Sci. Rep.*, vol. 3, p. 1720, Apr. 2013.
- [4] J. Zhang and C. Zong, "Deep neural networks in machine translation: An overview," *IEEE Intell. Syst.*, vol. 30, no. 5, pp. 16–25, Sep./Oct. 2015.
- [5] W. Yin, H. Schütze, B. Xiang, and B. Zhou, "ABCNN: Attention-based convolutional neural network for modeling sentence pairs," *Trans. Assoc. Comput. Linguistics*, vol. 4, pp. 259–272, Dec. 2016.
- [6] A. Ansari, M. Maknoja, and A. Shaikh, "Intelligent question answering system based on artificial neural network," in *Proc. IEEE ICETECH*, Coimbatore, India, Mar. 2016, pp. 758–763.
- [7] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," 2015, *arXiv:1509.00685*. [Online]. Available: <https://arxiv.org/abs/1509.00685>
- [8] J. J. Cheng, P. Li, Z. Y. Ding, S. Zhang, and H. Wang, "Sentiment classification of Chinese microblogging texts with global RNN," in *Proc. IEEE DSC*, Changsha, China, Jun. 2016, pp. 653–657.
- [9] Q. Huang, R. Chen, X. Zheng, and Z. Dong, "Deep sentiment representation based on CNN and LSTM," in *Proc. IEEE ICGI*, Fuzhou, China, Aug. 2017, pp. 30–33.
- [10] D. Tang, B. Qin, X. Feng, and T. Liu, "Target-dependent sentiment classification with long short term memory," 2015, *arXiv:1512.01100*. [Online]. Available: <http://arxiv.org/abs/1512.01100v1>
- [11] C. Xia, X. Li, Z. Wang, and M. Perc, "Doubly effects of information sharing on interdependent network reciprocity," *New J. Phys.*, vol. 20, no. 7, Jul. 2018, Art. no. 075005.
- [12] V. Mnih, N. Heess, K. Kavukcuoglu, and A. Graves, "Recurrent models of visual attention," 2014, *arXiv:1406.6247*. [Online]. Available: <https://arxiv.org/abs/1406.6247>
- [13] K. Xu, J. L. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, Attend and Tell: Neural image caption generation with visual attention," in *Proc. Int. Conf. Mach. Learn.*, vol. 3, Jun. 2015, pp. 2048–2057.
- [14] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: <https://arxiv.org/abs/1409.0473>
- [15] R. Nallapati, B. Zhou, C. N. dos Santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence RNNs and beyond," 2016, *arXiv:1602.06023*. [Online]. Available: <https://arxiv.org/abs/1602.06023>
- [16] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," 2016, *arXiv:1605.08900*. [Online]. Available: <https://arxiv.org/abs/1605.08900>
- [17] Y. Wang, M. Huang, L. Zhao, and X. Zhu, "Attention-based LSTM for aspect-level sentiment classification," in *Proc. EMNLP*, Austin, TX, USA, Nov. 2016, pp. 606–615.
- [18] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and L. Polosukhin, "Attention Is all you need," in *Proc. NIPS*, Long Beach, CA, USA, Dec. 2017, pp. 6000–6010.
- [19] M. Perc, "The Matthew effect in empirical data," *J. Roy. Soc. Interface*, vol. 11, no. 98, Sep. 2014, Art. no. 20140378.
- [20] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," in *Proc. IJCAI*, Melbourne, VIC, Australia, Aug. 2017, pp. 4068–4074.
- [21] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*. [Online]. Available: <https://arxiv.org/abs/1408.5882>
- [22] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," in *Proc. ACL*, Baltimore, MD, USA, Jun. 2014, pp. 655–665.
- [23] Y. Ren, Y. Zhang, M. Zhang, and D. Ji, "Context-sensitive Twitter sentiment classification using neural network," in *Proc. AAAI*, Phoenix, AZ, USA, Feb. 2016, pp. 215–221.
- [24] S. Rosenthal, N. Farra, and P. Nakov, "SemEval-2017 task 4: Sentiment analysis in Twitter," in *Proc. SemEval*, Vancouver, BC, Canada, Aug. 2017, pp. 502–518.
- [25] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. NIPS*, Lake Tahoe, NV, USA, Dec. 2013, pp. 3111–3119.
- [26] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proc. EMNLP*, Doha, Qatar, Oct. 2014, pp. 1532–1543.
- [27] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent convolutional neural networks for text classification," in *Proc. AAAI*, Austin, TX, USA, Jan. 2015, pp. 2267–2273.
- [28] L. Arras, G. Montavon, K.-R. Müller, and W. Samek, "Explaining recurrent neural network predictions in sentiment analysis," in *Proc. ACL*, Copenhagen, Denmark, Sep. 2017, pp. 159–168.
- [29] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Proc. ISCA*, Portland, OR, USA, Sep. 2012, pp. 194–197.
- [30] O. Abdelwahab, M. Bahgat, C. J. Lowrance, and A. Elmaghraby, "Effect of training set size on SVM and Naive Bayes for Twitter sentiment analysis," in *Proc. IEEE ISSPIT*, Abu Dhabi, United Arab Emirates, Dec. 2015, pp. 46–51.
- [31] T. Mikolov, M. Karafiát, L. Burget, J. Cernocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proc. ISCA*, Chiba, Japan, Sep. 2010, pp. 1045–1048.
- [32] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent Twitter sentiment classification," in *Proc. ACL*, Baltimore, Maryland, vol. 2, Jun. 2014, pp. 49–54.
- [33] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
- [34] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," 2015, *arXiv:1503.00075*. [Online]. Available: <https://arxiv.org/abs/1503.00075>
- [35] J. Li, M.-T. Luong, and D. Jurafsky, "A hierarchical neural autoencoder for paragraphs and documents," 2015, *arXiv:1506.01057*. [Online]. Available: <https://arxiv.org/abs/1506.01057>
- [36] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher, "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. ICML*, New York, NY, USA, vol. 48, Jun. 2016, pp. 1378–1387.



interests include granular computing, text mining, and intelligent information processing.



**JUN XIE** was born in Taiyuan, China, in 1979. She received the M.B.S. degree in intelligent autonomous system from Aalborg University, Denmark, in 2005, and the Ph.D. degree in circuit and system from the Taiyuan University of Technology, in 2009. She has been an Associate Professor with the College of Information Engineering (now named as College of Information and Computer), Taiyuan University of Technology. She has authored over 30 articles. Her research

**BO CHEN** is currently pursuing the master's degree with the College of Information and Computer, Taiyuan University of Technology. His main research interests include sentiment analysis and rough sets.



**XINGLONG GU** is currently pursuing the master's degree with the College of Information and Computer, Taiyuan University of Technology. His main research interests include text mining and sentiment analysis.



**XINYING XU** was born in Dingxiang, China, in 1979. He received the Ph.D. degree in circuit and system from the Taiyuan University of Technology, Taiyuan, China, in 2009. Since 2011, he has been an associate professor with the College of Information Engineering, Taiyuan University of Technology. He is currently with the College of Electrical and Power Engineering. He has authored over 40 articles. His research interests include computer vision, intelligent control, and intelligent information processing.

• • •



**FENGMEI LIANG** was born in Taiyuan, China, in 1969. She received the Ph.D. degree in circuit and system from the Taiyuan University of Technology, in 2009. She has been an Associate Professor with the College of Information Engineering (now named as College of Information and Computer), Taiyuan University of Technology. She has authored over 50 articles. Her research interests include computer vision and intelligent information processing.