Journal of Computer Technology and Software
ISSN:2998-2383
Vol. 1, No. 1, 2022

Enhancing Sentiment Analysis Using Attention Mechanism

Daniel Parker

Department of Computer Science, University of Texas at Dallas, USA

Abstract: The rise of social platforms in China has increased user-generated content (UGC), which often contains rich emotional information. Analyzing UGC sentiment helps identify group emotions on specific events, aiding decision-making for platforms and authorities. This paper introduces a novel sentiment analysis method that combines a bipolar word attention mechanism with a DSA-CNN model. By developing bipolar word vectors and leveraging an emotional dictionary, the model extracts features from positive and negative emotions, enhancing short text emotion detection. Experiments on three datasets demonstrate superior performance and faster convergence due to the attention mechanism. Future work aims to improve the sentiment dictionary and word vector training to address current limitations and better capture contextual influences on sentence emotion.

Keywords: Convolutional Neural Network; Bidirectional affective word vectors; Attention mechanism; Social media; Sentiment analysis.

1. Introduction

With the popularity of the Internet and mobile devices, there are more and more social platforms in China, creating conditions for user-generated content [1,2]. People have also been used to publish and share their views on the Internet, which contains a large number of user emotions and has attracted extensive attention from academia and the industry. By analyzing the emotional tendency of user-generated Content (UGC) on social media platforms, we can identify the group emotions of the majority of users for specific events, which is conducive to scientific decision-making of platforms and regulatory authorities [3-6].

UGC sentiment analysis, also known as opinion mining, mainly classifies the emotional polarity of UGC content and distinguishes whether a sentence is positive, negative, or neutral. Or a fine-grained emotional analysis can be carried out to divide the emotions into happiness, sadness, anger, fear, disgust, surprise, and so on [7] to determine which emotions the sentence belongs to. Text UGC sentiment analysis is one of the important contents of UGC sentiment analysis. There are three main methods for text UGC sentiment analysis: (1) research method based on sentiment lexicon. (2) Research methods based on traditional machine learning. It focuses on mining the features of sentences and then training the classifier to classify them. (3) Methods based on deep learning. Deep learning model-based text sentiment analysis methods generally use word vectors as input to classify UGC sentiment.

In recent years, given deep learning has achieved good results in natural language processing, computer vision, and other fields [8,9]. Many scholars use deep learning models to analyze UGC emotions. For example, Asghar et al.[10] establish a senti-eSystem by LSTM and Fuzzy logic to recognize customer emotion and satisfaction. Generally, on social media platforms, the length of UGC published by users is relatively short. For example, the length of a microblog is generally no more than 140 characters. There is no need for long-distance sequence modeling in this environment, so CNN has become one of the preferred models for the emotional analysis of social media [11].

Kim[12] used max-pooling and average-pooling for down-sampling. But the max-pooling method ignores the influence of other factors in the sentence. The average pooling method treats all the information as the same influence, while the decision emotion is often part of the sentence's emotional words. Yang[13] introduced the attention mechanism into the recurrent neural network to classify documents. The attention mechanism can automatically pay attention to some important information, assign different weights to features, and calculate the final features according to the weights. However, most of the query vectors of attention mechanism in text classification are randomly generated and trained in the network, whose results are affected by the initialization effect[13,14]. The existing sentiment dictionary contains a large amount of prior sentiment information, and we think that it can be combined with neural networks to improve the attention mechanism in sentiment classification.

To sum up, the attention mechanism is used to improve the pooling layer of the CNN model. A Convolutional Neural

Network Based Double Sentiment Word Attention Pooling (DSA-CNN) model is proposed. All the emotion words in the unsupervised emotion dataset were tagged with the sentiment lexicon, and the bidirectional affective word vectors of positive and negative emotions were obtained by Word2Vec. Since the same word appears in different places in different sentences, its influence may be different, and the long-distance location information cannot be reflected in the process of convolution and pooling. Therefore, the position embedding vector is combined with the word vector as the input of CNN. The positive emotion vector and negative emotion vector obtained by training were used as attention pooling query vectors, respectively, and the final emotion representation was obtained after pooling and merging. Each important information extracted from the convolutional kernel can be paid attention to through the attention pooling of the bidirectional affective word vector, and more hidden information can be mined from both positive and negative emotions so that the model can extract more accurate emotional features. This paper has the following three contributions:

(1) DSA-CNN model combining bidirectional sentiment word vector and attentional pooling mechanism is proposed to learn more effective features in short texts from both positive and negative emotions.

(2) The pre-trained method of affective word vectors is proposed, including the centroid method and the population method. Besides, two different corpora are established for training, and the effectiveness of the two methods is verified.

(3) The results of the experiment in multiple data sets show that the DSA-CNN model has achieved better performance compared with multiple classical classification models, and the speed of the model converges significantly.

2. Related works

2.1. Sentiment lexicon method

The key point of the dictionary-based emotion classification method is the construction of the dictionary. The construction of the sentiment lexicon can be divided into manual construction by experts and expansion by using the existing sentiment lexicon. The common General emotion dictionaries constructed by artificial methods mainly include Harvard University's General Inquirer Lexicon Sentiment lexicon (GI), SentiWordNet sentiment lexicon based on WordNet, Hownet Sentiment lexicon based on CNKI, and NTUSD Sentiment lexicon based on Taiwan University. The dictionary constructed manually by domain experts has high accuracy, but the efficiency of dictionary construction and application is low. The dictionary constructed manually is usually combined with other automatic methods. Or construct a domain sentiment lexicon for a particular domain. For example, Xu et al. [15] manually collected domain emotion words and polysemy emotion words in the hotel, digital, fruit, clothing, shampoo, and other fields, and constructed a sentiment lexicon containing a broader range of emotion words, thus improving the application effect of emotion analysis in the above fields. Due to the social network platform of UGC content updates very quickly, using dictionary corpus extension general feelings to improve the UGC analysis method has been the attention of many scholars, its basic idea is based on the existing emotional dictionary and the seed word list, according to the statistics or semantic information to calculate the new tendency of emotion Using the initial seed dictionary, Kaity et al. [16] built a semi-automatic framework to identify new emotional words in a massive corpus. Wu et al.[17]proposed a method to build a Weibo Chinese sentiment dictionary by labeling the 100 most commonly used emoticons and obtaining an emotion score by calculating the PMI of candidate words and emoji sets。

Although the classification method of sentiment lexicon is relatively simple and convenient, there are also many shortcomings in the above methods, such as over-reliance on the quality of sentiment lexicon, and due to the polysemy of emotion words, the applicability of sentiment lexicon method in different fields is poor.

2.2. Deep learning method

Compared with traditional machine learning methods, the deep learning method relieves the dependence of the model on feature engineering and has the feature of "end to end", so it is favored by the majority of scholars. Kim[12] improved the convolutional neural network for sentence classification and used the convolutional kernel to extract local features of the text. Despite its simple structure, it still performed well in multiple data sets. Since then, CNN has gradually been widely used for text classification tasks. Zhang et al.[18] proposed a character-level convolutional neural network, which changed the input mode at the word level and divided sentences into single English letters, numbers, and other characters as input, alleviating the problems caused by segmentation errors and spelling errors. Yan et al.[19] added two parallel DenseNet Networks to the CNN model to pay attention to the global features of the text, and combined the extracted different features to realize the sentiment analysis of the short text. Gan[20] proposed a Separable Extended Convolutional Neural Network (SA-SDCCN) based on sparse attention, which is mainly composed of a separable convolutional module, sparse attention layer, and output layer. Due to the separable structure and attention layer oriented to specific entity emotion, the model has good performance in terms of computation cost and performance. Zhou[21] used a two-word embedded convolutional neural network for text emotion classification and combined it with GloVe and Word2vec models to embed text, which achieved a better semantic expression effect.

In addition, Wang[22] used the Long Short-Term Memory network (LSTM) for sentiment prediction on Twitter. Lai et al.[23] proposed an RCNN model, which connected the word itself with the context vector on both sides to obtain the final expression of the word, in which the context vector of the word was trained by RNN. In this way, word ambiguity in sentences could be better eliminated. In addition to commonly used models, Zuo[24] applied Graph Neural Network (GNN) to emotion analysis and constructed a context-specific heterogeneous Graph Convolutional Network, which effectively identified implicit emotions in sentences combined with context.

The deep learning method has become commonly used in the sentiment analysis research method, and has achieved many remarkable achievements Improvements to the deep learning method work generally there are two research directions: one is improving the existing model structure, such as CNN with RNN model combining attention mechanism, or new models such as the Transformer is put forward [25], BERT[9] et al, can be applied to various natural language processing tasks; The other is to combine other information or existing resources to construct new emotional features to enhance online learning ability. For example, Yin et al.[26] proposed SCNN for emotional words, which learned the emotional embedding of each word based on SentiWordNet, and then input the word embedding and emotion embedding into CNN together. Luo[27] uses the LDA model to obtain the topic distribution of short texts and constructs the feature vector representation of short texts based on the topic distribution to improve the accuracy of sentiment classification of short texts. Alharbi et al. [28] added user behavior characteristics into the neural network model for emotion analysis. This paper uses two research directions for reference, improves the structure of the Text-CNN model, and uses the sentiment lexicon to extract the emotion words to train the affective word vectors and enhance the emotional learning ability of the model.

2.3. Attention mechanism

The attention mechanism was first proposed in the field of visual images, and Bahdanau[29] applied the attention mechanism to natural language processing. He took the hidden state at the previous moment as the query vector and changed the weight of each word so that the semantic vector C was also changed. Compared with the previous machine translation model, the semantic vector C was always the same, which could match the relationship between input and output more accurately. Since then, attention has gradually been applied to various NLP tasks, such as intelligent question answering and text categorization. Yang[13] proposed a hierarchical attention network for document classification. The model uses two-layer bi-directional GRUS to apply the attention mechanism at the word level and sentence level respectively to obtain feature representation at the sentence level and document level. Similarly, Zhou[14] proposed a bidirectional LSTM model combined with an attention mechanism to be used in text classification, combining the output of the LSTM hidden layer at all times with attention. Cheng et al. [30] combined the advantages of CNN and BiGRU and focused on the words in sentences that have an important influence on sentiment classification through the attention mechanism, thus improving the feature extraction ability of the model.

Existing studies have demonstrated the effectiveness of the combination of attention mechanism and neural network. Attention has become an important part of neural network structure in NLP tasks, which improves the efficiency and interpretability of neural networks. But it still has some defects in the text classification. A summary of all the above methods and their gaps are shown in Table 1.

Table 1. Summary of sentiment analysis methods

Sentiment analysis method	Article	Proposed or used method	Comments
Sentiment lexicon method	Xu et al. [15]	Constructed sentiment lexicon by expert	Simple and convenient, but poor in different fields and flexibility.
	Kaity et al. [16]	Constructed sentiment lexicon by a semi-automatic framework	
	Wu et al. [17]	Constructed sentiment lexicon by calculating PMI	
Deep learning method	Kim et al.[12]	Text-CNN	Fast train but pooling layer discards some information.
	Zhang et al.[18]	Char-CNN	
	Zhou et al.[21]	GloVe and Word2vec + CNN	
	Wang[22]	LSTM	Consider text dependency but slow training speed.
	Lai et al.[23]	RCNN	
	Zuo et al.[24]	GNN	Ignore word order information.
Attention mechanism	Yang et al. [13]	Hierarchical-BiLSTM + Attention	Initialization affects the effect.
	Cheng et al. [30]	CNN + BiGRU + Attention	

In this paper, the attention mechanism is applied to the improvement of the CNN pooling layer structure, which can extract text features more effectively. At the same time, a pre-trained method of affective word vectors is proposed. The pre-trained affective vectors have a high similarity with the emotion feature in the vector space. The affective vectors are used as query vectors in attention pooling, which can mine more information from the two emotion directions in the short text, reduce the influence of random initialization, and have stronger robustness.

3. Model design idea

The convolutional neural network model DSA-CNN proposed in this paper, which combines the attention mechanism of bidirectional emotional words, has four layers, namely, the embedding layer, the convolutional layer, the attention pooling layer, and the output layer, as shown in Figure 1. The embedding layer transforms a sentence into a word vector matrix and position vector matrix. The convolution layer extracts local features through convolution kernels of different sizes and generates feature maps. The attention pooling layer is based on the bidirectional affective word vectors obtained by training, and the important emotion features in the feature map are obtained. The output layer uses Softmax functions to obtain classification results.

3.1. Embedding layer

Since the text is a kind of unstructured data information, it cannot be directly calculated, and words can be transformed into word vectors through word embedding. For a text sequence T of length N = [w1, w2, …, wn]. Get each word by word embedding wi vector vi, where vi∈Rk, k is the word vector dimension. The word embedding operation is carried out by using a trained dictionary W∈R N×k, in which each word has a unique vector representation, which can be regarded as a process of looking up the dictionary.

The same word in different positions may contain different emotional intensities, such as some emphatic sentences. In addition, since the length of text content in social media is generally short, adding location information can also enrich the features of text content[31]. In this paper, position features are mapped into continuous vector representations. For the ith position in the sentence, the position of L-dimension is randomly generated and embedded into pi. Combine each word vector and position vector as a vector representation of words such as Eq.3.1:

$$x_i = v_i \odot p_i \quad 3.1$$

$x_i \in$ Rk+l, ⊙indicates column direction stitching. The text sequence T after the embedding layer is transformed into the two−dimensional matrix form such as Eq. 3.2:

$$S = x_1 \oplus x_2 \oplus \ldots \oplus x_n \quad 3.2$$

⊕represents the stitching according to the direction of row vectors, S∈R N×(k+l) is input to the next level.

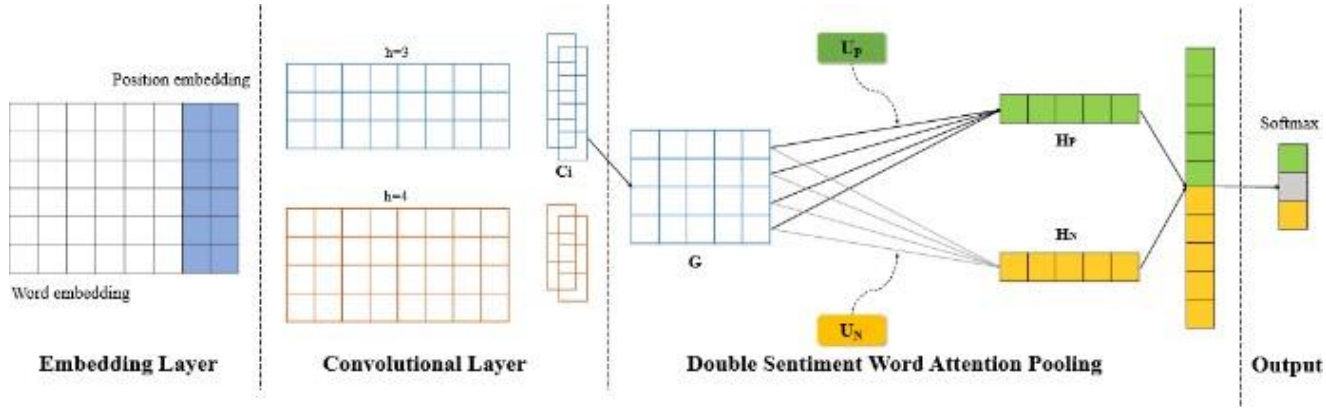

Figure 1. DSA-CNN model structure diagram

3.2. Convolution layer

In text classification, each row vector of the two-dimensional matrix *S* represents the vector representation of a word, and the semantic integrity of the word should be maintained during the convolution operation. Therefore, the position where the convolution window slides should be a complete word, that is, the width of the convolution kernel should be consistent with the row vector dimension of the embedded matrix *S*. In the convolution layer, convolution kernels of different lengths can be used to carry out convolution and different local features can be extracted. Therefore, for a convolution kernel m_i with length H, its overall shape should be H×(k+l). The convolution operation from x_i to x_{i+h-1} in the matrix *S* is carried out to obtain the local feature c_i, and the convolution process is shown as Eq.3.3:

$$c_i = f(m_i \cdot x_{i:i+h-1} + b) \quad 3.3$$

Where *b* is the bias term, *f ()* is the nonlinear activation function. ReLu function is used here.

In matrix *S*, the convolution kernel m_i slides from top to bottom for convolution operation with the step of 1. A total of (N − H +1) local features are extracted. As shown in Eq3.4, a convolution feature graph C_i can be obtained:

$$C_i = \{c_1, c_2, \cdots, c_{n-h+1}\} \quad 3.4$$

To fully extract features, multiple convolution kernels can be used, and each convolution kernel Mi will generate a convolution feature graph Ci. The same row in different feature graphs can be regarded as features at multiple levels extracted by the convolution kernel at the same position in the embedded matrix S. Therefore, feature graphs can be spliced in columns for subsequent operations. The splicing operationcan be expressed as Eq.3.5:

$$M = C_1 \odot C_2 \cdots \odot C_I \quad 3.5$$

Where, $M \in R^{(N-H+1)\times I}$ represents the complete feature graph after splicing, and *I* represents the number of convolution kernels used.

3.3. Bidirectional affective word vectors

In this paper, we propose the concept of bidirectional affective word vector, that is, the word vector representation of overall affective information considering positive and negative emotions, which is used as the query vector of the attention mechanism. The training of bidirectional affective word vectors can be divided into the following three steps.

(1) Construct a mixed sentiment lexicon. The bidirectional sentiment word vector should contain a large amount of emotional word information, so this paper combines two commonly used dictionaries, Hownet Sentiment lexicon and the Taiwan University Sentiment lexicon (NTUSD). Hownet dictionary contains 4,566 positive Chinese words and 4,370 negative Chinese words; NTUSD contains 2812 positive words and 8278 negative words. After removing repetitive words, the constructed fusion dictionary is shown in Table 2.

Table 2. An example of a fusion sentiment lexicon

	number	instance
Positive words	6506	Anhao, Qinshan
Negative words	11184	Elie, Shangxin

(2) Obtaining corpus and preprocessing. The training of sentimental word vectors is an unsupervised learning process, which requires a large number of text corpus, and the text corpus needs to contain sentimental information. Therefore, a crawler is used to crawl the text containing most of the sentiments on the network, including comments on multiple platforms such as Meituan, Qunar, JingDong, etc. Add the fusion sentiment dictionary to the custom word segmentation dictionary to ensure the accurate division of sentimental words. For each prediction, if the result after the word segmentation contains the positive sentimental word in the fusion sentiment dictionary, it is replaced with <pos> marker; Similarly, all negative sentimental words are replaced with <neg> markers.

(3) Training word vector. Word2vec is used to train the sentiment corpus and convert text vocabulary into a D-dimensional space vector. The vector corresponding to the <pos> mark in the trained word vector file is extracted as the positive emotion vector, denoted as Up; The vector corresponding to the <neg> marker is taken as the negative emotion vector Un. After the training of the sentiment anticipation library, Up and Un cover most of the sentiment information of sentimental words, that is, in the vector space, they have a high degree of similarity with positive and negative emotional features respectively.

3.4. Attention pooling layer of bidirectional sentimental words

Attention can be explained intuitively using human visual mechanisms. For example, our visual system tends to focus on information that is important in images and tends to ignore irrelevant information. In neural networks, the focus of the attention mechanism is to calculate the weight of each feature in a set of features, that is, according to some rules or some additional information, also known as the query vector, to judge which features are relatively important and give them a larger weight. Therefore, compared with average pooling, attention-based pooling can effectively distinguish the characteristics of different influences. For maximum pooling, attention pooling contains more valid information in the text. Firstly, the feature graph M is passed through a full connection layer in Eq. 3.6, which has two main purposes. It is to obtain deeper semantic features; on the other hand, the shape of M can be changed to make the dimension of its column direction the same as that of Up and Un, so that attention calculation can be carried out.

$$G = f(W \cdot M + B) \quad 3.6$$

G represents the generated deep semantic feature matrix, $W \in R^{l \times d}$ is the weight matrix, d is the dimension of U_p and U_n. Secondly, U_p is acted as a query vector of attention mechanism, using the dot product type calculation method to generate attention weighting α_i^P such as Eq. 3.7 , the weighted sum of every row in the G attention pooling, the formula is as Eq. 3.8:

$$\alpha_i^P = Softmax(g_i U_P^\Gamma) = \frac{\exp(g_i U_P^\Gamma)}{\sum \exp(g_i U_P^\Gamma)} \quad 3.7$$

$$H_P = \sum_{i=1}^{N-h+1} \alpha_i^P g_i \quad 3.8$$

Where g_i is the i-th row of G, U_P^Γ is the transpose of U_p, $H_P \in R^d$ is positive characteristic generated by attention pooling. By the same, the U_n is a query vector, we can get negative pooling features $H_N \in R^d$.

Dot product is similar to the vector cosine similarity. The weight of α_i would be larger if any g_i of the G contained the sentimental information, because U_p and U_n contain the characteristics of the positive and negative sentimental words, and the results of impact is bigger also. Finally, the final sentimental characteristics of H is got based on H_P and H_N, $H \in R^{2d}$. The entire algorithm flow of bidirectional attention pooling is shown as follows.

Algorithm: attention pooling of bidirectional affective words

Input: Convolution characteristic graph $M \in R^{(N-h+1) \times l}$
Output: emotional feature $H \in R^{2d}$
Begin:
1: $G = f(WM + B)$ //Fully connected layer
2: ***Positive emotional pooling:***
3: for i to l: // l is the row number of G
4: calculate α_i^P using U_P // Get the weight of positive attention (eq. 3.7)
5: end for
6: Calculating the positive pooling characteristics H_P // (eq. 3.8)
7: **Negative emotional pooling:**
8: for i to l:
9: calculate α_i^n using U_n //Negative attention weight
10: end for
11: Calculate the negative pooling characteristics H_N
12: $H = H_P \odot H_N$ // *Concat column*
End

The time complexity is defined as the sum of the complexities of each step of the method. Considering the input matrix $S \in R^{N \times (k+l)}$, the time complexity of a single convolutional layer is $O(H \times (k+l) \times I)$, I is the number of convolutional kernels and H is the length of the convolution kernel. The time complexity of the attention pooling layer needs $O(l)$ computations, where l is the row number of the pooling layer input.

3.5. Output layer

The output layer converts the final pooled feature H into a T-dimensional output vector through linear transformation firstly. SoftMax function is used to obtain the probability distribution of the output results in Eq. 3.9. The probability of the text belonging to positive, negative, and neutral was calculated, and the category with the largest probability is taken as the final classification result:

$$y = Softmax(W_H H + B_H) \quad 3.9$$

Where $y \in R^t$ represents the probability of belonging to each category. In addition, a Dropout layer is added to the output layer to reduce the impact of overfitting.

4. Experiment

4.1. Data set

Two social media Data sets, Weibo nCoV Data and NLPCCC 2014, and an e-commerce Data set YF_Dianping, were used as the experimental Data. Weibo NCOV Data dataset is blog posts related to the theme of "COVID-19" from January 1, 2020, to February 20, 2020, where 99,882 posts are marked, which could be divided into three categories: positive, negative, and neutral. Because there are too many neutral texts, to maintain the balance of the categories, part of the neutral text is deleted, so that the number of the three categories is closer. The number of processed texts is 48,760, and the training set and verification set are divided according to the ratio of 9:1.

NLPCCC2014 is a Chinese microblog sentiment assessment task data set released in 2014, in which the test set contains 13,998 posts containing content and the verification set contains 6,000 posts. There are eight sentimental labels, including None, Happiness, Like, Sadness, Disgust, Anger, Fear, and Surprise. In the paper, the label None is acted as neutral sentiment, the label Happiness and Like are labeled as positive sentiment, and the rest are divided into negative sentiment, which is used as the data set of the trip classification task. YF_Dianping is the data set of comments from the merchant of Dianping, and the overall rating of users is between 1 and 5. In the paper, comments of 1-2 points are classified as bad comments, comments of 3 points are classified as medium comments, and comments of 4-5 points are classified as favorable comments. A total of 100,000 comments data are selected and divided into a training set and verification set according to the ratio of 9:1. The specific number of each category of the three data sets is shown in Table 3, 4 and 5.

Table 3. Weibo nCoV Data

	Positive	Neutral	Negative	Total
Training set	16189	16653	11484	44326
Validation set	1620	1666	1148	4434
Total	17809	18319	12632	48760
Average length of Sentence	46			

Table 4. NLPCC2014

	Positive	Neutral	Negative	Total
Training set	3663	6591	3744	13998
Validation set	1483	3603	914	6000
Total	5146	10194	4658	19998
Average length of sentence	55			

Table 5. Yf_dianping

	Positive	Neutral	Negative	Total
Training set	36000	27000	27000	90000
Validation set	4000	3000	3000	10000
Total	40000	30000	30000	100000
Average length of sentence	90			

4.2. Word vector training and parameters

User's habits of sentimental expression are different on different platforms, multiple types of comments were crawled from MeiTuan (catering food, entertainment, etc.), Qunar (hotel), JingDong (including digital, clothing, daily necessities, etc.) except for social media data set to obtain more accurate universal sentiment vector expressions, then repeated comments and some highly similar comments, such as "favorable comments" and "favorable comments", were deleted. Finally, a total of 183,710 texts formed a sentimental corpus, which was used to train sentimental word vectors. Sentimental words in the corpus are replaced with <pos> and <neg> tags by using the method described in section 3.3. In general, the corpus contains 485,161 positive words and 303,415 negative words.

We use Word2vec to train bidirectional sentimental word vectors. Since the sentimental corpus in the paper was small, the CBOW model in Word2vec predicted the probability of target words through context, which was more suitable for a small data set, so the CBOW model was adopted. In addition, the minimum word frequency was set to 5, the context window was set to 5, and the bidirectional sentimental word vector dimension was set to 100. Word vector files were obtained by training, and vectors corresponding to **<*pos*>** and **<*neg*>** markers were extracted, namely, U_P and U_N. Unknown words were randomly initialized with a normal distribution.

4.3. Pretreatment and experimental parameters

The training device information is I5-9400 @2.9GHZ, 16GB memory with a GPU of 1650. The package version used to build and train the network is Tensorflow-1.10-GPU.

Datasets are preprocessed before training, including simple and complicated conversion, removal of special symbols and unrecognized characters, normalization of punctuation marks and removal of URL links in Weibo, etc., and word segmentation was carried out using the Jieba library. The maximum length of each sentence was set to 100. If the sentence was not enough, zero filling was used to complete the sentence, and the part beyond it was truncated. For position embedding, a normal distribution with a standard deviation of 0.1 was used for random initialization, and the characteristic dimension is 20. We determined the optimal parameters of a network using grid search, as shown in Table 6.

Table 6. Parameters of the network model

Parameter	Value
Convolution kernel size	3, 4, and 5
Number of convolution kernels	128, 128, and 128
Learning rate	0.0005
Dropout	0.5
Batch size	64

4.4. Contrast models

To verify the effectiveness of the model, the following six models were used to compare.

(1) Word2Vec + SVM. The word vector generated by the Word2Vec model was used as the text feature, the word vector was input into SVM for classification, and the kernel function was selected as RBF kernel.

(2) CNN-Max. In the CNN model proposed in Literature [20], Word2vec word vector was used as input, a multi-size convolution kernel was used to extract features, and maximum pooling was adopted in the pooling layer.

(3) AT-CNN. In the pooling layer, the attention pooling method was adopted, in which the attention query vector was randomly generated in the network and continuously trained in the network. The rest was the same as that in the literature [20].

(4) BiLSTM. In the bidirectional LSTM model, the output of the hidden layer in the last step of the LSTM time dimension was classified as the final extracted feature.

(5) AT-BiLSTM. The bidirectional LSTM model proposed in Literature [28] combined the attention mechanism, which combined the output of all-time dimensions with the attention mechanism, and the attention query vector was also randomly generated and trained in the network.

(6) RoBERTa-C. Cui advance pre-training "RoBERTa - WWM - ext, Chinese" model [32], by the Chinese Wikipedia, other Wikipedia, news, and the total number of words such as corpus, training a total of 5.4 B. Only its output layer parameters are changed in this article.

(7) DSA-CNN. The convolutional neural network model proposed in the paper combined the bidirectional sentimental word attention mechanism, which combined the position embedding vector with the word vector as the input of CNN. The pre-trained positive and negative affective word vectors were used as the query vectors of attention pooling, and the final affective features were obtained for classification after attention pooling and merging.

4.5. Experiment

4.5.1. Model effect analysis

The seven models were evaluated on the above three data sets for the tri-classification task, and all the experimental results were averaged by the five experiments. The overall results were shown in Fig. 2. The Accuracy rates and Macro F1 indexes of the three data sets were shown in Table 7, Table 8, and Table 9 respectively.

Table 7. Experimental results of Weibo NCOV Data

Model	**Accuracy**	**Macro F1**
SVM	0.4461	0.3644
CNN-Max	0.6284	0.6276
AT-CNN	0.6552	0.6567
BiLSTM	0.6712	0.5990
AT-BiLSTM	0.6754	0.6012
RoBERTa-C	0.6840	0.6846
DSA-CNN	**0.6880**	**0.6882**

Table 8. Experimental results of NLPCC

Model	Accuracy	Macro F1
SVM	0.5903	0.2709
CNN-Max	0.6382	0.5577
AT-CNN	0.6679	0.6105
BiLSTM	0.6714	0.5781
AT-BiLSTM	0.6812	0.6212
RoBERTa-C	0.6974	0.6551
DSA-CNN	**0.7192**	**0.6645**

Table 9. Experimental results of NLPCC yf_dianping

Model	Accuracy	Macro F1
SVM	0.6330	0.6218
CNN-Max	0.6805	0.5988
AT-CNN	0.7102	0.6315
BiLSTM	0.7155	0.6313
AT-BiLSTM	0.7247	0.6419
RoBERTa-C	0.6850	**0.6751**
DSA-CNN	**0.7344**	0.6550

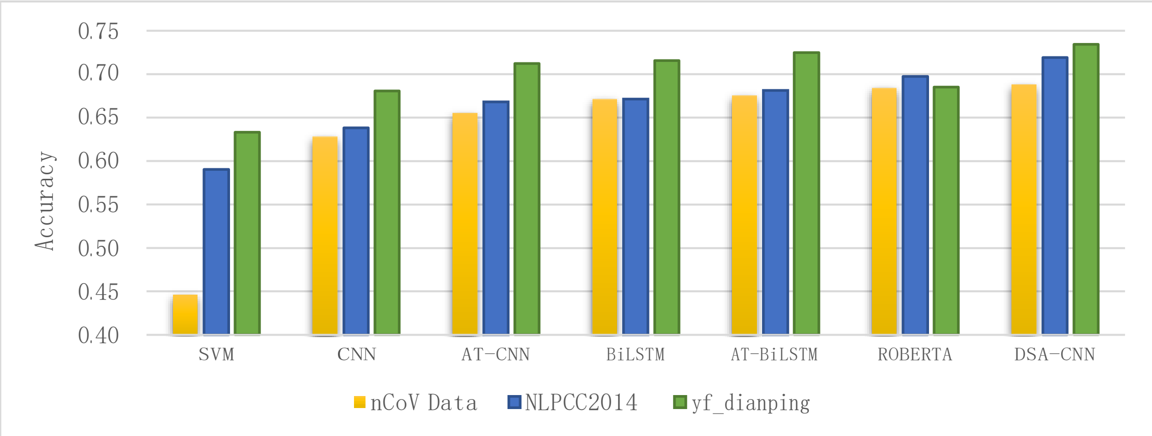

Fig. 2 Comparison of accuracy indexes

As can be seen from Figure 2, several neural network models have greatly improved compared with Word2Vec+SVM, and the DSA-CNN model proposed in this paper has the best performance in several data sets. In the Data set of Weibo nCoV Data, the Accuracy and Macro F1 of attention-pooling at-CNN used by DSA-CNN in all models are greatly improved compared with the Accuracy and Macro F1 of maximum pooling CNN-max, indicating that attention-pooling can pay attention to more features in the convolutional feature graph. Improved pooling effect. A comparison of several CNN and RNN models shows that RNN has higher accuracy than CNN, but the Macro F1 value is lower and greatly different from the CNN model, reaching a maximum of 8.68%. Therefore, the overall performance of several CNN models in this data set is better than THAT of RNN. As the text length of Weibo nCoV Data is shorter, the performance of CNN in the short text is better. Dsa-cnn has a slight performance improvement over the RoBERTa-C model. In addition, the Accuracy and Macro F1 of DSA-CNN are 3.28% and 3.15% higher than at-CNN respectively, indicating that attention pooling using affective word vectors has a better effect than random generation vectors. Moreover, the Accuracy of emotion feature mining from positive and negative emotional aspects is further improved.

In the NLPCC2014 data set, DSA-CNN is significantly improved compared with other CNN and LSTM models. Compared with RoBERTa-C, which has the second best performance, dSA-CNN's Accuracy and Macro F1 are improved by 2.18% and 0.96% respectively. Compared with AT-CNN and CNN-Max and AT-BilSTM and BiLSTM, at-CNN and AT-BilSTM using attention have some improvement over CNN-MAX and BiLSTM without using attention.

In the YF_DIANping data set, DSA-CNN still has the best accuracy, but the gap with AT-BILSTM is reduced. As the comments on e-commerce are generally longer, the advantages of the RNN model become more obvious when the text length becomes longer. However, the model proposed in this paper is still valid. The accuracy of RoBERTa-C on this dataset is not as good as other models, but the Macro F1 value is the highest, which may be because the pre-training model is not trained for e-commerce corpus, but the model has a strong ability to deal with unbalanced data.

4.5.2. Analysis of convergence speed and training time

In this paper, the training speeds of CNN-MAX, AT-CNN, BiLSTM, AT-BilSTM, and DSA-CNN neural networks in the NLPCC2014 data set are compared. Under the same learning rate, the convergence speeds of each model are compared as shown in Figure 3. The accuracy of CNN-MAX and BiLSTM models was very low in the early training period, and the two models reached optimal performance after the 13th and 9th Epoch respectively, with a slow convergence rate, and the accuracy tended to be stable in the late training period. At-CNN and AT-BilSTM use the attention mechanism to enable the network to pay attention to important information quickly and achieve the optimal performance AT about the 4th Epoch, and the convergence speed has been greatly improved. DSA-CNN adopts the pre-trained affective word vectors as the query vector, which reduces the impact of random initialization and further improves the convergence speed and optimal performance compared with AT-CNN and AT-BILSTM. Due to the fast convergence speed, DSA-CNN experienced the situation that the accuracy rate decreased due to over-fitting after the training of 15 epochs. In practice, the network training process can be ended in advance.

For different model structures and a different number of training parameters, the training time of the models will also be different. This paper further compares the total training time of several models and the training time when the optimal performance is achieved, as shown in Tables 10. Among them, CNN-MAX has the lowest total training duration, which only needs 75s. At-cnn and DSA-CNN have slightly increased time consumption due to increased training parameters. Due to the characteristics of RNN, the training time of the two

BiLSTM models is about 4 times that of DSA-CNN. By comparing the optimal duration of several models, the convergence speed of CNN-MAX is slow, and the optimal performance is slower than AT-CNN and DSA-CNN. Although the total training duration of DSA-CNN is 20 seconds longer than that of AT-CNN, the optimal training duration is the same, indicating that DSA-CNN does not slow down its effective training speed even if the network complexity is increased.

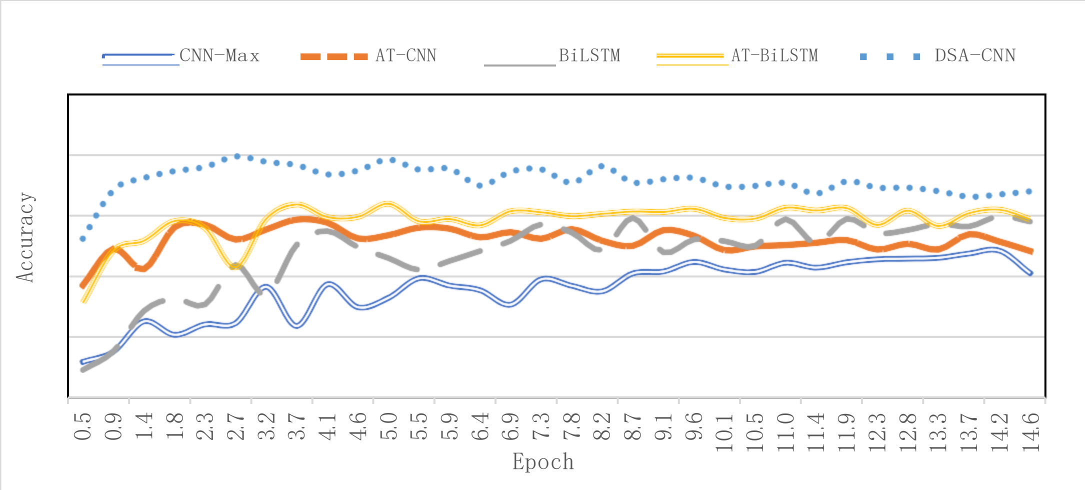

Fig. 3 Comparison of convergence speed of NLPCC2014 dataset

Table 10. Comparison of training duration in NLPCCC 2014 dataset

Model	Total duration(s)	Best duration(s)
CNN-Max	75	57.6
AT-CNN	83	24.3
BiLSTM	430	183.5
AT-BiLSTM	446	141.4
DSA-CNN	113	24.8

4.5.3. Model digestion analysis

To explore the influence of affective word vectors as query vectors of attention mechanism on the results, as well as the influence of the single-double attention pooling mechanism, the following comparison models are designed and compared with AT-CNN and DSA-CNN in NLPCC2014:

(1) DAT-CNN: The structure is consistent with the DSA-CNN model proposed in this paper, but the two attention-pooling query vectors are randomly initialized during training.

(2) PSA - CNN: use only the positive emotional words U_P in the pooling layer as query vectors, pooling of one-way attention.

(3) NSA - CNN: use only negative emotional words in pooling layer U_N as query vector, pooling of one-way attention.

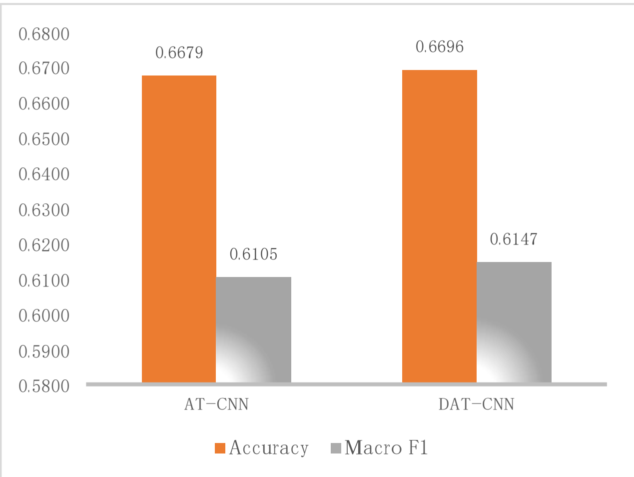

Fig. 4 Comparison of one-way and two-way random pooling

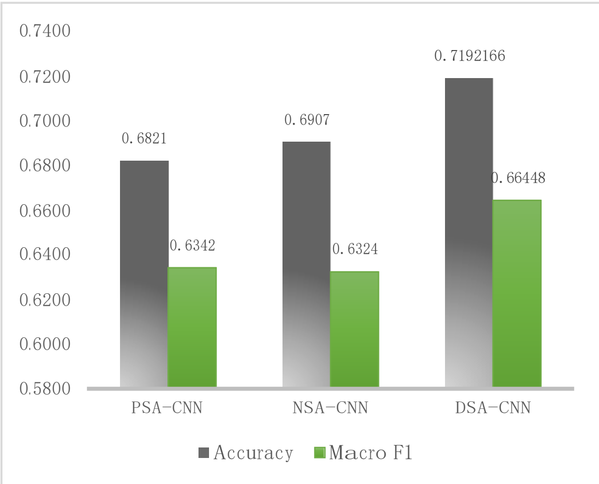

Fig. 5 Comparison of unidirectional and bi-directional affective word pooling

As can be seen from Figure 4, the experimental results of DAT-CNN and AT-CNN are not different, except for the Macro F1 index, which shows a certain improvement, and the accuracy of the two is the same. Compared with AT-CNN, DAT-CNN has one more feature extraction process of attention pooling in structure. However, it has little impact on the results, which may be due to the high similarity between the two randomly initialized query vectors and no distinguishing text features are extracted in the network, resulting in little impact on the classification results.

Combined with Figs. 4 and 5, PSA-CNN and NSA-CNN have significantly improved in both indicators compared with AT-CNN, indicating that more obvious emotional features can be extracted in attention pooling after affective word vectors are used as query vectors instead of random initialization, which is of great help to classification and discrimination. DSA-CNN is also significantly improved compared with PSA-CNN and NSA-CNN, and its accuracy is 3.31 and 2.50% higher respectively, indicating that two-way sentiment word pooling can mine more emotional information from the two aspects of positive emotion and negative emotion compared with one-way sentiment word pooling.

Compared with the randomly generated attention vector model DAT-CNN, the DSA-CNN model proposed in this paper has a great improvement, indicating that the trained bidirectional affective word vectors can promote the ability of emotion feature extraction in attention pooling and extract more effective features, avoiding the situation of low feature discrimination in DAT-CNN. The accuracy rate increased obviously. In addition, by comparing NSA-CNN and PSA-CNN, the overall performance of NSA-CNN is better than that of PSA-CNN, which may be due to the more discriminative features extracted from this data set by negative emotion word U_N as query vector.

4.5.4. Network learning ability analysis

To verify the learning ability of different neural networks, the data set is divided into training sets of different sizes. Yf_dianping with the most data was selected as the data set of this experiment, and its training set was randomly divided into five data sets with the sizes of 15000, 30000, 45,000, 60000, and 75000 according to the proportion of each class. Since RoBERTa-C is a pre-training model, it is excluded from this experiment. The learning effects of other models on data sets of different sizes are shown in the figure below.

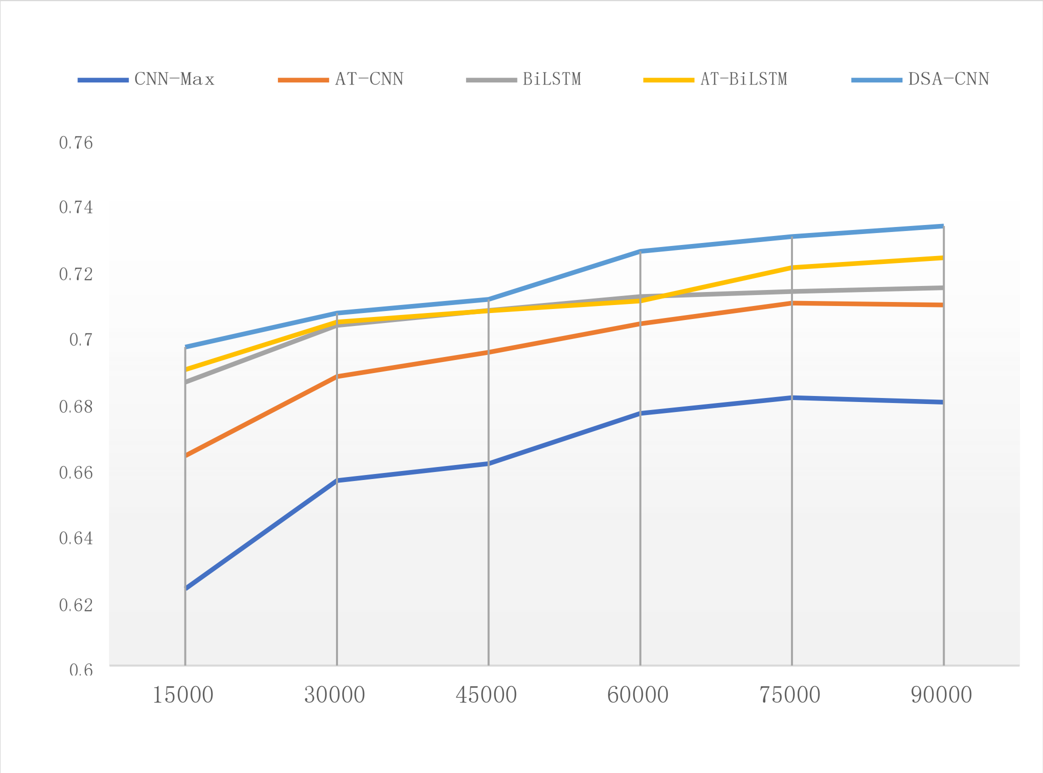

Fig. 6 Comparison of learning ability

As can be seen from Figure 6, when the size of the training set is 15000, the accuracy of the BiLSTM model and AT-BILSTM model both reach a high level, while the accuracy of the CNN-MAX model and AT-CNN model is relatively low AT this time, and the learning ability of LSTM model on small data set is stronger than that of ordinary CNN model. When the training set is increased to 30000, the accuracy of several models is significantly improved. After that, when the data set was further increased to 75,000, the accuracy of the CNN-MAX model and AT-CNN model was gradually improved, while the accuracy of BiLSTM was unchanged, while that of AT-BILSTM was slightly improved. For the model DSA-CNN proposed in this paper, on the one hand, its learning ability on small data sets is the same as that of LSTM, which is stronger than that of the ordinary CNN model, indicating that the trained affective word vectors enable CNN to obtain certain feature extraction ability on small data sets. Secondly, when the data set becomes larger, the accuracy of DSA-CNN also gradually improves to a small degree, indicating that the larger data volume enables the network to learn more knowledge.

4.5.5. Vector dimension analysis of emotional words

The dimension of affective word vectors will affect the feature extraction ability of the model. This paper analyzes this effect by training affective word vectors of different dimensions. In addition to the 100-dimensional bidirectional affective word vectors mentioned above, the same parameters in Section 4.2 were used to train the 50-dimensional, 150-dimensional, 200-dimensional, and 300-dimensional bidirectional affective word vectors, respectively. Comparative experiments were conducted on the NLPCC2014 data set. Model accuracy and total training duration are shown in Figs. 7 and 8.

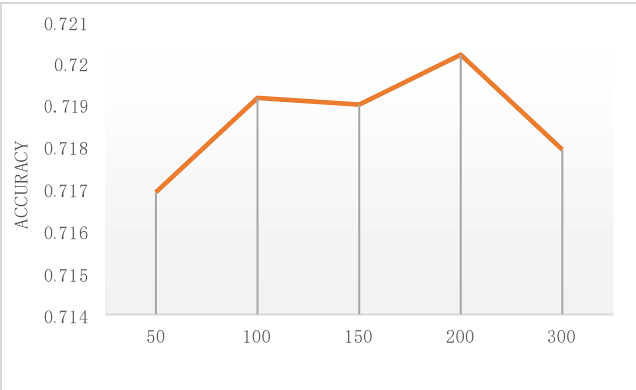

Fig. 7 Comparison of accuracy of affective word vectors dimension

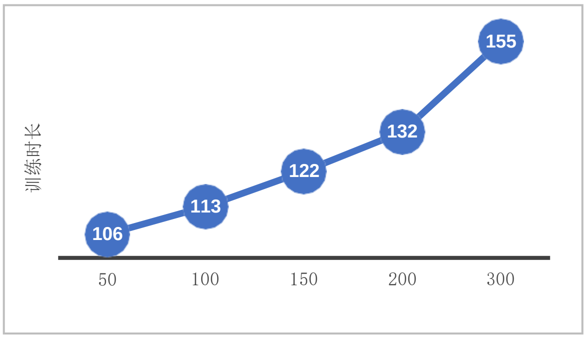

Fig. 8 Comparison of training duration of emotional word vector dimension

As can be seen from Fig. 7, when the affective word vectors increase from 50 dimensions to 100 dimensions, the model accuracy is greatly improved, indicating that 100 dimensions can reflect more features. From 100 to 150 dimensions, the accuracy rate has no change, but after using 200-dimension affective word vectors, the accuracy rate has a small increase; The accuracy of the 300-dimensional emotional word vector decreases due to its high dimension. In Figure 8, the training time of the model increases successively with the dimension of the word vector, which increases by 50 dimensions on average, and the training time of the model increases by about 9s. Through comprehensive consideration of the accuracy index and training duration index, the 100-dimension affective word vectors are the best, and the 200-dimension affective word vectors are the second.

4.5.6. Analysis of the training method of affective word vectors

This paper also compares the performance of bidirectional affective word vectors obtained from different sentiment corpora and different training methods. The corpus containing only Weibo text is called the Weibo corpus, and the corpus containing Weibo, Meituan, Jingdong, and other texts is called the mixed prediction database. The method proposed in 3.3 is called the population method. In addition, another training method is proposed: direct training after word segmentation is carried out in the corpus, instead of replacing markers, all positive and negative emotional words are extracted from the trained word vector files, and their centroids are obtained as positive and negative emotional word vectors respectively, which is called centroid method. The results of the following four groups of experimental training were compared as emotional word vectors in DSA-CNN :(1) microblog prediction base + population method; (2) Microblog prediction base + center of mass method; (3) Mixed prediction base + population method; (4) Mixed prediction library + center of mass method. The specific results are shown in the figure below.

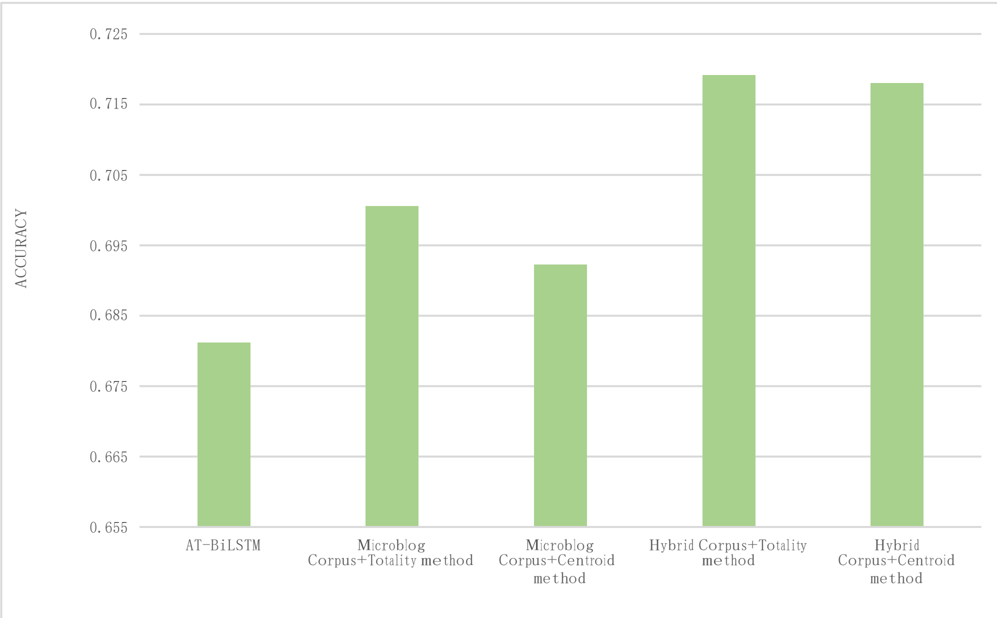

Fig. 9 Comparison of emotional word vector training methods

It can be seen from Fig. 9 that the DSA-CNN model of affective word vectors obtained by four methods has higher accuracy than at-BILSTM, which proves that the pre-training of affective word vectors by several methods is effective. Contrast different corpus training vector emotional words, using hybrid training corpus, the accuracy of two methods are higher than only using Weibo training corpus, two methods, that bigger, more abundant corpus contains more emotional information, help to improve accuracy in the process of the training vector emotional words, which makes the model accuracy. The comparison between the CPM and the POPULATION method shows that the accuracy of the population method is 0.83% higher than that of the CPM in the microblog corpus, and the accuracy of the two methods is almost the same in the mixed prediction database. It may be due to the small number of texts in the microblog corpus and the relatively small frequency of some emotional words, which leads to the reduced accuracy of the word vector of these emotional words training and the deviation of the calculated centroid. However, the population method eliminates part of the error by uniformly replacing emotional words as markers. If the training corpus is large enough and the frequency of each emotion word is large enough, the centroid method should also perform well.

4.5.7. Visual analysis of affective word vectors

To more clearly show the training results of bidirectional affective word vectors in different corpora and different methods, this paper visualizes the results of the four training methods mentioned in the previous section, uses principal component analysis (PCA) to reduce all vectors to a two-dimensional plane, and uses Matplotlib to draw scatter plots, as shown in Fig. 10. The trained bidirectional affective word vectors is represented by "x", and each specific emotion word is represented by a dot.

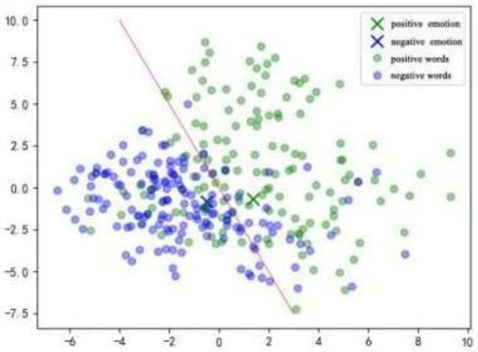

Fig. 10 Hybrid Corpus+Totality method

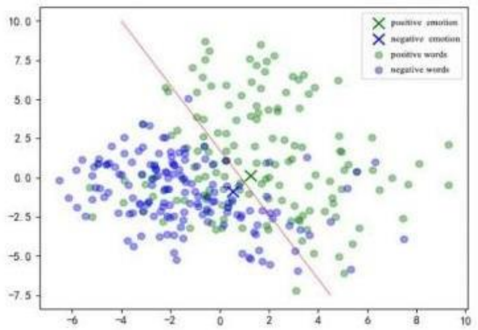

Fig. 11 Hybrid Corpus+Centroid method

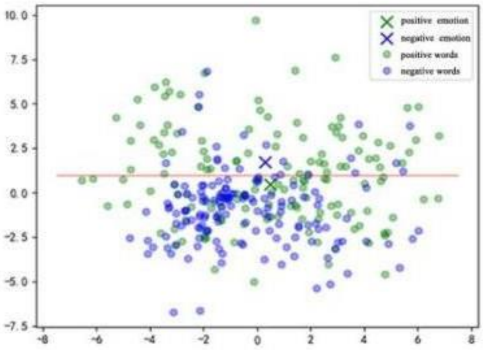

Fig. 12 Microblog Corpus+Totality method

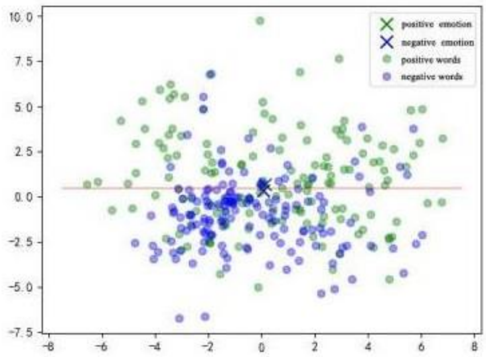

Fig. 13 Microblog Corpus+Centroid method

In the mixed corpus, it can be seen that positive words and negative words are divided into two categories, in which the distribution of negative words is more concentrated, while the distribution of positive words is slightly dispersed. This also explains to some extent why the overall performance of NSA-CNN is better than that of PSA-CNN because negative words are more discriminative. Compared with Figs. 10 and 11, the bidirectional affective word vectors obtained by the population method and the centroid method are distributed in different quadrants. The positive affective word vectors are closer to the majority of positive words, and the negative affective word vectors are closer to the majority of negative words, indicating that the bidirectional affective word vectors obtained by the two methods in the mixed corpus have certain effectiveness. The bidirectional sentiment word vector obtained by the population method has a long distance and a better effect.

In the microblog prediction database, positive words and negative words can still be divided into two categories, but the distribution is relatively scattered. Compared with Figure 12 and Figure 13, there is a certain distance between the bidirectional affective word vectors obtained by the population method, while the bidirectional affective word vectors obtained by the centroid method are basically in the same position without discrimination. The visualization results are consistent with the accuracy test results of the four methods in the previous section, indicating that the overall method proposed in this paper is more suitable for training affective word vectors in small corpora. In large corpora, the training results of both the global method and the center of the mass method are acceptable, and increasing the number of corpora can improve the accuracy of the results.

5. Conclusions

This paper proposes a combination of bipolar word attention mechanism essay this sentiment analysis method, and puts forward the concept of bipolar word vector, using existing resources to build the emotional dictionary, use emotional corpus training had positive and negative emotional words vector, as attention pooling query vector, and using bipolar pooling combined vector and attention, Features were extracted from positive emotion and negative emotion respectively to enhance the model's ability to capture short text emotion. Experiments show that the DSA-CNN model proposed in this paper achieves the best performance on all three data sets, and the convergence speed of the network is significantly improved due to the addition of the attention mechanism. In addition, the affective word vectors can make the model extract more obvious emotion features in attention pooling, avoid the problem of low differentiation of random pooling features, and promote the effect of attention pooling. The training method and training corpus of affective word vectors will influence the training effect and even affect the result of emotion classification.

There are also some shortcomings in this paper. For example, the mixed corpus used to train the use of bidirectional affective word vectors is still small, and the artificial method used to construct the fused sentiment dictionary is inefficient and cannot timely cover the new emotion words in the corpus. All these will affect the effect of bidirectional affective word vectors. In the next step, the construction method of the sentiment dictionary and the training method of the word vector can be improved to obtain a more accurate expression of the sentiment vector, and the influence of context on sentence emotion expression should be considered.

Conflicts of Interest

The authors declare that they have no competing interests.

Data Availability Statement

The data used to support the findings of this study are included within the article.

References

[1] Alsmadi I, Hoon G K. Term weighting scheme for short-text classification: Twitter corpuses[J]. Neural Computing and Applications, 2019, 31(8): 3819-3831.

[2] Cao J X, Xu S, Chen G J, et al. Regional topic discovery in online social networks [J]. Chinese Journal of Computers, 2017, 40(07): 1530-1542.

[3] Yin H, Yang S, Li J. Detecting Topic and Sentiment Dynamics Due to COVID-19 Pandemic Using Social Media[C]. Advanced Data Mining and Applications, 2020: 610-623.

[4] Yaqub U, Chun S A, Atluri V, et al. Analysis of political discourse on twitter in the context of the 2016 US presidential elections[J]. Government Information Quarterly, 2017, 34(4): 613-626.

[5] Siering M, Deokar A V, Janze C. Disentangling consumer recommendations: Explaining and predicting airline recommendations based on online reviews[J]. Decision Support Systems, 2018, 107: 52-63.

[6] Ren R, Wu D D, Liu T. Forecasting Stock Market Movement Direction Using Sentiment Analysis and Support Vector Machine[J]. IEEE Systems Journal, 2019, 13(1): 760-770.

[7] Ekman P. An argument for basic emotions[J]. Cognition and Emotion, 1992, 6(3-4): 169-200.

[8] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. arXiv e-prints, 2014: arXiv:1409.1556.

[9] Devlin J, Chang M-W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[J]. arXiv e-prints, 2018: arXiv:1810.04805.

[10] Asghar M Z, Subhan F, Ahmad H, et al. Senti-eSystem: A sentiment-based eSystem-using hybridized fuzzy and deep neural network for measuring customer satisfaction[J]. Software: Practice and Experience, 2021, 51(3): 571-594.

[11] He Y X, Sun S T, Niu F F, et al. A deep learning model with enhanced emotional semantics for microblog sentiment analysis[J]. Chinese Journal of Computers, 2017, 40(04): 773-790.

[12] Kim Y. Convolutional Neural Networks for Sentence Classification[C]. EMNLP, 2014.

[13] Yang Z, Yang D, Dyer C, et al. Hierarchical Attention Networks for Document Classification[C]. Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies, 2016: 1480-1489.

[14] Zhou P, Shi W, Tian J, et al. Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification[C]. Proceedings of the 54th annual meeting of the association for computational linguistics 2016: 207-212.

[15] Xu G, Yu Z, Yao H, et al. Chinese Text Sentiment Analysis Based on Extended Sentiment Dictionary[J]. IEEE Access, 2019, 7: 43749-43762.

[16] Kaity M, Balakrishnan V. An integrated semi-automated framework for domain-based polarity words extraction from an unannotated non-English corpus[J]. The Journal of Supercomputing, 2020, 76(12): 9772-9799.

[17] Wu F, Huang Y, Song Y, et al. Towards building a high-quality microblog-specific Chinese sentiment lexicon[J]. Decision Support Systems, 2016, 87: 39-49.

[18] Zhang X, Zhao J, Lecun Y. Character-level convolutional networks for text classification[C]. Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, 2015: 649–657.

[19] Yan L, Han J, Yue Y, et al. Sentiment Analysis of Short Texts Based on Parallel DenseNet[J]. Computers, Materials & Continua, 2021, 69(1): 51--65.

[20] Gan C, Wang L, Zhang Z, et al. Sparse attention based separable dilated convolutional neural network for targeted sentiment analysis[J]. Knowledge-Based Systems, 2020, 188: 104827.

[21] Zhou M, Liu D, Zheng Y, et al. A text sentiment classification model using double word embedding methods[J]. Multimedia Tools and Applications, 2020: 1-20.

[22] Wang X, Liu Y, Sun C, et al. Predicting Polarities of Tweets by Composing Word Embeddings with Long Short-Term Memory[C]. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2015: 1343-1353.

[23] Lai S, Xu L, Liu K, et al. Recurrent convolutional neural networks for text classification[C]. Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015: 2267–2273.

[24] Zuo E, Zhao H, Chen B, et al. Context-Specific Heterogeneous Graph Convolutional Network for Implicit Sentiment Analysis[J]. IEEE Access, 2020, 8: 37967-37975.

[25] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017: 6000–6010.

[26] Yin R, Li P, Wang B. Sentiment Lexical-Augmented Convolutional Neural Networks for Sentiment Analysis[C]. 2017 IEEE Second International Conference on Data Science in Cyberspace (DSC), 2017: 630-635.

[27] Luo L-X. Network text sentiment analysis method combining LDA text representation and GRU-CNN[J]. Personal and Ubiquitous Computing, 2019, 23(3): 405-412.

[28] Alharbi A S M, De Doncker E. Twitter sentiment analysis with a deep neural network: An enhanced approach using user behavioral information[J]. Cognitive Systems Research, 2019, 54: 50-61.

[29] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate[J]. CoRR, 2015, abs/1409.0473.

[30] Cheng Y, Yao L, Xiang G, et al. Text Sentiment Orientation Analysis Based on Multi-Channel CNN and Bidirectional GRU With Attention Mechanism[J]. IEEE Access, 2020, 8: 134964-134975.

[31] Chen K, Liang B, Ke W D, et al. Sentiment analysis of Chinese microblog based on multi-channel convolutional Neural Network[J]. Journal of Computer Research and Development, 2018, 55(05): 945-957.