Emotional Polarity Attention Mechanism for Text Sentiment Analysis

Junwei Zhang[1], Tianheng Wang[1], Chenyang Wang[2,3(✉)], Yude Bai[4(✉)], Yupeng Zhang[1], and Yantao Li[1]

[1] Center for Artificial Intelligence and Intelligent Medicine, Hangzhou Institute of Medicine, Chinese Academy of Sciences, Zhejiang Province, 310018, China.
[2] College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, Guangdong, 518060, China.
[3] Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), Shenzhen, Guangdong, 518132, China.
[4] School of Software, Tiangong University, Tianjin, 300387, China.
{zhangjunwei, wangtianheng, zhangyupeng, liyantao}@him.cas.cn
chenyangwang@ieee.org baiyude@tiangong.edu.cn

Abstract. Text Sentiment Analysis (TSA) is a classic research topic in the field of Natural Language Processing (NLP) and has important application value for downstream tasks. Previous TSA methods focus on accurately representing emotional words in the semantic space to complete the recognition of text emotions. However, traditional methods are difficult to establish the intrinsic strong connections between emotional words in fine granularity and to accurately distinguish emotional tendencies. The research work of psychologists such as Carroll Ellis Izard, David Krech, and Robert Pluchik pointed out that for humans, our emotions can be compounded from the most basic emotions to construct many types of emotional states. Based on the classic attention mechanism, this paper proposes an Emotional Polarity Attention Mechanism (EPAM), which establishes strong connections between emotional words through emotional compounding theory, to enhance the understanding and representation accuracy of emotional words. We embed EPAM as a network layer into the current mainstream text classification models to build experimental group models and then verify the effectiveness of EPAM. Comparative experiments on four standard text classification datasets confirm that our model is effective and able to mine the intrinsic connections of emotional words.

Keywords: Text Sentiment Analysis · Emotional Compounding Theory · Attention Mechanism

1 Introduction

Text Sentiment Analysis (TSA) aims to identify the emotional tendencies conveyed by text based on emotional words, and then infer the author's stance on an event or topic [9]. Coarse-grained TSA models generally divide the sentiment tendency of text into *Positive*, *Negative* and *Neutral* sentiment levels.

Fine-grained TSA models will divide the emotional tendency into more detailed emotional states, such as *Happy*, *Sad*, *Angry*, *Rage*, etc. TSA is a classic research topic in the field of Natural Language Processing (NLP), which has an important application value on many downstream tasks, such as public opinion analysis [37, 18], consumer behavior prediction [1, 8], and product recommendation [3, 10].

Previous TSA models mainly leveraged contextual word or text representation learning technology to obtain word representations of emotional words or text representations of emotional texts, and then implement emotion recognition [24, 33, 35, 36, 34]. Mikolov et al. [22] drew on the idea of the Log-Bilinear model [23] to propose the Word2Vec model, and implemented two frameworks, CBOW and Skip-gram, which are often used to obtain word representations. Following RNN, LSTM and GRU, Kim [14] uses a one-dimensional convolutional neural network to obtain text representations of emotional texts and then implement text sentiment analysis. After Vaswani et al. [29] proposed the attention mechanism, researchers have successively built pre-trained language models such as BERT and GPT, and achieved unprecedented success in the field of NLP. With the support of pre-trained language models, TSA tasks have also made great progress. Most TSA tasks attempt to use pre-trained language models to obtain emotional text representations and then implement emotion recognition [17].

Traditional context-based representation learning methods mainly learn the word representation of the target word or the text representation of the target text based on all vocabulary in the entire context. **These methods can only learn the relative positional relationship between emotional word representations in the semantic space, and it is difficult to establish an intrinsic strong connection between the representations.** Therefore, the emotional word representation and emotional text representation obtained by the traditional context-based representation learning methods are not suitable for TSA tasks that have strict requirements on emotional discrimination, which will lead to misjudgment of emotional tendencies.

The research work of psychologists such as Carroll Ellis Izard [11], David Krech [15], and Robert Plutchik [25] pointed out that "for humans, our emotions can be compounded from the most basic emotions to construct many types of emotional states [28]." In other words, all human emotions can be expressed as a superposition of several basic emotions. Based on this, this paper constructs an **Emotional Polarity Attention Mechanism (EPAM)** based on the classic attention mechanism, and tries to establish an inherent strong connection between emotional words through superposition of basic emotional words, that is **emotional compounding theory**, to enhance the representation ability of emotional words and texts. Specifically, we obtain word representation based on the superposition form of basic emotional words by transforming the classic attention mechanism model, and then obtain text representation. Among them, the basic emotional words use the most primitive emotional polarity words, such as *Ecstasy*, *Admiration*, *Terror*, *Amazement*, *Grief*, *Loathing*, *Anger*, *Vigilance*, etc. In addition, we design an empty state "*None*" to represent words without emotional tendencies. In the implementation process, EPAM is embedded as a network layer into the current and mainstream text classification models to construct the experimental group models, and then verify its effectiveness on four mainstream TSA datasets.

The contributions of this paper are summarized as follows:

- Combined with the emotion composite theory in psychology, a representation method based on the superposition of emotional polarity words is proposed;
- Based on the classic attention mechanism, the attention mechanism model of emotional polarity word superposition is constructed;
- Text emotion classification models are built based on the proposed emotion polarity attention mechanism, and experimental results prove that this scheme can establish strong intrinsic connections between emotional words.

2 Related Work

2.1 Machine Learning-based TSA Methods

TSA methods based on Machine Learning (ML) technology use classic ML technology to mine the explicit features of emotional texts to complete text classification. Dey et al. [2] used the Naive Bayes algorithm and the KNN algorithm to implement TSA on movie reviews and hotel evaluation information. Jin et al. [12] used word vectorization hidden Markov framework to deal with TSA, which uses the sequence labeling method to mark various entities related to the task and the views corresponding to the entities, and then determine the emotional tendency of the text. Liu et al. [21] used a collaborative training algorithm to implement semi-automatic annotation of the corpus, and used a support vector machine algorithm to complete TSA. Liu et al. [19] combined conditional random fields with large entropy model to conduct TSA, which extracted feature information such as words, word positions, context information, etc., and used conditional random fields for sequence annotation to extract Unigram and Bigram feature representations, and finally select the large entropy model to determine the emotional tendency of the entire sentence.

TSA methods based on ML have attracted great attention from researchers because of their mature technology and high model accuracy. However, these methods have certain limitations when dealing with large-scale complex tasks, because ML methods require annotation and feature extraction of text corpus at the lexical level.

2.2 Deep Learning-based TSA Methods

Representation learning methods in Deep Learning (DL) are often used in TSA tasks to obtain emotional word representations or emotional text representations, and then implement TSA [30]. Dong et al. [6] proposed a TSA method based on adaptive recurrent neural network, which uses context information and syntactic rules to adaptively learn the emotional labels of emotional words, thereby achieving text classification. Chang et al. [4] proposed a fine-grained sentiment analysis method based on DL, which involves research objects from different corpora and covers analysis perspective from vocabulary level to chapter level. Wang et al. [31] integrated a tree-shaped long- and short-term memory neural network at the output end of the convolutional neural network to enhance the recognition ability of deep semantic information by adding sentence structure features. Eric et al. [7] proposed a TSA method based on image and text fusion to solve that traditional Weibo sentiment analysis methods ignore pictures, special symbols and contextual information, resulting in low accuracy of sentiment analysis methods. Tang et al. [27] proposed a message-level Weibo sentiment analysis system, which combines specific emotional word vectors with manually selected emoticons, semantic dictionaries and other feature information, and uses support vector machines to perform text classification.

Representation learning methods based on DL technology can avoid the troubles of manual labeling and filtering features of traditional ML. However, representation learning methods driven by big data only obtain the potential positional relationships between representations in the semantic space, that is a weak connection, so they are not suitable for TSA tasks. **This paper will construct emotional word representation and emotional text representation methods based on the emotional compounding theory in psychology to establish a strong connection between representations and thereby enhance the distinguishability of representations in the emotion recognition process.**

3 Methodology

3.1 Emotional Compounding Theory

Carroll Ellis Izard [11] is an American research psychologist known for his contributions to Differential Emotions Theory, which used factor analysis to propose eleven basic human emotions, namely *Interest*, *Surprise*, *Pain*, *Disgust*, *Joy*, *Anger*, *Fear*, *Sadness*, *Shyness*, *Contempt* and *Guilt*. In the definition of the **emotional compounding theory**, there are three types of compound emotions generated by the combination of basic emotions: a mixture of basic emotions, such as *interest-pleasure*, *fear-shyness*, *fear-guilt-pain-anger*, etc.; a mixture of basic emotions and internal drives, such as *drive-interest-pleasure*, *pain-fear-anger*, etc.; a combination of basic emotions and cognition, such as *vitality-interest-anger*, *suspicious-fear-guilt*, etc. There are hundreds of compound emotions, some of which can be named. For example, the compound *anger-disgust-contempt* could be named *Hostility*. *Anger* is a hot emotion, and *Contempt* and *Disgust* are both cold emotions, and their combination determines the degree of aggression in hostility. Another example is the compound *fear-guilt-pain-anger*, which is a classic emotion of anxiety, where the relative strengths of the *Anger* and *Pain* components determine whether anxiety is excitatory or inhibitory. Such complex emotions are difficult to name.

American psychologist David Krech [15] regards *Happiness*, *Sadness*, *Anger* and *Fear* as four basic emotions. Robert Plutchik [25] first published the cone

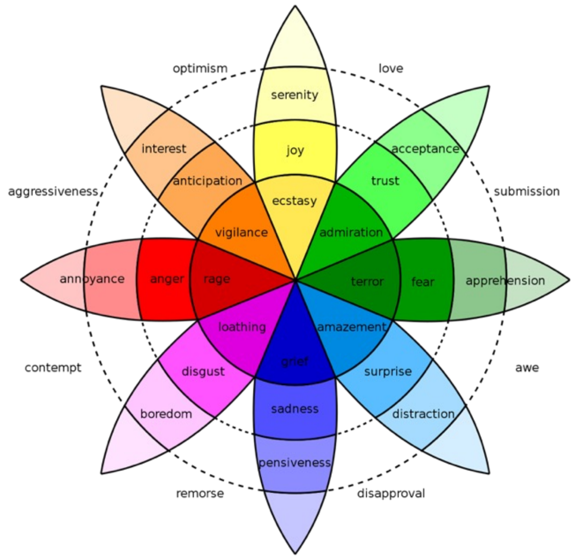

Fig. 1. Plutchik's Wheel of Emotions.

model (3D Model) or wheel model (2D Model) to describe the degree of correlation of emotions, as shown in Fig. 1. He identified eight major emotional pairs: *Joy* and *Sadness*, *Anger* and *Fear*, *Trust* and *Disgust*, and *Surprise* and *Anticipation*. Additionally, his circular model connects the concepts of the emotional circle and the color wheel. As indicated by the colors, primary emotions can be expressed in different intensities and can be mixed with each other to form different moods. Pluchick believed that human beings have eight basic emotions, namely *Fear*, *Anger*, *Joy*, *Sadness*, *Trust*, *Disgust*, *Curiosity*, and *Surprise*. Other emotions are acquired and are the result of compounding of basic emotions.

Based on the above-mentioned research by psychologists, we realize that many human emotions can actually be reduced to a mixture of several basic emotions. In other words, many human emotions can be described as superimposed forms of several basic emotions. **For TSA tasks, it is inappropriate to construct isolated word vectors for all emotional words, which will lose the intrinsic strong connections between emotional words and is not conducive to revealing the true emotional tendencies of humans. Based on this, this paper will try to leverage the Emotional Compounding Theory to construct deep relationships between emotional words, thereby improving the accuracy of TSA models.**

3.2 Emotional Polarity Attention Mechanism

Classical attention mechanism: The essence of the classic attention mechanism is actually an addressing process, as shown in Fig. 2. Given a task-related

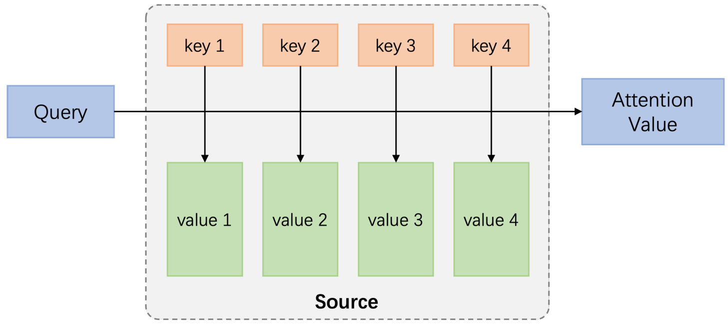

Fig. 2. Classical Attention Mechanism.

query $Query$, the output of the attention block is obtained by calculating the attention distribution of $Query$ with the key Key and appending it to the value $Value$. This process is actually a manifestation of the attention mechanism easing the complexity of the neural network model. It does not need to pass all N input information to the neural network for calculation, but only needs to select some task-related information to pass to the neural network. The attention mechanism can be divided into three steps: information input, calculate the attention distribution, and calculate the weighted average of the input information according to the attention distribution. The three steps are introduced in the following:

- The first step is to calculate the similarity or correlation between $Query$ and Key. Different measurement functions or calculation mechanisms can be introduced to calculate the similarity or correlation between $Query$ and a certain key_i, in which key_i belongs to Key. The most common measurement methods include the dot product of the two vectors, the Consine similarity of the two vectors, or the solution with the help of additional neural networks. The specific formula is as follows:
  Dot product operation:
  $$Sim_i = Similarity(Query, key_i) = Query \cdot key_i; \tag{1}$$
  Consine similarity operation:
  $$Sim_i = Similarity(Query, key_i) = \frac{Query \cdot key_i}{\|Query\| \cdot \|key_i\|}; \tag{2}$$
  MLP neural network:
  $$Sim_i = Similarity(Query, key_i) = MLP(Query, key_i). \tag{3}$$
- The second step normalizes the calculation results of the first step using the Softmax normalization function. This operation can not only normalize

the similarity or correlation results, but also highlight the weight of important elements through the inherent mechanism of the Softmax function. The generally used calculation formula is as follows:

$$\alpha_i = Softmax(Sim_i) = \frac{e^{Sim_i}}{\sum_{j=1} e^{Sim_j}}. \tag{4}$$

– The third step is to weight and sum $value_i$ according to the weight α_i, in which $value_i$ belongs to *Value*. The specific calculation formula is as follows:

$$Attention(Query, Source) = \sum_{i=1} \alpha_i \cdot value_i. \tag{5}$$

Through the calculation of the above three steps, the *Attention Value* for *Query* can be obtained. Most of the attention mechanism calculation methods currently used conform to the above three-step abstract calculation process.

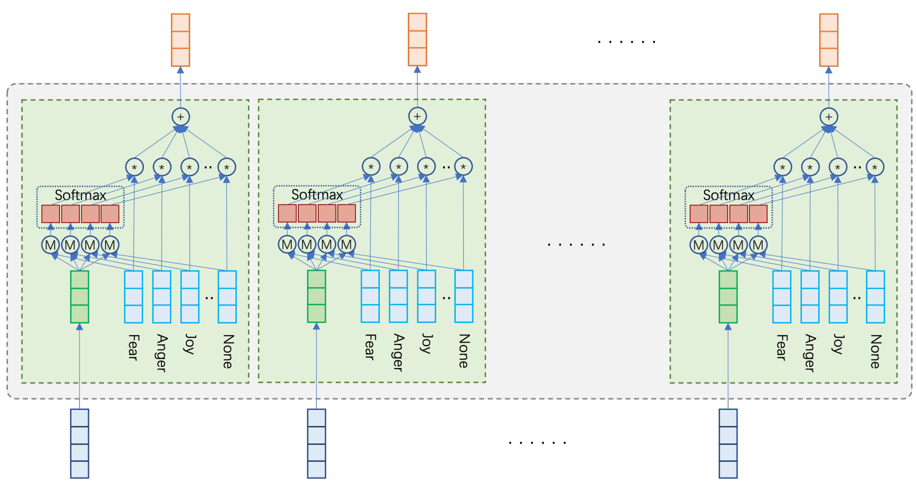

Fig. 3. Schematic diagram of EPAM. The circled M represents the measurement operation, the circled $\star$ represents the multiplication operation of a scalar and a vector, and the circled + represents an addition operation of a vector and a vector.

Emotional polarity attention mechanism: As shown in Fig. 3, the internal structure of the Emotional Polarity Attention Mechanism (EPAM) can be decomposed into multiple small components with the same structure and similar functions. The function of each component is to learn a superposition state according to the emotional polarity words, such as a probability superposition form of some emotional polarity word vectors, such as *Fear*, *Anger*, *Joy*, etc. Emotional polarity words use the eight basic emotions proposed by Pluchick, and also include a non-emotional word *None*, which is used to represent words that do not have emotional tendencies in the input information. The above word vectors are derived from text vectors of corresponding word meaning definitions in the dictionary.

For a given word vector in the target text, Q_i, $i \in \{1, 2, \cdots, n\}$, where n represents the number of word vectors in the target text, it will be passed to the corresponding component block for further processing. The word vector Q_i can be a vector learned by a language model, or a vector generated by a vector learning model such as One-hot and Word2vec representation. However, since the above three methods have their own advantages, they can be selected according to the needs in specific tasks. In this paper, we will use the above three methods as the input vectors to train the model and compare their pros and cons. In addition, we also need to carry out vectorized representation of the eight basic emotions and one non-emotional tendency marker *None*, E_k, $k \in \{1...m\}$, where m represents the number of emotional polarity words and non-emotional tendency marker, and emotional polarity word vectors all use the same representation method in each component block.

For the word vectors in the target text and emotional polarity word vectors, we implement linear transformation operations to obtain $query_i \in Query$, $key_k \in Key$ and $value_k \in Value$, namely

$$query_i = W_1 Q_i, \quad key_k = W_2 E_k, \quad \text{and} \quad value_k = W_3 E_k, \tag{6}$$

where W represents a matrix to realize the linear operation on Q_i and E_k, and the subscripts are used to distinguish different matrices. W is the parameter matrix that needs to be learned from the data, and the specific value is determined by the training process.

After getting *Query*, *Key*, we use $query_i$ to measure key_k, that is, the dot product operation, to get a similarity or correlation value between $query_i$ and key_k,

$$sim_{i,k} = Similarity(query_i, key_k) = query_i \cdot key_k, \tag{7}$$

which is shown in Fig. 3 by circled M. We perform a normalization operation on the obtained $sim_{i,k}$,

$$\alpha_{i,k} = Softmax(sim_{i,k}) = \frac{e^{sim_{i,k}}}{\sum_{k=1}^{m} e^{sim_{i,k}}}. \tag{8}$$

Thus, we have obtained the similarity between the word vector in the target text and each emotional polarity word vector, or called the probability distribution of similarity or correlation. According to this probability distribution, we can construct the word vector in the target text as a superposition state of basic emotional words,

$$Attention(query_i, Source) = \sum_{k=1}^{m} \alpha_{i,k} \cdot value_k. \tag{9}$$

At this point, we have completed the overall structural description of EPAM. It is obvious from the mathematical description that EPAM has strong similarities with the traditional attention mechanism. For example, the same measurement method can be used, and they are both vector-to-vector mappings. The difference is that the output of the traditional attention mechanism is the superposition of input value vectors, while the output of EPAM is the superposition of emotional polarity word vectors. Moreover, it can be seen from the mathematical form or structural diagram that although the number of emotional polarity words needs to be determined according to the specific situation, it is far lower than the number of word vectors superimposed by the traditional attention mechanism.

How to choose emotional polarity words is completely based on the TSA task, and it can also be said to be determined by the output categories of the TSA system. If the output is positive or negative, positive or negative, like or dislike, etc., then the emotional polarity word vectors can choose an orthogonal emotional polarity words. When the output category is a fine-grained emotion category, it can be selected according to the emotion classification methods of Carroll Ellis Izard [11], David Krech [15], and Robert Plutchik [25].

3.3 TSA Model with Emotional Polarity Attention Mechanism

In order to verify the effectiveness of the EPAM for the TSA task, we construct our TSA model based on classic and mainstream text sentiment classification models, such as FastText, TextCNN, TextRNN, TextBiRNN, TextAttBiRNN, HAN, RCNN, RCNNVariant, etc. We embed the EPAM as a network layer between the classification layer and the model layer of the mainstream text sentiment classification models. In addition, all TSA models will use the pre-trained language model BERT, Word2vec model and One-hot model to obtain the word vectors in the target text and emotional polarity word vectors respectively.

4 Experiments

4.1 Datasets and Evaluation Metrics

We select four general binary sentiment classification datasets to validate our model, IMDB[5], Review Polarity v2.0[6], Rotten IMDB v1.0[7] and RT-Polarity Data v1.0[8]. These four datasets are used by the TSA task and have certain representativeness. The reason for choosing the binary classification dataset is that the binary classification task is often used to verify the effect of the new model, and in the TSA task, machine learning algorithms have reached the upper limit in the binary classification task, which is more conducive to verifying the effect of our model.

[5] https://www.imdb.com/interfaces/
[6] https://www.cs.cornell.edu/people/pabo/movie-review-data/
[7] https://www.imdb.com/interfaces/
[8] https://emilhvitfeldt.github.io/textdata/reference/dataset_sentence_polarity.html

The division ratio of training, verification and test sets: Some datasets have been split into training and test sets but not validation sets, while some datasets have not been split at all. In order to be compatible with the division ratios of all datasets, here we use a division ratio of $1:1$ to divide the training set and the remaining data, and on this basis, the remaining data are divided into the verification set and the test set in a ratio of $2:8$. The division ratio of this kind of dataset can be well compatible with the division ratio of existing datasets. The disadvantage of this division ratio is that the number of validation set is relatively small, but we made this sacrifice for a larger testset.

Experimental evaluation metrics: The experimental evaluation metrics of this paper are similar to other TSA papers, and the accuracy rate is used as the only evaluation metric. At the same time, in order to reduce the experimental deviation caused by chance, we conduct ten groups for each experiment, and the final results are taken as the average value.

4.2 Baseline Models and Parameter Settings

Baseline models: We construct the experimental group models by adding an EPAM layer to some well-known text classification models, and the comparison group models by adding a fully connected layer to the original models. What needs to be emphasized here is that we try to achieve the same number of parameters to be learned in the embedded EPAM layer and the embedded fully connected layer, but it is difficult to add only one fully connected layer. Therefore, the number of parameters they need to learn will vary in quantity, but it will not substantially affect the experimental results. The original models used to construct the experimental group models and the comparison group models include FastText [13], TextCNN [5], TextRNN [20], TextBiRNN [20], TextAttBiRNN [26], HAN [32], RCNN [16] and RCNNVariant [16].

Parameters settings: In this paper, we set the following parameters to conduct the experiments. *Max_features* indicates the maximum number of features used by the model, that is the number of vocabulary, *Max_features=5000*. *Maxlen* indicates the maximum length of the sentence input into the model, *Maxlen=400*. If the length of the sentence input into the model is greater than *Maxlen*, the extra part will be removed; if it is less than *Maxlen*, it will be filled with zeros. *Batch_size* indicates the number of samples learned in each batch, *Batch_size=32*. For deep learning, if only one sample is learned in each batch, the learning effect will fall into a jittering state and be difficult to converge; if a larger value is used, it will help the model converge and obtain a better learning effect. *Embedding_dims* indicates the dimension of word embedding, *Embedding_dims=50*. *Epochs* indicates the training rounds of the model, and the value is set to *Epochs=200*. During the implementation process, all models involved in this paper use early stopping to end the model training process early. When all models have been trained continuously for 100 times and the loss does not decrease, the model training process will end.

4.3 Experimental Results and Analysis

In order to reduce the impact of different representation methods on the final experimental results, we used One-hot, Word2vec, and pre-trained language models as the embedding layer to implement experiments. The comparative experiments are divided into three groups, One-hot, Word2vec, and pre-trained language model BERT are used to obtain word embedding respectively.

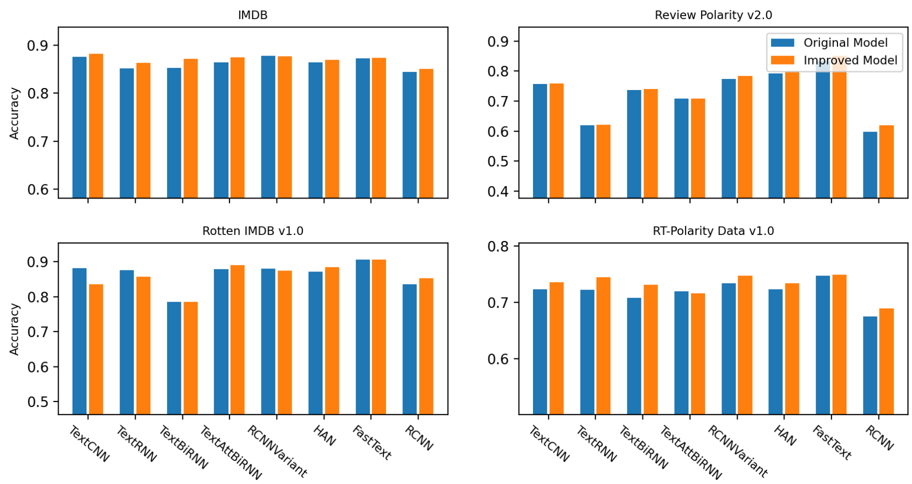

Fig. 4. Comparison between the proposed models and the original models based on One-hot model as the embedding layer under IMDB, Review Polarity v2.0, Rotten IMDB v1.0 and RT-Polarity Data v1.0 datasets.

One-hot as the embedding layer: The One-hot model uses binary encoding to assign unique representations to all words appearing in the corpus, where all words can be represented as a unique string that is used to distinguish words. Although the method of using the One-hot model to create word vectors is prone to shortcomings such as dimensionality explosion and data sparseness, it is adopted by deep learning models because of its simple implementation and ability to effectively distinguish words. In order to verify the broad applicability of our model, our experiments also considered the method of obtaining word vectors based on the One-hot model. Since the word vector obtained by the One-hot model has the problem of data sparseness, we will use a linear layer to linearly transform the original vector and compress it into a smaller space to achieve the purpose of dimensionality reduction.

The experimental results under the dataset IMDB and Review Polarity v2.0 are shown in Fig. 4. From the experimental results, we can see that in the case of One-hot representation of word vectors, the accuracy of the experimental group models in the IMDB dataset has improved. However, the performance of the our models shows a similar trend compared with the original models in the Review Polarity v2.0 dataset. The possible reason for this result is that the Review Polarity v2.0 dataset has reached the best situation and there is not much room for improvement, or it may be that our models are not suitable for this dataset.

The experimental results under the dataset Rotten IMDB v1.0 and RT-Polarity Data v1.0 are shown in Fig. 4. It can be seen from the diagram that under the Rotten IMDB v1.0 dataset, the experimental results of the experimental group did not improve on the three models, TextCNN, TextRNN, and RCNNvariant, and there was a significant decline in the models, TextCNN and TextRNN. Among the three models, other models have improved significantly, but overall, the improvement is not large. In the RF-Polarity Data v1.0 dataset, all models have improved except for TextAttBiRNN. Overall, the performance of our models on this dataset is excellent.

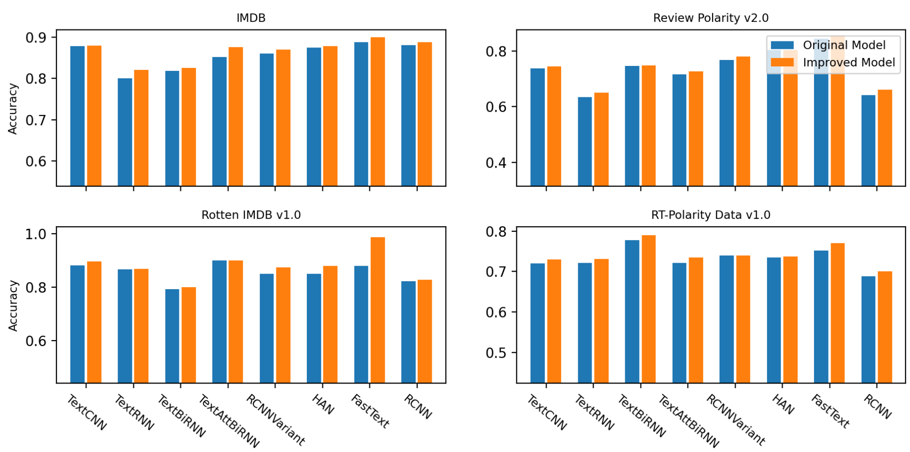

Fig. 5. Comparison between the proposed models and the original models based on Word2vec model as the embedding layer under IMDB, Review Polarity v2.0, Rotten IMDB v1.0 and RT-Polarity Data v1.0 datasets.

Word2vec as the embedding layer: Before the language model was proposed, the word vector representation method based on the Word2vec model had always been the mainstream word embedding method. One of the significant advantages of the Word2vec model is that it can calculate addition and subtraction operations between vectors, and can get very meaningful vectors, such as "*Father Emperor*" + "*Mother Queen*" = "*Prince*" or "*Princess*". However, the Word2vec model also has its own shortcomings. For polysemous words, this method is difficult to represent the mixed form of multiple word meanings. The Word2vec model is not always beneficial for TSA or certain NLP tasks.

For TSA task, it will greatly reduce the accuracy of text classification. In this part of the experiment, we first learn the word vector representation of each word through the Word2vec model, and input the obtained word vector into the experimental models.

The experimental results under the dataset IMDB and Review Polarity v2.0 are shown in Fig. 5. It can be seen from the experimental results that when using the Word2vec model to obtain word vectors, the accuracy of the experimental group models on the IMDB dataset has improved, but on the Review Polarity v2.0 dataset, the accuracy is not obvious promote. The possible reason for this result is that the Review Polarity v2.0 dataset has reached the best situation, and there is not much room for improvement. In the method of characterizing word vectors with the One-hot model, the same results also appears on this dataset, indicating that the dataset has reached its optimal situation.

The experimental results under the dataset Rotten IMDB v1.0 and RT-Polarity Data v1.0 are shown in Fig. 5. It can be seen from the experimental results that under the Rotten IMDB v1.0 dataset, except for FastText and HAN, other models did not significantly improve the experimental results. Under the RF-Polarity Data v1.0 dataset, the experimental group models have not improved except under the RCNNVariant model, and other models have improved. It can be seen that under this dataset, the performance of our models is well.

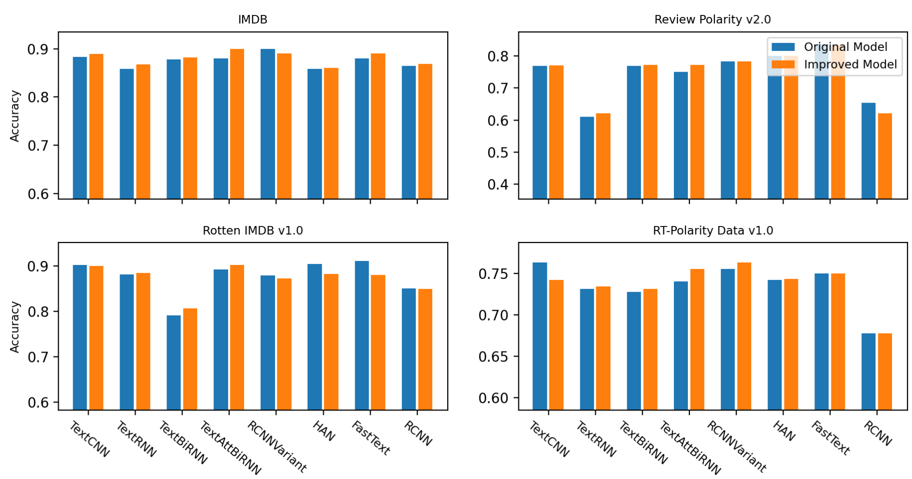

Fig. 6. Comparison between the proposed models and the original models based on pre-trained language model BERT as the embedding layer under IMDB, Review Polarity v2.0, Rotten IMDB v1.0 and RT-Polarity Data v1.0 datasets.

BERT as the embedding layer: The introduction of large language models based on the attention mechanism, such as BERT, GPT, Llama, etc., has brought a new dawn to NLP tasks and greatly improved the effectiveness of the original models. This paper uses the pre-trained language model BERT to obtain word vectors without fine-tuning learning. We input the text into the BERT model to obtain the context-based word representation of each word, and then input the word vectors into our experimental group models and comparison group models for model training and learning. That is to say, we replace the embedding layer with the large language model BERT.

The experimental results under the dataset IMDB and Review Polarity v2.0 are shown in Fig. 6. It can be seen from the experimental results that when word vector representation is obtained from the pre-trained language model BERT, the experimental results on the IMDB dataset are significantly improved; the experimental results on the Review Polarity v2.0 dataset are not significantly improved. The experimental results are the same as the first two sets of experiments, which means that this dataset has reached the upper limit of its performance and there is not much room for improvement.

The experimental results under the dataset Rotten IMDB v1.0 and RT-Polarity Data v1.0 are shown in Fig. 6. It can be seen from the experimental results that under the Rotten IMDB v1.0 dataset, the experimental results of our models are not competitive with the four models, TextCNN, RCNNVariant, HAN and FastText, and are significantly lower than the comparison models. Compared with models TextBiRNN and TextAttBiRNN, our models show some advantages. In the RF-Polarity Data v1.0 dataset, our models outperform the contrasting models except TextCNN. Overall, the performance of our models on RT-Polarity Data v1.0 dataset is competitive.

5 Conclusions

This paper starts from human cognitive psychology, based on the research work of psychologists on emotion, such as Carroll Ellis Izzard, David Krech, and Robert Plutchik, and proposes to use the form of emotional compounding to represent the emotional words involved in the TSA task. Establishing the inner strong connection between emotional words is necessary because the quality or accuracy depends entirely on understanding the emotional words in the text. Traditional methods do not attempt to construct strong correlations or connections between emotional words, but only establish relative positional relationships between emotional words in the semantic space, which is an obvious flaw of traditional works. Based on the classic attention mechanism, this paper builds a emotional polarity attention mechanism, and embeds the emotional polarity attention mechanism into the current and mainstream text classification models as a network layer to construct the experimental group models, and then verify the validity and correctness of the representation method. The comparative experiments on four datasets basically prove that our model is effective and can mine the intrinsic strong correlation of emotional words.

In the future work, we will try to develop the network layer of the attention mechanism vertically, and expand the advantages, so as to achieve the purpose of greatly improving the model effect. We believe that the value of this work is significant, and it will affect the emotional word vectors constructed in the future.

Acknowledgements Our work is supported by the National Key Research and Development Program of China (No. 2022YFC3600902).

References

1. Badea, L.M.: Predicting consumer behavior with artificial neural networks. Procedia Economics and Finance **15**, 238–246 (2014)
2. Berger, A., Della Pietra, S.A., Della Pietra, V.J.: A maximum entropy approach to natural language processing. Computational linguistics **22**(1), 39–71 (1996)
3. Bonab, H., Aliannejadi, M., Vardasbi, A., Kanoulas, E., Allan, J.: Cross-market product recommendation. In: CIKM. pp. 110–119 (2021)
4. Chang, G., Huo, H.: A method of fine-grained short text sentiment analysis based on machine learning. Neural Network World **28**(4), 325–344 (2018)
5. Chen, Y.: Convolutional neural network for sentence classification. Master's thesis, University of Waterloo (2015)
6. Dong, L., Wei, F., Tan, C., Tang, D., Zhou, M., Xu, K.: Adaptive recursive neural network for target-dependent twitter sentiment classification. In: ACL. pp. 49–54 (2014)
7. Gitau, E.: An approach for using twitter to perform sentiment analysis in Kenya. Ph.D. thesis, University of Nairobi (2011)
8. Hicham, N., Karim, S.: Machine learning applications for consumer behavior prediction. In: The Proceedings of the International Conference on Smart City Applications. pp. 666–675 (2022)
9. Hu, R., Rui, L., Zeng, P., Chen, L., Fan, X.: Text sentiment analysis: A review. In: ICCC. pp. 2283–2288. IEEE (2018)
10. Iftikhar, A., Ghazanfar, M.A., Ayub, M., Mehmood, Z., Maqsood, M.: An improved product recommendation method for collaborative filtering. IEEE Access **8**, 123841–123857 (2020)
11. Izard, C.E.: The psychology of emotions. Springer Science & Business Media (1991)
12. Jin, W., Ho, H.H., Srihari, R.K.: A novel lexicalized hmm-based learning framework for web opinion mining. In: ICML. vol. 10 (2009)
13. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: EACL. pp. 427–431 (2017)
14. Kim, Y.: Convolutional neural networks for sentence classification. In: Conference on Empirical Methods in Natural Language Processing (2014)
15. Krech, D., Petrinovich, L.F., McGaugh, J.L.: Knowing, thinking, and believing : Festschrift for professor david krech (1976)
16. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: AAAI. vol. 29 (2015)
17. Li, W., Jin, B., Quan, Y.: Review of research on text sentiment analysis based on deep learning. OALib (2020)
18. Liu, D., Zhang, H., Yu, H., Zhao, X., Wang, W., Liu, X., Ma, L.: Research on network public opinion analysis and monitor method based on big data technology. ICEIEC pp. 195–199 (2020)

19. Liu, P., Meng, H.: Seemgo: Conditional random fields labeling and maximum entropy classification for aspect based sentiment analysis. In: SemEval. pp. 527–531 (2014)
20. Liu, P., Qiu, X., Huang, X.: Recurrent neural network for text classification with multi-task learning. In: IJCAI. pp. 2873–2879 (2016)
21. Liu, S., Li, F., Li, F., Cheng, X., Shen, H.: Adaptive co-training svm for sentiment classification on tweets. In: CIKM. pp. 2079–2088 (2013)
22. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS. pp. 3111–3119 (2013)
23. Mnih, A., Hinton, G.E.: A scalable hierarchical distributed language model. In: NIPS (2008)
24. Nandwani, P., Verma, R.: A review on sentiment analysis and emotion detection from text. Social Network Analysis and Mining **11**(1), 81 (2021)
25. Plutchik, R.: Emotions and life: Perspectives from psychology, biology, and evolution. American Psychological Association (2003)
26. Raffel, C., Ellis, D.P.: Feed-forward networks with attention can solve some long-term memory problems. arXiv preprint arXiv:1512.08756 (2015)
27. Tang, D., Wei, F., Qin, B., Liu, T., Zhou, M.: Coooolll: A deep learning system for twitter sentiment classification. In: SemEval. pp. 208–212 (2014)
28. Turner, J.H.: Human emotions. Blessings for the Long Night (2022)
29. Vaswani, A., Shazeer, N.M., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. ArXiv **abs/1706.03762** (2017)
30. Wankhade, M., Rao, A.C.S., Kulkarni, C.: A survey on sentiment analysis methods, applications, and challenges. Artificial Intelligence Review **55**(7), 5731–5780 (2022)
31. Xu, D., Tian, Z., Lai, R., Kong, X., Tan, Z., Shi, W.: Deep learning based emotion analysis of microblog texts. Information Fusion **64**, 1–11 (2020)
32. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: Hierarchical attention networks for document classification. In: ACL. pp. 1480–1489 (2016)
33. Zhang, J., He, R., Guo, F.: Quantum-inspired representation for long-tail senses of word sense disambiguation. In: AAAI. pp. 13949–13957 (2023)
34. Zhang, J., He, R., Guo, F., Liu, C.: Quantum interference model for semantic biases of glosses in word sense disambiguation. In: AAAI. pp. 19551–19559 (2024)
35. Zhang, J., He, R., Guo, F., Ma, J., Xiao, M.: Disentangled representation for long-tail senses of word sense disambiguation. In: CIKM. pp. 2569–2579 (2022)
36. Zhang, J., Hou, Y., Li, Z., Zhang, L., Chen, X.: Strong statistical correlation revealed by quantum entanglement for supervised learning. In: ECAI. vol. 325, pp. 1650–1657 (2020)
37. Zhou, Z., Zhou, X., Qian, L.: Online public opinion analysis on infrastructure megaprojects: Toward an analytical framework. Journal of Management in Engineering **37**, 04020105 (2021)