Review Sentiment Orientation Analysis based on Deep Learning

To cite this article: Yixuan Liu and Hailing Xiong 2019 J. Phys.: Conf. Ser. 1267 012009

View the article online for updates and enhancements.
Review Sentiment Orientation Analysis based on Deep Learning

Yixuan Liu¹ and Hailing Xiong²,*

¹College of Computer & Information Science, Southwest University, 400715, China
²Business College of Southwest University, 402460, China

* xionghl@swu.edu.cn

Abstract. The increase of network users has led to a large number of commentary languages on various network platforms. Traditional manual processing is time-consuming and labor-intensive. We need a mechanized way to process these commentary corpora and quickly uncover the emotional tendencies. A method of sentimental orientation analysis of comment text based on deep learning is proposed. First, we used GloVe model to train the word vector. Then, give the different weight on word vector by using TF-IDF. Finally, the processed word vectors would be classified by TextCNN. Experiments were carried out on the six categories of commodity review data crawled by Jingdong. This method can effectively identify the emotional tendency of the review text, which is more accurate than the traditional deep learning method.

1. Introduction

According to China Internet Network Information Center shows that the number of Internet users in China is close to 732 million. Most of them choose to surf the Internet on platforms such as TaoBao, JingDong, Weibo etc. and it will generate a lot of natural language. How to quickly and accurately capture these comments and discover the emotional tendency behind them is of great commercial value. The sentiment orientation analysis mainly focuses on the subjective commentary text and judges its emotional orientation, namely positive, negative and neutral [1]. However, natural language is an advanced human language. Unlike audio and image data, when a computer processes natural language, it needs to convert natural language text into computer-readable data. Here we use the method of word vector to represent the text vocabulary of natural language. Solving the problem of representation of natural language, we can then use the deep learning method to analyze the sentiment orientation.

2. Related work

Sentiment orientation analysis is an important sub-task of sentiment analysis[2]. Emotional analysis was first proposed by Nasukawa T et al. [3]. They proposed that sentiment analysis is not simply to divide the whole document into positive or negative, but rather extract specific emotions related to the subject of the document from the document. There are many classification methods for sentiment analysis. Liu B[4] systematically introduced various aspects of sentiment analysis in a book published in 2012. According to the research method of sentiment analysis, it can be divided into supervised and unsupervised methods.

A supervised approach is to manually label the training corpus and then use a machine learning...
model. Bermingham A et al. [5] used traditional machine learning methods to classify emotions in Twitter. This supervised method can achieve extremely high precision, and its disadvantages are also obvious. This method relies heavily on the quality and quantity of training data, but it is very difficult to obtain a large amount of high-quality corpus. Moreover, the portability of this method is poor, network new words and network buzzwords are constantly being updated, and the accuracy of supervised sentiment analysis methods will be significantly reduced. In another method, the unsupervised method generally performs emotional analysis by analyzing the data statistically and calculating the emotional distribution in the corpus by probability. Lin et al. [6] added a sentiment layer based on the LDA (Latent Dirichlet Allocation) model, and became a JST model with a 4-layer structure to obtain the emotional orientation of each topic. Taking into account the respective advantages of supervised and unsupervised methods, Maas A L et al. [7] combine these two methods to obtain emotionally significant word vectors.

However, both supervised and unsupervised methods require artificially designed features, and the design of the features requires expert domain knowledge, which means a lot of labor costs. With the continuous development of deep learning, scholars have used deep learning in the field of natural language processing. The core of deep learning is that it does not require artificially designed features but is learned through training data. Deep learning has produced very gratifying results in the tasks of natural language understanding, especially subject classification, sentiment analysis, automatic question answering and language translation [8]. Collobert R [9] proposed a unified neural network architecture and corresponding learning algorithms that can be applied to perform various natural language processing tasks. Taking into account the context information in the word Kim Y [10] tried to use CNN for sentence modeling to solve the emotional classification task. After the experiment, it also achieved good results. On this basis, Socher R et al. [11, 12] used RNN, MRNN, RNTN and other recurrent neural networks to process natural language, and considered the syntactic structure of the text into the neural network model, in sentence and The phrase level has achieved good results. There are timing problems in the processing of sentences. Tai K S et al. [13] used LSTM to solve the timing problems of sentences for sentiment analysis. The input of the above model is a word vector. The word vector is the most basic result of deep learning applied in the field of natural language processing, but the traditional word vector only contains semantic and grammatical information and can’t solve the emotional problem of context. Some scholars try to add emotional information in the research process to get the corresponding emotional word vector. Tang D et al. [14] established a three-layer online learning sentiment word vector SSWE to learn from a large number of weakly supervised tweets without a lot of manual annotation. Man Lan et al. [15] proposed three models based on CNN to learn the emotional word vector (SWV), which integrates emotional information and semantic and syntactic information into word representations in three different strategies. In this paper, aiming at the word’s different sentiment weight in a sentence, we creatively added weighted word vectors to get greater sentiment classification.

3. Weighted sentiment word embedding

3.1 GloVe word embedding model

With the development of deep learning, many scholars also apply deep learning, neural networks and other methods to the field of natural language processing. Since neural networks only accept numerical input types, we must use the natural language text we use. Process it and convert it to a numeric type. The word vector will appear. The emergence of word vectors is also a fundamental task for deep learning in recent years. The word vector model used in text is Stanford's GloVe (Global Vectors for Word Representation) [16]. It is very similar to our commonly used Word2vec idea, but GloVe fully considers the co-occurrence of words and is more able to distinguish the meaning of words. And it combines the advantages of LSA, CBOW, training faster, and the scalability of large-scale corpus algorithms is also very good. Performance is also good on small corpora or small vectors. The evaluation of the word vector is evaluated using the similarity of words.
According to Table 1, the word similarity effect of the Glove training word vector is still relatively good.

3.2. TF-IDF weighted word embedding

3.2.1 Introduce of TF-IDF. TF-IDF (term frequency–inverse document frequency) is a word information weighting technique commonly used for information retrieval (data mining), as shown in the following formula 3.1. Its main idea is that the importance of a word increases proportionally with the number of times it appears in the file, but at the same time it decreases inversely with the frequency it appears in the corpus. Used to assess how important a word is to a document or a corpus.

$$TF - IDF(X) = TF(X) \times IDF(X)$$  \hspace{1cm} (3.1)

The term TF (term frequency) in the formula represents the word frequency, that is, the number of times a given word appears in the file. IDF (inverse document frequency) represents the reverse file frequency, that is, the total number of documents in the corpus divided by the total number of articles containing the phrase plus one, and then processing by log.

After the word segmentation and de-punctuation work on the comment text corpus, the corpus of the word segmentation is trained on the one hand using the GloVe word vector model to train the word vector, and on the other hand, the TF-IDF method is used to train each word weight. Then, the obtained weights are multiplied by the trained word vectors, and finally the word vectors with weight information are obtained. The flow chart is shown in Figure 1.

| Product quality   | Similarity |
|-------------------|------------|
| Quality           | 0.9122     |
| Product           | 0.9014     |
| Commodity         | 0.8932     |
| Good quality      | 0.8415     |
| Texture           | 0.8387     |
| Goods             | 0.7984     |
| service           | 0.7481     |

**Table 1.** The 7 words most similar to "product quality"
4. Sentiment Orientation classification model

Using TextCNN model to emotionally classify the weighted word vectors. The main model process is shown in Figure 2.

The model consists of four layers. The first layer is the word vector input layer. Each word in the text comment corpus is mapped to the word vector space. It is assumed that there are a total of N word vectors, and each is K-dimensional. Then the word vector input layer is an N*K-dimensional image. Here, a sliding window of length H is set, and the entire word vector input matrix is swept by the sliding window to calculate a feature vector C corresponding to H, as shown in the following formula 3.2, where b is an offset.

\[
c = f \left( w \cdot x_i + b \right)
\] (3.2)

The second layer is the convolution layer. Different filters act on the word vector input layer to generate different feature maps. The third layer is the pooling layer. The feature map generated by the convolution layer is taken. The maximum, the advantage of doing this is that it can handle comment corpus of varying lengths. The fourth layer is the softmax layer, which is the output vector of the pooling layer by Equation 3.3 below.

\[
p \left( w | \text{Context} \left( w \right) \right) = \frac{e^{y_{w,i}}}{\sum_{i=1}^{N} e^{y_{w,i}}}
\] (3.3)

Calculate the probability of derogatory and derogatory tendency. The greater the probability, the sentiment tendency of the sentence. In the TextCNN model training, we use the data set for NLPCC2014 evaluation task 2, which contains Chinese and English training corpus. For this article, only Chinese polarity comment corpus is selected as the training. In the TextCNN original text, multiple data sets were used for training. Due to the data acquisition problem, only one polar corpus training set was selected in this paper.

5. Simulation Experiment and Result Analysis

5.1 experiment data and process

The experimental data used in this paper is the commodity review corpus obtained from the Jingdong reptile. A total of six commodities (covering electrical appliances, books, hotels, daily necessities, etc.),
a total of 190,533 comments. Among them, the positive evaluation is a total of 89882: The negative evaluation is 100651: The data of the direct crawling needs to be pre-processed first. 1. First, the text is de-emphasized: remove some meaningless comments and the system defaults. 2. Secondly, mechanical compression and de-emphasis: text review corpus has consecutive repeated sentences, such as: "Good, good, good, good, good, good, good, good, good." This sentence is common. The long corpus of meaning needs to be deleted. 3. Breaking sentence deletion: Delete some short text comment corpus, generally 4-8 international characters are more reasonable lower limit, combined with text comment corpus, the shortest special we set the lower limit to 4. After the pre-processing is finished, the corpus is processed for word segmentation, and when the word segmentation is completed, the punctuation marks are also removed. Here we use the jieba participle. In the word segmentation, we will remove the stop words. The stop word dictionary here is a series of commonly used stop word dictionaries summarized by organizations such as HowNet. Since there are many typos in the online commentary, we use a synonym replacement function of jieba to replace the wrong word expression with the correct word expression, which can improve the usability of our crawler corpus data. In the word segmentation, the user can also customize the dictionary. According to the corpus characteristics of the crawler, we summarize a custom dictionary to improve the accuracy of the word segmentation. After the good words are divided, the words are provided to the GloVe model for training.

5.2. Comparison of experimental results
In this experiment we will use Recall, Precision and F1-Measure as the evaluation indicators of this model.

The comparison experiments here can be compared with word2vec+TextCNN Glove+Bayesian Glove+SVM Glove+TextCNN. Ppos represents the accuracy of positive reviews, Rpos represents the recall rate of positive reviews, Fpos represents the F1 value of positive reviews, Pneg represents the accuracy of negative reviews, Rneg represents the recall rate of negative reviews, Fneg represents the F1 value of negative reviews, Farg A weighted average of the F1 values representing positive and negative comments. The performance comparison of the different methods is shown in Table 2 below.

| methods                  | Ppos | Rpos | Fpos | Pneg | Rneg | Fneg | Farg |
|--------------------------|------|------|------|------|------|------|------|
| Word2Vec+TextCNN         | 0.8814 | 0.8921 | 0.8916 | 0.9048  | 0.8828 | 0.8931 | 0.8880 |
| GloVe+TF-IDF+TextCNN     | 0.9024 | 0.9048 | 0.9034 | 0.9185  | 0.9037 | 0.9167 | 0.9102 |
| GloVe+TextCNN            | 0.8832 | 0.8974 | 0.8917 | 0.9006  | 0.8876 | 0.8954 | 0.8915 |
| Word2Vec+SVM             | 0.8149 | 0.9024 | 0.8735 | 0.8969  | 0.8415 | 0.8977 | 0.8546 |
| Word2Vec+Bayes           | 0.7941 | 0.8014 | 0.8145 | 0.8019  | 0.7965 | 0.7816 | 0.7891 |
| GloVe+Bayes              | 0.7887 | 0.8048 | 0.8164 | 0.8047  | 0.7961 | 0.7841 | 0.7901 |
| GloVe+SVM                | 0.8148 | 0.9028 | 0.8812 | 0.8971  | 0.8414 | 0.8678 | 0.8546 |

From the experimental results, we can see that the method used in this paper has a certain improvement in performance in all aspects compared with other traditional methods, which shows that the weight of the given vector can achieve a certain improvement effect of sentiment analysis accuracy. We can see that the accuracy and recall rate of negative reviews is slightly higher than positive reviews. This may be the reason why negative commentary corpus is more than positive commentary corpus.
6. Conclusions
This paper introduces the weight of word vector by using TF-IDF method, strengthens the weight of sentiment word vector, and makes the emotional word vector play a greater role. The effectiveness of this method is also proved by experimental verification. Here we only interpret the emotional classification of emotions. If we want to conduct further emotional analysis, we can analyze the specific emotional attributes. In this paper, the analysis of emotions only includes two kinds of meanings and derogatory meanings. If we want to consider the comprehensive emotions of the commentary, we should also add neutral emotions. The sentiment analysis is divided into neutral, derogatory and derogatory. In addition, there are many ways to express Chinese, and it is necessary to conduct in-depth research on emotional expressions containing metaphors.

Acknowledgment
This work was partially supported by “National Natural Science Foundation of China under Grant (No. 41271292)".

References
[1] K. Schouten and F. Frasincar, Survey on Aspect-Level Sentiment Analysis[M]. 2016.
[2] J. Sheng-Yi, G. Lin-Dong, W. Lian-Xi, and F. Si-Hui, "Survey on Opinion Target Extraction[J]," ACTA AUTOMATICA SINICA, vol. v.44, no. 07, pp. 15-32, 2018.
[3] T. Nasukawa and J. Yi, "Sentiment analysis: Capturing favorability using natural language processing[C]," in International Conference on Knowledge Capture, 2003.
[4] B. Liu, Sentiment Analysis and Opinion Mining[J]. 2012.
[5] A. Bermingham and A. F. Smeaton, "Classifying sentiment in microblogs: Is brevity an advantage?[J]," 2010.
[6] C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis[C]," presented at the Proceedings of the 18th ACM conference on Information and knowledge management, Hong Kong, China, 2009.
[7] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis[C]," presented at the Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1, Portland, Oregon, 2011.
[8] Y. LeCun, Y. Bengio, and G. J. N. Hinton, "Deep learning[J]," vol. 521, no. 7553, p. 436, 2015.
[9] R. Collobert et al., "Natural Language Processing (Almost) from Scratch %J J. Mach. Learn. Res[J]," vol. 12, pp. 2493-2537, 2011.
[10] K. Y, "Convolutional Neural Networks for Sentence Classification[J]," arXiv preprint arXiv:1408.5882, 2015.
[11] R. Socher, C. Manning, and A. Y Ng, Learning Continuous Phrase Representations and Syntactic Parsing with Recursive Neural Networks[J]. 2010.
[12] S. R. P. A, W. J. Y, and e. al, "Recursive deep models for semantic compositionality over a sentiment treebank[C]," presented at the Empirical Methods in Natural Language Processing, 2013.
[13] S. T. Kai, R. Socher, and C. D. J. C. S. Manning, "Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks[J]," vol. 5, no. 1, p. 36., 2015.
[14] D. Tang, F. Wei, N. Yang, M. Zhou, T. Liu, and B. Qin, Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification[J]. 2014, pp. 1555-1565.
[15] L. Man, Z. Zhang, L. Yue, and W. Ju, "Three Convolutional Neural Network-based models for learning Sentiment Word Vectors towards sentiment analysis[C]," in International Joint Conference on Neural Networks, 2016.
[16] P. J. S. R, and M. C, "Glove: Global Vectors for Word Representation[C]," presented at the Empirical Methods in Natural Language Processing, 2014.