Sentiment Analysis of Reviews Based on Deep Learning Model

Qi Wang
Key Laboratory of Convergent Media and Intelligent Technology (Communication University of China), Ministry of Education
School of Computer Science and Cybersecurity,
Communication University of China
Beijing, China
Simonwang1982@sina.com

Lei Sun
Key Laboratory of Convergent Media and Intelligent Technology (Communication University of China), Ministry of Education
Collaborative Innovation Center, Communication
University of China
Beijing, China
sunl1996@126.com

Zheng Chen
Collaborative Innovation Center, Communication
University of China
Beijing, China

***Abstract*—As known as opinion mining, sentiment analysis is a work by using the "natural language processing" method to find out the author's attitude, emotion or evaluation on certain topics. This paper using a dataset by Mass et al from its original Stanford AI Repository, and a commonly pre-processing method——word embedding, and establish a deep learning model for sentiment analysis. From the perspective of data analysis, learn about the movie preferences and cultural characteristics of audiences from domestic and foreign. In the experiment, we compared the performance of RNN, LSTM and GRU in natural language processing, and improve the efficiency and accuracy of sentiment analysis by a fusion model which integrating recurrent neural networks variant at the output of convolutional neural network.**

Keywords—sentiment analysis; natural language processing; deep learning; neural network

I. Introduction

Along with the rapid development of the Internet information age, there is now a huge amount of text data on the Internet. If you can analyze the information from these text data without fixed structure, it will promote the development of a series of applications such as automatic analysis decision-making, network public opinion analysis, emergency warning, and commodity sales. It has great scientific value and Practical application prospects.

The traditional methods used in the past research to solve the problem of text sentiment analysis can achieve certain effects when the amount of data is limited or the semantics of the text are relatively simple. However, with the development of the Internet, the amount of data collected is getting larger and larger, and the expression of language on the Internet is more and more abundant. It is difficult to solve text analysis problems quickly and effectively using traditional methods. Scholars are constantly pursuing new technologies and methods. With the advancement of science and technology, the academic community has begun to pay close attention to the promising approach to deep learning.

In 2006, Geoffrey Hinton, a well-known professor of machine learning artificial intelligence, known as the "father of neural networks", pointed out the following concepts: First, the neural network structure with multiple hidden layers is characterized by excellent performance. The learned features are more representative and more suitable for processing visualization. And the problem of classification; second, the difficulty of training deep neural networks before and after is difficult to solve by initializing each layer[1]. In 2010, scholar Tomas Mikolov published the paper "Efficient Estimation of Word Representations in Vector Space", and then open sourced a tool for calculating word vectors, Word2vec, which converts words into space vectors, and then he built them. A language model based on cyclic neural networks[2]. In 2013, Socher proposed the RNTN (Recursive Neural Tensor Network) model, which introduces the concept of tensor, which can reduce the number of parameters of the model and improve the performance of the model. Due to gradient disappearance or gradient explosion, traditional RNN is difficult to solve the impact of long-term dependence. Subsequently, Socher improved on the basis of RNN, and then proposed the LSTM text sentiment classification model[3]. LSTM solved this problem through the gate mechanism. The problem is therefore more in line with the requirements of text processing[4].

II. Text Preprocessing

When dealing with this work, the first issue is how to input natural language into the model. In natural language processing, words are the most basic part, text is constituted by the words. So, to deal with the problem of NLP, we must take the words first. The word vector provides a numerical representation of text data, which is the basis for text data to be processed by computers. Nowadays, word vectors have been widely used in NLP tasks. Researchers have also proposed a number of models for generating word vectors and developed them into practical tools for everyone to use, such as Word2Vec. Soon after the word2vec model was proposed, Jeffrey Pennington et al. thought that although the skip-gram model is excellent in calculating synonyms, they only train the model in the local context window, and it rarely uses some statistical information in the corpus, so Jeffrey Pennington et al. proposed a new model GloVe. The idea of GloVe is similar to Word2Vec, for Global Vectors, the global corpus statistics are captured directly by the model. It utilizes the main benefit of count data while simultaneously capturing the meaningful linear substructures prevalent in recent log-bilinear prediction-based methods[5]. In this paper, using GloVe per-trained word vectors as weights on embedding layer can improve the effect of sentiment analysis.

III. NEURAL NETWORKS MODELS

A. *Recurrent neural networks*

Recurrent neural networks are a superset of feedforward neural networks. The RNN model can be used to process sequence data. Many works showed that recurrent neural networks can be successfully used in a number of tasks in natural language processing (NLP). The figure 1 below shows the structure of the RNN more clearly.

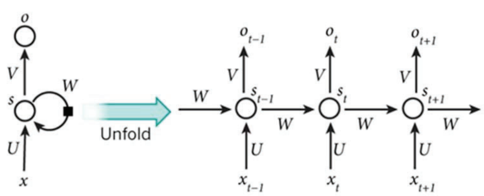

Fig. 1. The structure of the RNN

In the figure1, x_t is the input at time t, o_t is the output at time t, S_t is the memory at time t, U is the weight matrix from the input layer to the hidden layer, and V is the weight matrix from the hidden layer to the output layer. By using (1) and (2), the calculation method of RNN can be represented.

$$o_t = g(v \cdot S_t) \tag{1}$$

$$s_t = f(U \cdot x_t + w \cdot S_{t-1}) \tag{2}$$

B. *Long Short-Term Memory*

RNN contains a large number of parameters and is difficult to train, so a series of RNN optimizations, such as network structure, solution algorithm and parallelization, appear. Due to gradient disappearance or gradient explosion, traditional RNN is difficult to solve the impact of long-term dependence. Subsequently, Socher improved on the basis of RNN, and then proposed the long short-term memory network. Generally called LSTM—is a variant of RNN that can learn long-term dependencies. LSTM has designed input gate, forget gate and output gate to process long-term information and current information to maintain long-term dependency information. The standard LSTM is defined by equation (3)-equation (8). $\odot$ denotes the element-wise product and x denotes the input to the layer, while h denotes the output. In many cases, LSTM has achieved considerable success and has been widely used.

$$y_t = tanh\left(W_y x_t + R_y h_{t-1} + b_y\right) \tag{3}$$

$$i_t = \sigma(W_i x_t + R_i h_{t-1} + b_i + \omega_i \odot c_{t-1}) \tag{4}$$

$$f_t = \sigma\left(W_f x_t + R_f h_{t-1} + b_f + \omega_f \odot c_{t-1}\right) \tag{5}$$

$$c_t = i_t \odot y_t + f_t \odot c_{t-1} \tag{6}$$

$$o_t = \sigma(W_o x_t + R_o h_{t-1} + b_o + \omega_o \odot c_t) \tag{7}$$

$$h_t = o_t \odot \tanh(c_t) \tag{8}$$

C. *Gated Recurrent Unit*

The gated recurrent unit was proposed in [6] as a simpler alternative to the LSTM. The highway network has gates to ensure the unobstructed information flow across the depth of the network. Compared with LSTM, GRU has only two gates which are shown in the figure 2 (reset gate r and update gate z) and does not have the memory cell. Equations (9)-(12) are the detailed equations for r, z, h and $\tilde{h}$.

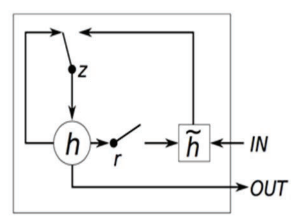

Fig. 2. An illustration of the GRU

$$z_t = \sigma(W_z x_t + R_z h_{t-1} + b_z) \tag{9}$$

$$r_t = \sigma(W_r x_t + R_r h_{t-1} + b_r) \tag{10}$$

$$y_t = tanh(W_h x_t + R_h(h_{t-1} \odot r_t) + b_h) \tag{11}$$

$$h_t = z_t \odot y_t + (1 - z_t) \odot h_{t-1} \tag{12}$$

IV. EXPERIMENT-BASED RESULTS

A. Dataset

Large Movie Review Dataset by Mass et al from its original Stanford AI Repository contains 25k highly polar movie reviews for training and 25k for testing, each text is labeled with positive or negative. This dataset can be found and downloaded in Kaggle. So, binary sentiment classification use this dataset to compare the effects of different deep learning models and algorithms. In order to verify the classification performance of the model mentioned in this paper, comparative experiments were conducted. The parameters used in the models are as follows in Table 1.

TABLE I. PARAMETERS OF MODELS

Parameter	value
Word vector dimension	100
Loss	Binary_crossentropy
optimizer	adam
filters	100
Kernel_size	3
activation	relu

B. SimpleRNN, LSTM and GRU

From this dataset, we carried out a comparison among the neural network mentioned above. Word vector dimension to be able to get a better performance when the 100-dimensional[7], and the end result is the mean of five experiments with epochs of 5. The sentiment classification results are shown in Table 2.LSTM and GRU both are better than Simple RNN because they can catch Long-Term Dependencies. The observation is that for models with units of 16 or 32, GRU performed slightly better than LSTM . However, the RNNs were found to benefit clearly better from deeper widths than the LSTMs or GRUs. In this comparison, the effect of GRU in this sentiment analysis work is similar to LSTM. And 1-layer GRUs are slightly better than LSTM, for GRU has fewer parameters, it's relatively easy to train and the problem of over-fitting is a little smaller, so it can perform better when training data is less.

TABLE II. ACCURACY OF MODELS

Models		Units		
		16	*32*	*64*
RNN	1	0.8223	0.8394	0.8234
	2	0.8379	0.8480	0.7881
LSTM	1	0.8598	0.8590	0.8564
	2	0.8574	0.8552	0.8577
GRU	1	0.8599	0.8600	0.8450
	2	0.8599	0.8548	0.8533

C. BiLSTM and BiGRU

Both LSTM and GRU can only predict the output of the next moment based on the timing information of the previous moment. However, in some problems, the output of the current moment is not only related to the previous state, but also may be related to the future state. For example, predicting a missing word in a sentence needs not only to judge according to the previous paper, but also to consider the content behind it, and truly based on context. Therefore, proposes bidirectional recurrent neural network models which are combinations of forward network and backward network. The results are shown in Table 3. The observation is that Bidirectional LSTM and Bidirectional GRU are slightly better than LSTM and GRU with a units of 16.

TABLE III. ACCURACY OF MODELS

Models	Units	
	16	*32*
BiLSTM	0.8634	0.8558
BiGRU	0.8620	0.8519

D. Fusion Model of convolution neural network and bidirectional recurrent neural network

In [8], the convolutional LSTM (ConvLSTM) was proposed and used to build an end-to-end trainable model for the precipitation nowcasting problem. The paper presented a neural network of fusion model, which improve the efficiency and accuracy of sentiment analysis. The above experiments show that Bidirectional LSTM and GRU perform better than one-way networks. Convolution neural network has the function of local feature extraction, so CNN can be used to extract key information similar to n-gram in sentences[9]. Thus, by using pre-trained word vectors GloVec and integrating bidirectional recurrent neural networks variant at the output of convolutional neural network, the accuracy of sentiment analysis can be better. The structure of fusion model with Convolution and BiGRU is shown in figure 3.

TABLE IV. ACCURACY OF FUSION MODELS

Models	Units	
	16	*32*
Conv + BiGRU	0.8742	0.8701
Conv + BiLSTM	0.8720	0.8443
GloVe + Conv + BiGRU	0.8878	0.8731
GloVe + Conv + BiLSTM	0.8933	0.8533

All results are shown in Table 4. We can find that the performances of the models with units of 32 are not so good for this task. Also, it can be seen that Conv-BiLSTM with GloVe pre-trained word vectors has the highest accuracy, which is mainly due to two reasons. First, the fusion model use the convolutional neural network to extract the advantages of local features, and use the bidirectional recurrent neural network to take into account the global feature advantages of text sequences. Another reason is that, pre-training of word vectors is an important ingredient in deep learning for NLP, and using word vectors that fits corpus the can provide better optimization than changing the model structure. In this experiment, using the word vectors that trained by dataset of this paper can make the accuracy reach 92 or more.

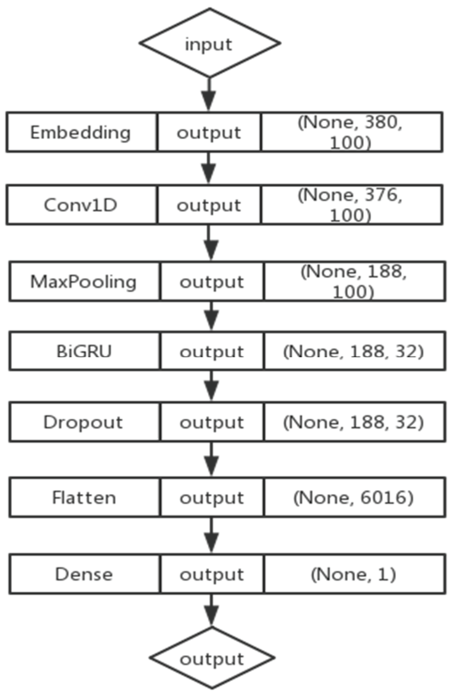

Fig. 3. The fusion model of ConvBiGRU

V. CONCLUSIONS

Through this experiment, we have an understanding of the principles of commonly used neural network algorithms and how to apply them in natural language processing. Through the IMDB natural language processing test to compare the effects of neural networks, the following conclusions are drawn: the use of RNN and its variants in the field of natural language processing has better performance than the traditional perceptron, but the computing power is restricted by hardware. More resources are used for training and running. Therefore, in practical applications, it is necessary to select appropriate models and algorithms based on data size, training cost, etc., while deep learning commonly used neural network algorithms through model fusion, improved algorithms and other methods, also greatly reduce training costs and improve accuracy. potential.

This paper only selects the dataset of the movie reviews for sentiment analysis. In other different text datasets, different fields have special vocabulary and description methods, so the difference in sentiment orientation in different kinds of text data is obvious. Different applications can establish their own corpus in various fields, in this way can sentiment analysis of text achieve better results. This experiment proves that a simple neural network can achieve higher accuracy on the text classification problem. Through the model fusion and parameter optimization, the accuracy can be further improved, which can be seen based on the deep learning neural network. Natural language processing applications also have great application value and research space.

VI. ACKNOWLEDGMENTS

This paper is supported by "the Fundamental Research Funds for the Central Universities" 、"Emotional Mining Analysis of Multilingual Network Movie Review"(NO.3132018XNG1837）and "Regionalized Public Opinion Analysis and Data Report Generation System"(NO.3132018XNG1804).

REFERENCES

[1] Hinton G E , Osindero S , Teh Y W . A Fast Learning Algorithm for Deep Belief Nets[J]. Neural Computation, 2014, 18(7):1527-1554

[2] Mikolov T , Chen K , Corrado G , et al. Efficient Estimation of Word Representations in Vector Space[J]. Computer Science, 2013.

[3] Graves A. Long Short-Term Memory[M]// Supervised Sequence Labelling with Recurrent Neural Networks. 2012.

[4] Tai K S, Socher R, Manning C D. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks[J]. Computer Science, 2015, 5(1):: 36.

[5] Pennington, Jeffrey ; Socher, Richard ; Manning, Christopher D: Glove: Global Vectors for Word Representation.. 14. In: EMNLP, 2014, S. 1532—1543

[6] Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry ¨ Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[7] CHEN Bo, Text classification method based on the cycle structured convolutional neural network [J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition),2018,30(05):705-710

[8] Shi X , Chen Z , Wang H , et al. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting[J]. 2015.

[9] Kalchbrenner N , Grefenstette E , Blunsom P . A Convolutional Neural Network for Modelling Sentences[J]. Eprint Arxiv, 2014, 1.