

# Weibo Comments Sentiment Analysis Based on Deep Learning Model

Xixiang Hu\*, Yu Zhang and Hongli Zhang

School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

\*Corresponding author

**Abstract**—In this paper, the sentiment analysis based on the deep learning model was studied. By comparing the effects of the shallow learning model SVM and the deep learning model LSTM on the classification of Weibo comments, we found that the classification result of the LSTM model is better than that of the SVM model under the same group of training set and test set. One of the main reasons lies in that the LSTM model exploits the information of word order during training. Firstly, we need to crawl the original Weibo data from Website and pre-process these data. Secondly, with the help of the neural network language model inside word2vec, the word embedding as the input of the SVM model and the LSTM model are trained. Thirdly, inputting the training set and test set constructed by the word embedding into the SVM model and the LSTM model respectively, then we got the experimental results. Finally, by comparing the experimental results of the SVM model and the LSTM model, we found a way to improve the accuracy of sentiment classification by introducing the information of word order in the model.

**Keywords**—sentiment analysis; word2vec; SVM; LSTM

## I. INTRODUCTION

With the development of the worldwide Internet, Weibo is becoming the main platform for Internet Citizens to express their emotions and opinions. By 2015, the number of netizens have reached 669 million, while the real registered users of Weibo have reached about 400 million and monthly active users have reached 212 million. The social network of Weibo can largely reflect Chinese people's opinions and sentiments. So it is imperative for the government and enterprises to make a timely decision if they can know those opinions and sentiments. Now, sentiment analysis technology has been widely used in the field of opinion tracking, product evaluation and so on. However, in the era of big data, a large amount of data is produced on the Internet every day and if we use the traditional methods to analyze these data will cost a lot of manpower and material resource. So in this paper, we will use a deep learning model to deal with these data. Under the deep learning model, features can be extracted easily and the accuracy in sentiment analysis can be improved as well.

## II. RESEARCH METHODS

### A. Research Process

In this paper, the whole research consists of three stages.

#### 1) Data preprocessing:

Selecting the appropriate data from Weibo and saving it locally. Removing the dirty data that will affect the results of the sentiment analysis and keeping comments with length greater than 5, since most of these comments contain the information of word order. After the data cleaning, the remaining comments need to be marked with the sentiment. Then the comments need to be expressed as word embeddings that computer can recognize and this part of work done by word2vec. The output of word2vec is used as the input of the SVM model and the LSTM model. Yoav Goldberg and R Pieraccini explained the internal calculation process in neural network language model[1,2], furthermore, with the development of deep learning, more and more deep learning neural networks are being used to produce word embeddings. Sundermeyer introduced a method that using language model based on LSTM neural network to produce word embeddings [3]. Mikolov and others introduced a method that using recurrent neural network language model to produce word embeddings[4,5]. Training the word embeddings is main the work in this stage and this work will take a long time.

#### 2) Structing the models:

In this stage, the SVM is used to compare with the LSTM model. The first thing is confirming the parameters of the SVM and the LSTM, then input the experimental data to the SVM model and the LSTM model respectively. Then the experimental results can be obtained. Liang introduced a method to classify the web text by the SVM model[6] and Fu introduced a method to classify the indistinguishable Chinese official documents by the SVM model[7]. In this paper, the SVM model is used for binary classification task. Then the LSTM model is used to carry out the same experiment with the same data and get the results. Li introduced a method to deal with speech recognition task with the LSTM model[8].

#### 3) Analyzing the results:

In this stage, by comparing the experimental results from the SVM and the LSTM, the final conclusion is that the information word order can improve the accuracy in sentiment classification. The specific research process is depicted in Figure 1.

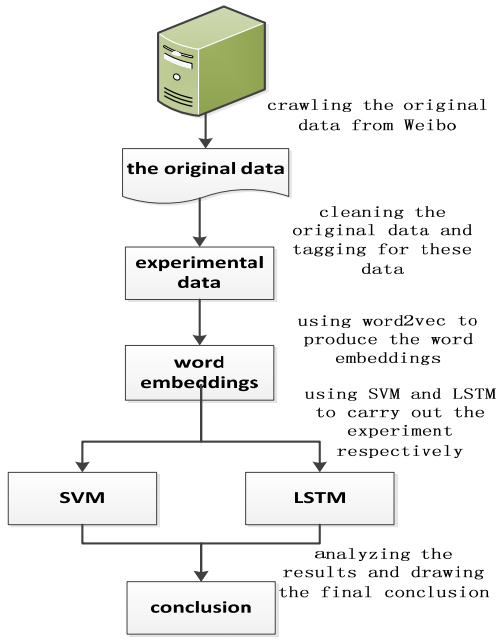


FIGURE 1. THE RESEARCH PROCESS DIAGRAM

#### B. Neural Network Language Model Word2vec

In the traditional machine learning models, one-hot representation is often used as the input of the models, it is a long vector that is used to represent a word. The length of this vector equals to the size of dictionary, in this vector, only one location is 1 and other locations are 0. The location of 1 corresponds to the location of the word in dictionary. This representation is simple, however, there are two drawbacks in this representation:

- It is vulnerable to the dimension disaster, especially for algorithm in the deep learning.
- It can't describe the similarity between the words and is easy to cause lexical gap.

Word2vec is a group of related models that are used to produce the word embeddings, which was pushed-off by Google in 2013. There are two neural network language models in Word2vec, Continuous Bag-of-Words (CBOW) and Skip-Gram. They are improved models based on the neural network language model. In the two models, the hidden layer which has a large amount of computation in the neural network language model is removed, however, the projection layer is introduced. The function of this projection layer is to sum the vectors from input layer. The main difference between CBOW and Skip-Gram is that CBOW is the model that according to the context of  $w$  to solve the probability that  $w$  appears. The learning target of CBOW is to maximize the log likelihood function, as in (1).

$$L = \sum_{w \in C} \log p(w | \text{Context}(w)) \quad (1)$$

Skip-Gram is the model according to the  $w$  to solve the probability that the context of  $w$  appears. The learning target of Skip-Gram is to maximize the function, as in (2).

$$p(\text{Context}(w) | w) = \prod_{u \in \text{Context}(w)} p(u | w) \quad (2)$$

### III. EXPERIMENTAL MODELS

#### A. Machine Learning Model SVM

Support Vector Machine (SVM), is a machine learning model that is used in statistics and classification tasks widely. The SVM is a general liner classifier which is characterized by the ability to minimize the empirical error and maximize the geometric marginal zone, so the SVM is also known as the maximum margin classifier. The SVM also offers a variety of kernel functions that can map vectors to a higher dimensional vector space, there is a maximum interval hyperplane in this space. There are two parallel hyperplanes on both sides of the maximum interval hyperplane, the maximum interval hyperplane maximizes the distance between two parallel hyperplanes. It assumes that the larger distance between two parallel hyperplanes is, the smaller total error of the classifier, therefore, the objective function is (3).

$$\min \frac{1}{2} \|w\|^2 \quad s.t., y_i (w^T x_i + b) \geq 1, i = 1, \dots, n \quad (3)$$

This is a Lagrange optimization problem, we can obtain the weight matrix  $W$  and bias  $b$  of the optimal hyperplane by the Lagrange multiplier approach. Then we can get (4).

$$L(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^n \alpha_i (y_i (w^T x_i + b) - 1) \quad (4)$$

The Gauss kernel function is adopted in the experiment, the Gauss kernel function is also called Radial Basis Function (RBF). RBF is a scalar function along the radial symmetry, usually defined as a monotonic function that represents the Euclidean distance between an arbitrary point  $x$  and the center  $z$  in this space. The function is (5).

$$K(\|x - z\|) = \exp\left(-\frac{\|x - z\|^2}{2\sigma^2}\right) \quad (5)$$

If the distance between  $x$  and  $z$  is very close, the value of the kernel function is 1, conversely, the value of the kernel function is 0.  $\sigma$  is a parameter which controls the radial range of this function.

#### B. Deep Learning Model LSTM

Long Short Term Memory Network (LSTM) is an improved recurrent neural network. It successfully overcomes the defects of the original recurrent neural network, which has become the most popular recurrent neural network and achieved good results, in speech recognition, NLP and other fields. LSTM has the function of filtering information through a carefully designed structure called gate. The structure of gate in the LSTM is depicted in Figure 2.

The gate is actually a fully connected layer. The input is a

vector and output is a real number between 0 and 1. Assuming that  $W$  is the weight vector,  $b$  is a bias term, then the gate can be represented as (6).

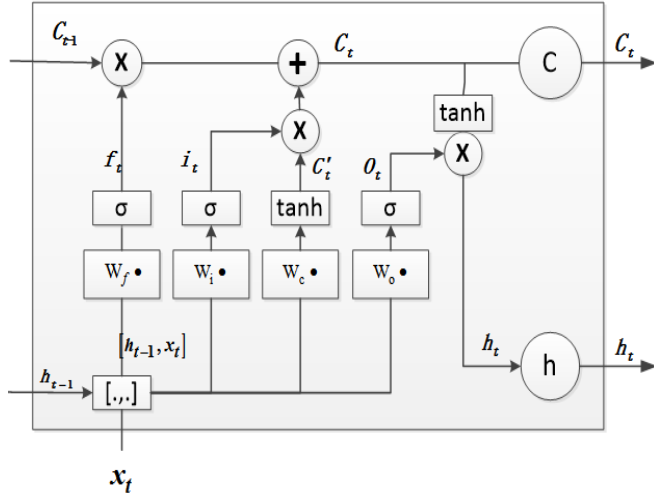


FIGURE II. THE STRUCTURE OF GATE IN LSTM

$$g(x) = \sigma(Wx + b) \quad (6)$$

The  $\sigma$  is Sigmoid function, the LSTM uses two gates to control the content of the element state  $c$ . One is the forget gate which determines how much information can be reserved from unit  $c_{t-1}$  at the last moment to unit  $c_t$  at this time. The other is the input gate which determines how much input information  $x_t$  can entered the unit  $c_t$ . LSTM uses the output gate to control how much information from unit  $c_t$  can input to the current output  $h_t$ . So the function of forget gate is (7).

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (7)$$

$W_f$  is the weight matrix of the forget gate,  $[h_{t-1}, x_t]$  means that the two vectors are spliced into a longer vector,  $b_f$  is the bias term of the forget gate,  $\sigma$  is Sigmoid. If the dimension of input is  $d_x$ , the dimension of hidden layer is  $d_h$ , the dimension of unit state is  $d_c$ , so the dimension  $W_f$  is  $d_c \times (d_h + d_x)$ . Actually,  $W_f$  is made up of two matrices, the one is  $W_{fh}$  which corresponds to the output term  $h_{t-1}$  and its dimension is  $d_c \times d_h$ , the other is  $W_{fx}$  which corresponds to input term  $x_t$  and its dimension is  $d_c \times d_x$ , therefore,  $W_f$  satisfies the equation in (8).

$$[W_f] \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} = [W_{fh} \quad W_{fx}] \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} = W_{fh} h_{t-1} + W_{fx} x_t \quad (8)$$

The input gate is the function in (9).

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (9)$$

$W_i$  is weight matrix of the input gate,  $b_i$  is the bias term of the input gate. The current input unit state is  $\tilde{c}_t$ , which is calculated based on the previous output and the input, as in (10).

$$\tilde{c}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (10)$$

The unit state at current time,  $c_t$  which is generated by two products, one is the last unit state  $c_{t-1}$  multiplies the forget gate  $f_t$ , the other is  $\tilde{c}_t$  multiplies  $i_t$ , so the  $c_t$  is a function as in (11).

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t \quad (11)$$

In this way, LSTM combines the current memory  $\tilde{c}_t$  and the long-term memory  $c_{t-1}$  to form a new unit state  $c_t$ . Thus, owing to the forget gate, LSTM can store the information which is a long time ago. Due to the input gate, LSTM can avoid the irrelevant content into current memory. The output gate controls the effect of long-term memory on the current output and its function is (12).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (12)$$

The final output of LSTM is determined by the output gate and the unit state. Its function is (13).

$$h_t = o_t \cdot \tanh(c_t) \quad (13)$$

#### IV. EXPERIMENT

##### A. Data Preprocessing

Firstly, crawling the comments from Weibo in 2016, after data cleaning, about 15000 comments are left and the ratio of the training set and the test set is 4:1. Data cleaning is to remove duplicate comments and comments that are less than 5 characters. The duplicate comments will have an impact on the experimental results and in the comments that less than 5 characters, many comments don't include the information of word order, so these data doesn't meet the requirement of this experiment. In order to improve the accuracy of the word embeddings, adding the news corpus, which size is 2G and is provided by Sougou laboratory, to word2vec to train with comments together. Finally, word2vec outputs 4 kinds of word embeddings, their dimensions are 50, 100, 150 and 200.

##### B. Experiment Results Using SVM

In order to get the optimal parameters of SVM, we carried out the experiment of the cross validation. The best C is 8, the best gamma is 0.00048828125. All the experiments are provided with the same optimal parameters. In this paper, we

have done the experiments on the four kinds of dimensions. The experimental results of the SVM model are shown in the Table I. The three indicators are used to evaluate the SVM model and the LSTM model, they are precision, recall and Figure1.

According to the experimental results shown in table I, the precision of positive comments is high, which the recall rate of positive comments is low. However, for the negative comments, the precision is low and recall rate is high. According to the confusion matrix, we can know that a lot of positive comments are classified as negative comments falsely. From the values of F1 we can know that SVM has a better effect on negative comments than positive comments, in other words, the SVM model is sensitive to negative comments, it is more likely to classify the comments that are blurred into negative comments.

### C. Experiment Results Using LSTM

Firstly, setting the hyper-parameters for the LSTM networks, in the input layer, inputting the word embeddings respectively. Their dimensions are 50, 100, 150 and 200. The hidden layer is LSTM network. The output layer is a fully connected layer, in this layer, the Sigmoid function is the activation function. In the whole deep learning neural network, the cross entropy loss function is used as the loss function and the number of iterations is set to 3 times. The experimental results of the LSTM model are shown in Table II.

According to the experimental results, we can know that the LSTM model can improve the values of precision, recall, distinctly, and Figure1 compared with the SVM model, that is to say the LSTM model is better than the SVM model in the field of sentiment analysis.

TABLE I. THE EXPERIMENTAL RESULTS OF SVM

Dimension of word embeddings	POS_P	POS_R	POS_F1	NEG_P	NEG_R	NEG_F1
50	0.84	0.703	0.765	0.745	0.866	0.801
100	0.846	0.711	0.773	0.751	0.871	0.807
150	0.838	0.707	0.767	0.746	0.863	0.801
200	0.851	0.711	0.775	0.752	0.876	0.809

TABLE II. THE EXPERIMENTAL RESULTS OF LSTM

Dimension of word embeddings	POS_P	POS_R	POS_F1	NEG_P	NEG_R	NEG_F1
50	0.878	0.741	0.804	0.776	0.897	0.832
100	0.897	0.745	0.814	0.782	0.915	0.843
150	0.868	0.761	0.811	0.787	0.885	0.833
200	0.897	0.739	0.81	0.778	0.915	0.841

### D. The Accuracies of SVM and LSTM

The accuracies of SVM and LSTM are depicted in Figure 3. From the Figure 3, we can know that the highest accuracy of LSTM model is 83%, however, the highest accuracy of SVM model is 79.33%. So we can draw the final conclusion: by comparing with the SVM model, the LSTM model can improve the accuracy of classification, one of the main reasons lies in that the LSTM model reserves the information of word order in computing process.

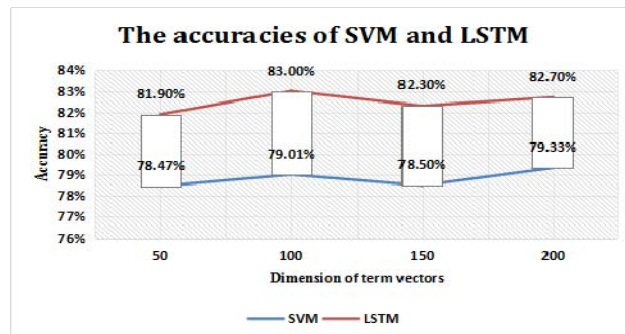


FIGURE III. THE ACCURACIES OF SVM AND LSTM

### V. CONCLUSION

In this paper, the highest accuracy of the LSTM model is 3.67% higher than that of the SVM model. According to the results, the main difference between the SVM model and the LSTM model is that the SVM model doesn't utilize the information of word order during training, however, the LSTM model retains that information during training. So, we can draw the conclusion that in order to improve the accuracy of sentiment classification, it is necessary to preserve the information of word order. In this paper, we use 12,000 Weibo comments as training set and 3,000 Weibo comments as test set, so it can be inferred that there will be a greater gap between the SVM model and the LSTM model under the larger data set, which needs to be validated in future studies. Another advantage of the LSTM model is that it can records the previous information. This processing mode is more in line with the human way of learning. With the increasing amount of data, the advantage of the LSTM model is more obvious. We believe that the LSTM model can be one of the best models in the field of sentiment analysis.

### ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (NSFC, Grant no. 61601146).

### REFERENCES

- [1] Goldberg Y. A Primer on Neural Network Models for Natural Language Processing[J]. Computer Science, 2015.
- [2] Pieraccini R, Levin E. Natural Language Understanding with Distributed Representation[J]. Nato Asi, 2015, 147:139--155.
- [3] Sundermeyer M, Schlüter R, Ney H. LSTM Neural Networks for Language Modeling[C]// Interspeech. 2012:601-608.
- [4] Mikolov T, Yih S W, Zweig G. Linguistic Regularities in Continuous Space Word Representations[C]// 2013:296-301.
- [5] Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model[C]// INTERSPEECH 2010, Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September. DBLP, 2010:1045-1048.
- [6] Liang J Z. SVM multi-classifier and Web document classification[C]// International Conference on Machine Learning and Cybernetics. IEEE, 2004:1347-1351 vol.3.
- [7] Fu J H, Lee S L. A multi-class SVM classification system based on learning methods from indistinguishable chinese official documents[J]. Expert Systems with Applications, 2012, 39(3):3127-3134.
- [8] Li J, Mohamed A, Zweig G, et al. LSTM time and frequency recurrence for automatic speech recognition[C]// Automatic Speech Recognition and Understanding. IEEE, 2015:187-19.