The Journal of Engineering

*The 3rd Asian Conference on Artificial Intelligence Technology (ACAIT 2019)*

IET Journals
The Institution of Engineering and Technology

# Micro-blog sentiment classification using Doc2vec + SVM model with data purification

eISSN 2051-3305
Received on 11th October 2019
Accepted on 19th November 2019
E-First on 6th July 2020
doi: 10.1049/joe.2019.1159
www.ietdl.org

*Yinghong Liang*[1] ✉, *Haitao Liu*[1], *Su Zhang*[2]

[1]*Department of Software Engineering, Jingling Institute of Technology, Nanjing, People's Republic of China*
[2]*Department of Computer Engineering, Suzhou Vocational University, Suzhou, People's Republic of China*
✉ *E-mail: liangyh@jit.edu.cn*

**Abstract:** As a Chinese version of twitter, micro-blog has been popular for many years. On this platform, a lot of comments are generated explosively every day. These comments contain the public's opinions on various topics, which have wide applications in both academic and industrial fields. In recent years, deep learning and some classification algorithms have been applied to sentiment analysis, and good results have been achieved. However, micro-blog sentiment classification is a challenging task, because micro-blog messages are short and noisy, and contain massive user-invented acronyms and informal words. Unfortunately, most researchers pay more attention to analyse the data after deep learning, but only simply remove the noisy data before using algorithm, so the result of sentiment analysis has reached a bottleneck. Here, the authors first purify the data using varied methods before deep learning, then, the Support Vector Machine (SVM) classification algorithm is applied to sentiment classification of micro-blog using many types of features. Through comparing with the method of simply pre-processing data, the results show that their approach can improve the performance of micro-blog sentiment classification effectively and efficiently.

## 1 Introduction

As a hotspot of current research, sentiment analysis has attracted the attention of many researchers. Sentiment analysis generally includes three aspects: sentiment polarity analysis; subjective and objective analysis; sentiment intensity classification. Generally speaking, sentiment classification is the sentiment polarity classification.

Since the work in [1], machine learning methods, especially supervised learning approaches, have become the main-stream sentiment classification method [2, 3]. They are widely used in micro-blog sentiment classification [4–8]. For example, Bermingham and Smeaton [6] applied Support Vector Machine (SVM) and Multinomial Naive Bayes methods to classify the sentiments of tweets. Go *et al.* [8] proposed a distant supervision method, which several manually emoticons were selected to label corresponding micro-blog messages. Hu *et al.* [9] proposed an unsupervised micro-blog sentiment classification approach by incorporating various kinds of sentiment signals. The major advantage of this method is that it does not rely on any manually labelled dataset. However, since emoticons contain heavy noise when used as sentiment labels, the accuracies of this method are limited [9]. Tang *et al.* [10] learned the distributed representations of English n-grams using word embedding techniques. Then they classified these n-grams into positive or negative using SVM and built a tweet-specific sentiment lexicon.

Recently, deep learning approaches emerge as powerful computational models that discover intricate semantic representations of texts automatically from data without feature engineering [11]. Especially, recent advances in deep neural networks demonstrate the effectiveness of representation learning for text [12]. A key advantage of these neural architectures is that they can capture the meaning of linguistic phenomena ranging from individual words to longer-range linguistic contexts at the sentence level.

Doc2vec model uses the idea of deep learning to simplify the processing of text content into vector operations in K-dimensional vector space, which takes into account the order and semantic features of words. Compared with other deep learning methods, it has certain advantages in sentiment analysis. It is worth mentioning that Fei Ye [4] achieved better results by adding network extension terms feature and Wu Dongyin *et al.* [5] combined Gauss Transfer Learning with Deep Learning also improves the accuracy of sentiment analysis. Unfortunately, most researchers pay more attention to analyse the data after deep learning, but only simply remove the noisy data before using algorithm, so the result of sentiment analysis has reached a bottleneck.

In this paper, we purify the data before Doc2vec processing and SVM classification algorithm. Our model can not only deal with noisy corpus adequately, but also take into account the context and semantic features of sentences. More importantly, we can explore whether data purification before using algorithm is effective for emotional analysis.

## 2 Details of Doc2vec model with data purification

Most researchers simply deal with the corpus before deep learning. In our approach, we pay more attention on this stage, a set of useless parts of speech, sentiment dictionaries and stop words dictionaries have been applied to this process. The data purification procedure is shown in Figs. 1 and 2, and the green area is the most researchers' method for data processing, the yellow areas are the additional steps of our approach for getting Doc2Vec and SVM training data.

### 2.1 Text data purification for Doc2Vec training

The words used in micro-blog texts are very colloquial and networked. There are a lot of expressions, invalid information and the latest Internet buzzwords. The process of data purification is to remove useless information and extract effective information for subsequent analysis.

There are two training processes in our method, namely Doc2Vec training and SVM training, respectively. The data purification procedures for these two steps are not the same.

For example: Unprocessed sentence: “三星 n/手机 n/很好 d/用 v/比较 d/人性化 a”

Processed data for Doc2Vec training:“三星 n/手机 n/很好 d/用 v/比较 d/人性化 a”

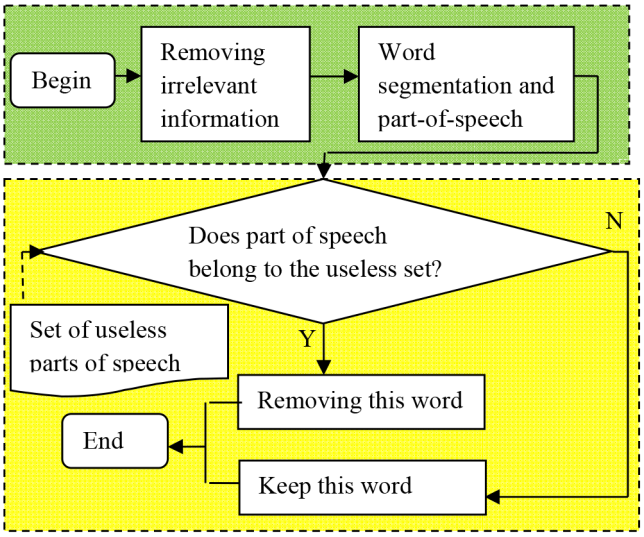


**Fig. 1** *Steps of our approach for getting Doc2Vec training data*

Processed data for SVM training: '手机 n/很好 d/人性化/a'

There are a large number of stop words in everyday language, such as '今天', '我' and other words without any sentiment tendency. These words not only increase the complexity of sentences and processing time, but also may have the opposite effect on affective analysis. So, after segmentation, these stop words must be removed. The steps of our approach for getting SVM training data are shown in Fig. 2.

### *2.1.1 Steps for getting Doc2Vec training data:*

(i) Using regular expressions to remove completely unrelated information in the text, such as @xx, the address beginning with 'http: ' and # XX # topic information.
(ii) Using 'Jieba segmentation' to segment the original text, we can get the segmentation result with part-of-speech.
(iii) If the part of speech belongs to the useless set ('w', 'x', 'un'), it is filtered immediately, and all the remaining words are retained for Doc2Vec training.

*2.1.2 Steps for getting SVM training data:* In the data available for Doc2Vec training, we continue to purify:

(i) If the resulting words are included in the dictionary of stop words, they are filtered directly (look up the dictionary of stop words).
(ii) If the words obtained are included in the sentiment dictionary, they are retained directly (look up the sentiment dictionary).
(iii) If the part of speech belongs to the useful set ('Ag', 'a', 'a d', 'a n', 'd', 'e', 'i', 'j', 'l', 'n', 'o', 'vg', 'v', 'v d', 'v n', 'y', 'z'), it is retained or filtered (to find the useless part of speech set).

The data obtained from the above process are used for the training of SVM.

## 3 Converting sentences into vectors

Doc2Vec is an unsupervised algorithm that can project a sentence, document or paragraph into vector space with rich semantic expression.

There are two main steps in the process of doc2vec:

Step 1: Training model: Word vector, parameter sum of softmax and paragraph vector/sentence vector are obtained from training data.
Step 2: Inference stage: For new paragraph, get its vector expression. Specifically, adding more columns to the matrix, using the above-mentioned methods training data, a gradient descent method was used to get new vector, thus, obtaining the vector expression of the new paragraph.

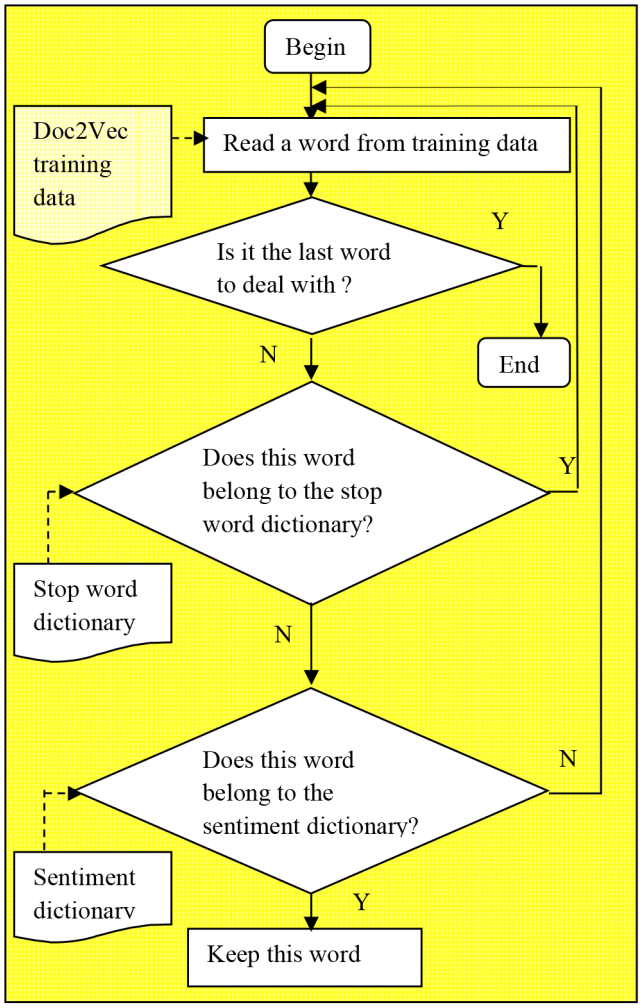


**Fig. 2** *Steps of our approach for getting SVM training data*

We use Doc2Vec which included in Gensim tool kit to transform sentences into vectors. The specific steps are as follows:

(i) Converting sentences to gensim.models.doc2vec. LabeledSentence objects.
(ii) Setting parameters of Doc2Vec training (size = 64, window = 5, min_count = 1, workers = 4, alpha = 0.025, min_alpha = 0.025), select min_count = 1 to contain all the words, so as to prevent the absence of words in the model when SVM is to be used.
(iii) 20 iteration training.
(iv) Save the training model for SVM.

After these four steps, sentences are successfully transformed into vectors, which are ready to the process of SVM classifiers.

## 4 Sentiment classification experiment using SVM

In view of the practical problems proposed in this paper, SVM algorithm is used as classification algorithm. In addition, considering that Nave Bayes algorithm and sentiment dictionary matching method are also commonly used in sentiment analysis, we compare the effects of these three models in sentiment classification of micro-blog.

### *4.1 Use of corpus*

We used COAE2013 and COAE2015 Chinese sentiment analysis corpus, dividing it randomly into training set (80%) and test set (20%).

### 4.2 Use of characteristics

The design of appropriate features has an important impact on the classification effect of the model. By analysing the characteristics of micro-blog text, we use the following three types of features:

(i) Context and Semantic Information: Sentence vector information obtained after Doc2Vec contains a lot of context and semantic knowledge.
(ii) Frequency of Sentiment Words: One of the most important features of sentiment polarity judgment is the use of sentiment words. For example, the words '开心', "兴奋" and "点赞" appear in the micro-blog commentary, which is likely to express the positive polarity; if the words "伤心", "难过" appear, the commentary is likely to express the negative polarity. The method of extracting sentiment words is mainly based on the dictionary of sentiment words. In the actual processing, the sentiment lexicon set provided by HowNet is used as the sentiment dictionary. In addition, according to the characteristics of micro-blog comments, the sentiment dictionary has been expanded, such as the popular words '杠精' and '命运共同体' in 2018.
(iii) Part of speech: After using the 'Jieba segmentation', we get the part of speech of each word. Such parts of speech as 'Ag', 'a', 'a d', 'a n', 'd', 'e', 'i', 'j', 'l', 'n', 'o', 'vg', 'v', 'v d', 'v n', 'y', 'z' are words that have an effect on sentiment analysis, and ultimately retain them.

### 4.3 Sentiment classification model and process

We use SVM implementation package module in Scikit-learn. Statistical analysis of all sentences shows that the length of sentences is basically concentrated in 35 words, so 35 is chosen as the vector length.

We use three characteristics in this paper:

The first one is the vector information containing context and semantics which is processed by Doc2Vec.
The second one is the word frequency of sentiment words and vector information.
The third one is the vector information whether the query is obtained from useful parts of speech.

The above three kinds of vector information are fused and input into SVM for sentiment classification. The data fusion algorithm is shown in Fig. 3:

The sentiment classification model is shown in Fig. 4.

The steps of sentiment classification using SVM are as follows:

(i) Using Doc2vec model to convert all sentences into vectors.
(ii) If the length of a sentence is > 35, make up with 0; if the length of a sentence is < 35, select the first 35 words to use.
(iii) Adjusting parameters. GridSearchCV method is used to adjust the kernel function, C (penalty parameter) and gamma value.

The values of kernel function are {linear kernel function, polynomial kernel function, Gauss kernel function}, C is np.linspace (0.5, 1.4, 10), and gamma is 1/(np.linspace (3, 4.8, 10)*1000

The test set is tested using the adjusted SVM model (Kernel = Gauss Kernel, Gamma = 1/3000, C = 1.4). Accuracy, recall rate and F1 value were obtained.

## 5 Results and analysis

### 5.1 Baseline methods

In order to verify the validity of Doc2vec + SVM model, we use Nave Bayes and sentiment dictionary matching method to recognise emotions under the same corpus. These two methods are commonly used in affective analysis, so comparing the classification results of SVM with them can explain the effect of SVM model to a certain extent.

Input: Vectorized data set
Output: Fusion characteristic matrix
Method: Perform the following steps
Step1: Travel through each comment
Step2: Get feature vector 1 though Doc2Vec
Step3: Get feature vector 2 though calculating sentiment Words
Step4: Get feature vector 3 though querying part of speech set
Step5: Merge the three features into one feature vector using hstack () as output feature

**Fig. 3** *Algorithm: The data fusion algorithm*

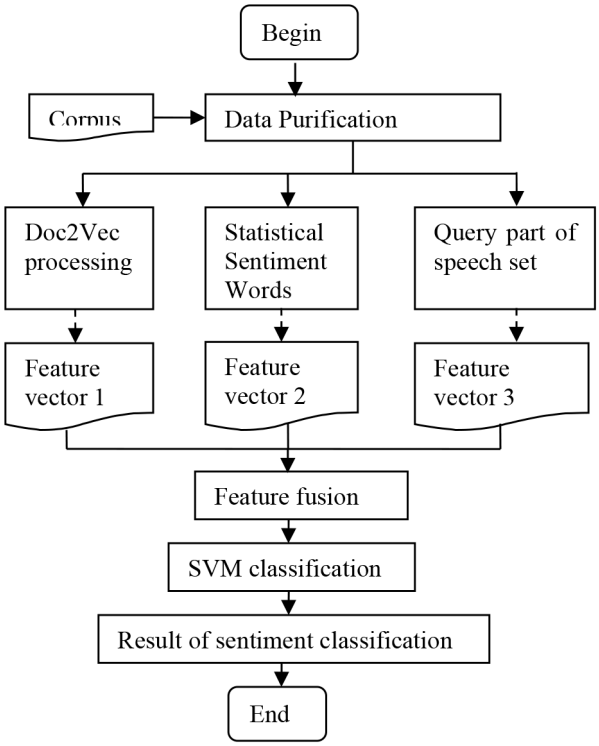


**Fig. 4** *Sentiment classification model*

**Table 1** Comparison of experimental results (using the same corpus)

| Method | Precision, % | Recall, % | F1, % |
|---|---|---|---|
| Nave bayes | 89.87 | 70.67 | 79.12 |
| Sentiment dictionary Matching | 85.15 | 65.43 | 74 |
| Doc2vec + SVM | 92.87 | 89.90 | 91.36 |

**Table 2** Result comparison with state-of- art methods

| Method | Precision, % | Recall, % | F1, % |
|---|---|---|---|
| Extension feature [4] | 82.5 | 78.4 | 80.39 |
| Gauss transfer learning [5] | 86.8 | 82.36 | 84.52 |
| Doc2vec + SVM | 92.87 | 89.90 | 91.36 |

### 5.2 Comparison of experimental results

The accuracy, precision, recall and F1 value of the evaluation indicators are used. The comparative experimental results used are shown in Tables 1 and 2.

We also compare the result of our method with the art-of-state methods, and Gauss Transfer Learning [5] used the same corpus with us.

By comparing the experimental data in Tables 1 and 2, it can be found that compared with Nave Bayes, sentiment dictionary matching and other state-of-art methods, the proposed Doc2vec +

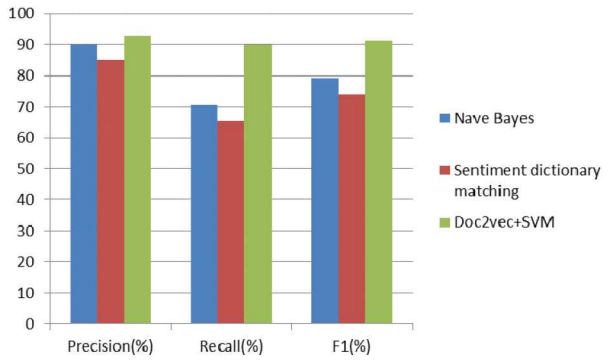


**Fig. 5** *Result comparison with Nave Bayes and sentiment dictionary matching*

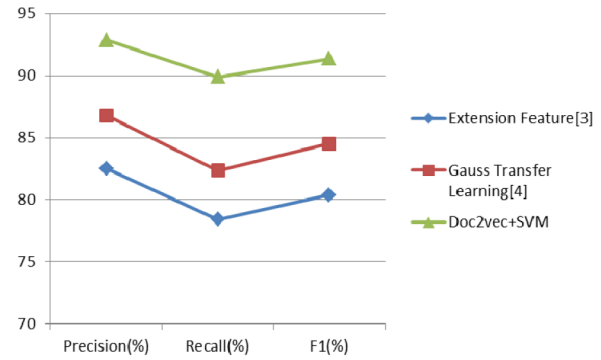


**Fig. 6** *Result comparison with state-of-art methods*

SVM-based sentiment analysis method has higher precision, recall and F1 values.

As can be seen from Figs. 5 and 6, the results using the proposed method in this paper is higher than other state-of-art methods, which shows that Doc2vec + SVM-based sentiment analysis method is more effective than Nave Bayes and other state-of-art methods.

This is because the sentiment dictionary method uses only one feature, while Nave Bayes ignores context and data magnitude. The Doc2vec + SVM-based sentiment recognition method proposed in this paper adopts context and semantic information. At the same time, it pays attention to data pre-processing before using algorithm. Therefore, it overcomes the shortcomings of the above methods and achieves good results.

## 6 Conclusion

In this work, we introduce a Doc2vec + SVM approach, which can incorporate context and semantics information in the process of sentiment classification. A data purification method is used before deep learning, removing more noisy information. The method proposed in this paper has achieved good results compared with Nave Bayes, sentiment dictionary matching and state-of-art methods, which verifies the validity of Doc2vec + SVM algorithm in the field of micro-blog sentiment analysis.

## 7 Acknowledgements

The work in this paper is supported by the National Natural Science Foundation of China under Grant No (61402134); Jiangsu Province '333' project (BRA2015108); high level talent work start-up project of Jinling Institute of Technology (40620022).

## 8 References

[1] Pang, B., Lee, L., Vaithyanathan, S.: 'Thumbs up?: sentiment classification using machine learning techniques'. Proc. of the ACL-02 Conf. on Empirical Methods in Natural Language Processing – Volume 10, Association for Computational Linguistics, Philadelphia, USA., 2002, pp. 79–86

[2] Liu, B.: 'Sentiment analysis and opinion mining', *Synth. Lect. Hum. Lang. Technol.*, 2012, **5**, (1), pp. 1–167

[3] Pang, B., Lee, L.: 'Opinion mining and sentiment analysis', *Found. Trends Inf. Retriev.*, 2008, **2**, (2), pp. 1–135

[4] Ye, F.: 'Sentiment classification for Chinese micro-blog based on the extension of network terms feature', *Adv. Comput. Comput. Sci.*, 2018, **551**, (1), pp. 231–241

[5] Wu, D., Gui, L., Chen, Z.: 'Sentiment analysis based on deep representation learning and Gauss process transfer learning', *J. Chin. Inf.*, 2017, **31**, (1), pp. 169–176

[6] Bermingham, A., Smeaton, A.F.: 'Classifying sentiment in microblogs: is brevity an advantage?' Proc. of the 19th ACM Int. Conf. on Information and Knowledge Management, ACM, New York, USA., 2010, pp. 1833–1836

[7] Wu, F., Song, Y., Huang, Y.: 'Microblog sentiment classification with contextual knowledge regularization'. Proc. of the Twenty-Ninth AAAI Conf. on Artificial Intelligence, Austin, USA., 2015, pp. 2332–2338

[8] Go, A., Bhayani, R., Huang, L.: 'Twitter sentiment classification using distant supervision'. CS224N Project Report, Stanford, 2009, pp. 1–12

[9] Hu, X., Tang, J., Gao, H., *et al.*: 'Unsupervised sentiment analysis with emotional signals'. Proc. of the 22nd Int. Conf. on World Wide Web, New York, USA., 2013, pp. 607–618

[10] Tang, D., Wei, F., Qin, B., *et al.*: 'Building large-scale twitter-specific sentiment lexicon: a representation learning approach'. Proc. of the 24th Int. Conf. on Computational Linguistics, Dublin, Ireland, 2014, pp. 172–182

[11] Hassan, A., Mahmood, A.: 'Deep learning approach for sentiment analysis of short texts'. 3rd Int. Conf. on Control, Automation and Robotics (ICCAR), Nagoya, Japan, 2017, pp. 705–710

[12] Joana, G., Alcione, P., Coelho de Andrade, G., *et al.*: 'A deep learning approach for sentiment analysis applied to hotel's reviews'. NLDB: Natural Language Processing and Information Systems, Alicante, Spain, 2018, pp. 48–56