Microblog Sentiment Analysis Based on Paragraph Vectors

Chengcheng Hu*, Xuliang Song
National Engineering Laboratory for Information Security Technologies, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, 100091, China.

* Corresponding author. Tel.:+86-10-82546719; email: huchengcheng@iie.ac.cn
Manuscript submitted February 9, 2015; accepted May 10, 2015.
doi: 10.17706/jcp.11.1.83-90

Abstract: Microblog sentiment analysis aims at discovering the users' attitude of hot events. Difficulties of microblog sentiment analysis lie on the short length of text and lack of labeled corpora. Para2vec based on deep learning attracts people's attention recently and the low-dimensional paragraph vectors trained by para2vec get excellent results on sentiment analysis. But when applying it for sentiment analysis on microblogs, we find it does not work so well as on ordinary texts. In this paper, we analyse the weakness of microblog sentiment analysis based on paragraph vectors. And then, we propose two categories of methods, model extension and emotional tendency vectors, to improve the model para2vec. The experimental results confirmed the rationality of our methods. Data analysis shows that our improved methods can effectively reduce the adverse effects of the short text and greatly improve the accuracy of sentiment analysis.

Key words: Paragraph vector, word vector, short text, unigram, bigram, emotional tendency vector, unsupervised learning.

1. Introduction

A survey released by China Internet Network Information Center (CNNIC) stated that Chinese microblog users scale of 275 million, covering 43.6% of internet users as of June 2014. Microblog has gradually evolved into a popular opinion platform. Natural language processing research on microblog become a new hotspot, sentiment analysis is one of the key topics of them [1].

Sentiment analysis, or opinion mining, is an active area of study that analyses people's opinions, sentiments, evaluations, attitudes, and emotions via the computational treatment of subjectivity in text [2]. The researches follow Pang *et al.*'s work (2002), which treat sentiment classification of texts as a special case of text categorization issue [3].

A series of difficulties lie in sentiment analysis on microblog compared to ordinary text: the short length of text and lack of labeled corpora. Predecessors explore the field of sentiment analysis on microblog, which is called twitter abroad [4]. Pak *et al*. started study of sentiment analysis on twitter in 2005 [5]. They organized tweets as data sets and realized classifiers for sentiment analysis based on Naive Bayes, support vector machine and conditional random field. Read *et al*. proposed the use of emoticons in twitter and carried out a detailed demonstration [6].

In most studies before, text features are represented by simple bag-of-words techniques. But the bag-of-words techniques have certain disadvantages: high dimension of feature vectors, sensitiveness to various language styles, disregard of context semantics, losing word order information etc.

In 2014, Mesnil and Mikolov use advanced deep learning techniques para2vec for sentiment analysis on a well-known dataset of IMDB movie reviews and their experiment convinced a relative improvement of about 30% compared to bag-of-words techniques [7]. Their research stimulate people's attempt to applying paragraph vectors. Different from most of the conventional feature extraction, para2vec takes consideration of context semantics using low-dimensional representations. Thus, it is very hot in the past few months.

However, there are several challenges towards achieving the best possible accuracy when we apply para2vec to microblog sentiment analysis.it is not clear if paragraph vectors provide any significant gain over simple bag-of-words techniques. In this paper, we analysis the disadvantage of microblog sentiment analysis based on paragraph vectors and then propose two categories of methods, model extension and emotional tendency vectors, to improve the model para2vec.When we experiment on CCF2012, the Chinese microblog sentiment classification competition corpora, we achieve new state-of-the-art results, better than the original model para2vec, yielding a relative improvement of more than 8% in terms of accuracy rate. Our methods convincingly beat bag-of-words models, giving a relative improvement of about 23%. What can be proud of is that we outplay the best results on accuracy and F value.

2. Background Knowledge and Discussed Problems

2.1. Word Vector

We start by discussing previous method word2vec. This method is the inspiration for paragraph vectors.

Word2vec is a tool based on deep learning and released by Google in 2013. This tool adopts two architectures, continuous bag-of-words (CBOW) and skip-gram, to learn the vector representations of words, whose dimension is generally 50 to 300. similarity in vector space can be used to represent the similarity of text semantic. An interesting character of word2vec is its additive compositionality. the official example is: the vector ("king") - vector ("man") + vector ("woman") $\approx$ vector ("queen") [8].

A well-known framework for word2vec is shown in Fig. 1. The task is to predict a word given the other words in a context. In this framework, every word is mapped to a unique vector. The concatenation or sum of the vectors is then used as features for prediction of the next word in a sentence [9]. Fig. 1 shows context of three words ("I", "like" and "watch") is used to predict the fourth word ("movies").

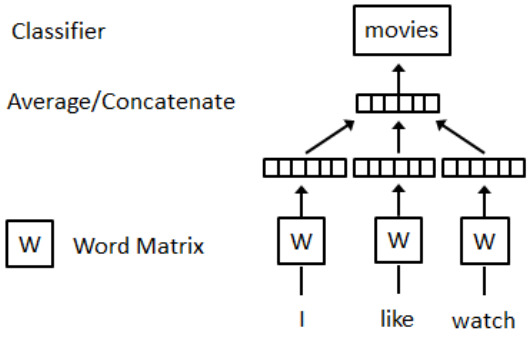


Fig. 1. Framework for word2vec.

2.2. Paragraph Vector

Para2vec for learning paragraph vectors is inspired by the method word2vec. The paragraph vectors are also asked to contribute to the prediction task of the next word given many contexts sampled from the paragraph [8]. More formally, the only change in this model compared to the framework for word2vec is showed in Fig. 2, where context is constructed from W and D. The paragraph token can be thought of as another word.

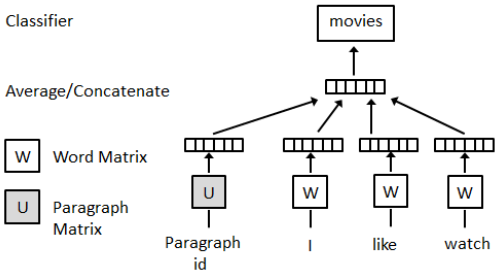


Fig. 2. Framework for para2vec.

2.3. Corpora for Experiment

We use the API provided by Sina to gather microblogs, 3000 for each of the two topics, a total of 6000. Then we chose 1000 for each topic to tag randomly. The results are shown in Table 1 after correction.

Table 1. Statistical Results of Experimental Corpora

topics	Positive	Negative	neutral	Total (each topic)	Neutral proportion
1 Looks good, free list	203	573	324	1000	32.4%
2 Our team qualifies in advance	641	85	274	1000	27.4%
total	844	658	598	2000	29.9%

We start to train 200-dimensional microblog vectors by para2vec with the 2000 microblogs as input. Then we conduct experiments for sentiment analysis to compare microblog vectors with the tf-idf bag-of-words techniques. As can be seen in Fig. 3, microblog vectors trained by para2vec hardly beat traditional bag-of-words techniques as Mesnil and Mikolov did on ordinary texts. After analysis, we find that it is the short length of text that leads to the low quality of microblog vectors. So we proposed two categories of methods, model extension and emotional tendency vectors, to compensate for the defects of short microblogs in para2vec. Our methods will be showed in the next chapter.

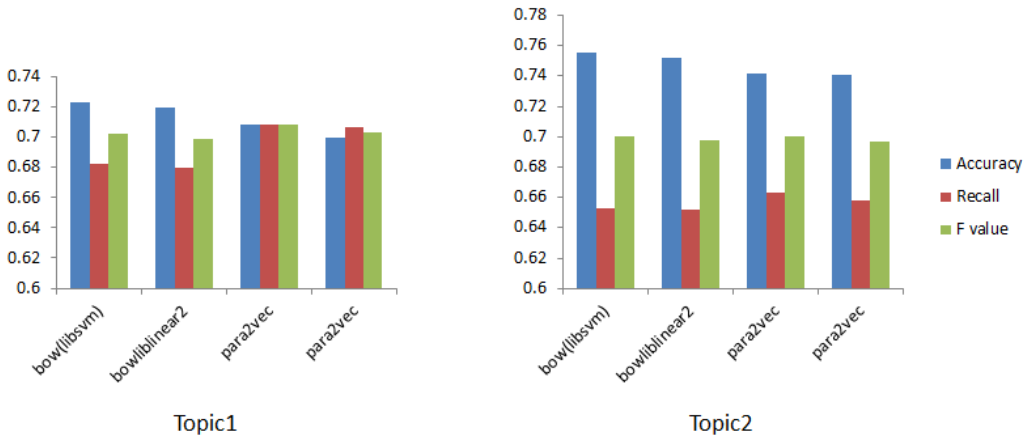


Fig. 3. Para2vec in comparison with bag-of-words.

3. Our Methods

3.1. Extension of Para2vec

Short length of text leads to the infrequent occurrence of each microblog token in para2vec, reducing the training results subsequently. In order to compensate for the weakness of microblog vectors in para2vec, we

try to carry out the extension during training process. Inspired by the bigram model in natural language processing, we attempt to change the input format of para2vec from unigram to bigram. Thus, the input of para2vec is expressed as *microblog ID, bigram$_1$, bigram$_2$, bigram$_3$, …, bigram$_m$*. Among the expression, *m* refers to the number of bigram elements in each microblog. For example, the unigram input of the sentence ("I like watch movies") in para2vec is "*microblog ID, I, like, watch, movies*", changing to bigram input as "*microblog ID, I–like, watch–movies*". Bigram of microblogs owns the ability to express the semantics more detailedly and precisely than unigram. When we use the microblog vectors trained by bigram input to conduct sentiment analysis, we find that it can distinguish delicate and ambiguous emotions like doubt and denial.

Aiming at increasing the frequency of paragraph token in principle, we combine unigram with bigram as input, called unigram + bigram. It is expressed as *microblog ID, unbigram$_1$, unbigram$_2$, unbigram$_3$, …, unbigram$_n$ + bigram$_1$, bigram$_2$, bigram$_3$, …, bigram$_m$*. Among the expression, *n* refers to the number of unigram elements and *m* refers to the number of bigram elements in each microblog. For example, the sentence ("I like watch movies") is represented as "*microblog ID, I, like, watch, movies, I–like, like–watch, watch–movies*". When we make use of the input unigram + bigram to train paragraph vectors, we extend the length of the microblog in form as well as increase the frequency of microblog token in para2vec in principle. So we can obtain microblog vectors which are closer to the real semantics and emotions. We prove that microblog vectors with precise representation will be more effective for sentiment analysis.

Another valuable characteristic of the model para2vec is its unsupervised learning. So a large scale of unlabeled cheap microblogs can be made use of as an extension of corpora. Considering that it is quite easy to get plenty of unlabeled microblog, we supplement the training corpora with these cheap data. Thus we utilize all the microblogs gathered before, including 2000 labeled and 4000 unlabeled microblogs, to train microblog vectors. It can be trusted that the experimental results will increase with the help of 4000 unlabeled microblogs.

3.2. Emotional Tendency Vector

There is a special advantage in para2vec. When we are training paragraph vectors, we can get word vectors at the same time. Assisted by emotion dictionary, we introduce these additional products to calculate emotional tendency vector of each word. Thus we can get the emotional tendency vector of each microblog after a series of composition to support the sentiment analysis. We will unfold the details of the calculation next.

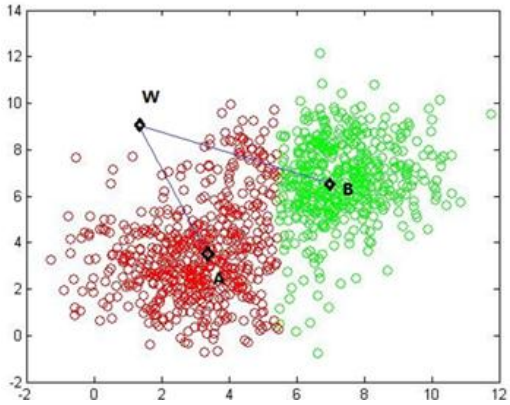


Fig. 4. Emotional words space.

Emotional dictionary is a collection that contains emotional words and classifies them by their emotional tendencies: positive and negative.

Hownet emotional dictionary:

Positive: love, appreciate, happy, curious, cheers,…

Negative: sad, suspect, despise, unsatisfied, regret, ...

Fig. 4 show the demonstration when mapping the emotional words vectors to vector space. *A* is the center of the positive words the same as *B* is the center of the negative words. We can calculate the emotional tendency vector d_i of each word W_i:

$$d_i = (d_{iA}, d_{iB}) \tag{1}$$

In Equation (1), d_{iA} is the cosine distance between word vector W_i and *A*, and so is it to d_{iB} :

$$d_{iA} = \cos(W_i, A), d_{iB} = \cos(W_i, B) \tag{2}$$

Then calculate the emotional tendency vector of each microblog D :

$$D = (D_A, D_B) \tag{3}$$

In Equation(3), $D_A = \frac{1}{n_w}\sum_{i=1}^{n_w} d_{iA}, D_B = \frac{1}{n_w}\sum_{i=1}^{n_w} d_{iB}$, n_w represents the number of words in each microblog.

We obtain a new 202-dimensional vector by adding the emotional tendency vector to the microblog vector. The 202-dimensional vectors are used for sentiment analysis. Then, we compare the result with the original microblog vector.

4. Experiment

Table 2 reports the results of each individual method in our experiments. The classifier is libsvm. We have found that the bag of words model performed the worst, with para2vec slightly better than it. The most competitive methods are the comprehensive methods based on our improvement. By analysis of Fig. 5, we find the results are enhancing consecutively with our methods, especially after introducing unigram + bigram input. Thus, the rationality of our methods was verified.

Table 2. Experimental Result of Corpora Sina

Methods	Topic 1			Topic 2		
	Accuracy	Recall	F value	Accuracy	Recall	F value
1	0.723	0.682	0.702	0.755	0.653	0.700
2	0.708	**0.708**	**0.708**	0.742	**0.663**	**0.700**
3	0.714	0.710	0.712	**0.758**	0.665	0.708
4	**0.758**	0.715	0.736	0.773	0.673	0.72
5	0.764	0.726	0.745	0.775	0.674	0.721
6	**0.771**	**0.727**	**0.748**	**0.778**	**0.677**	**0.724**

Methods:

1: Bag of words model

2: Unigram training (para2vec)

3: Bigram training (para2vec)

4: (Unigram+Bigram) training (para2vec)

5: (Unigram+Bigram) training +Emotional vector (para2vec)

6: (Unigram+Bigram) training +Emotional vector +Unlabeled data (para2vec)

After examining the classification results, we find that our methods have outstanding performance in

discriminating complex and ambiguous sentiment like doubt and denial. Table 3 presents the incorrect classifications in original para2vec, which we distinguish them successfully with the help of our improvements.

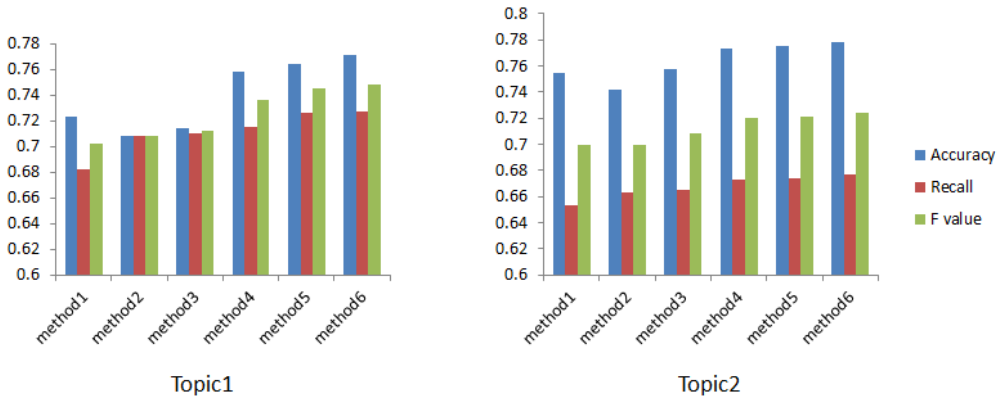


Fig. 5. Experimental result of corpora Sina.

Table 3. Correct Classifications of Our Methods

1	Our team got the first in the group and qualified in advance, is there anything more difficult than this?
2	Since our team qualified in advance, why don't you work hard!!!!
3	Fortunately I did not give up because I got you finally.

CCF is influential in Chinese microblog sentiment analysis competition. We conducted experiments on microblog corpora of CCF2012, which is gathered from Tencent microblog, including 20 topics and 20,000 microblogs. As is showed in Table 4 and Fig. 6, we achieve new state-of-the-art results, better than the original model para2vec, yielding a relative improvement of more than 8% in terms of accuracy rate. Our method convincingly beats bag-of-words models, giving a relative improvement of about 23%. What can be proud of is that we outplay the best results on accuracy and F value.

Table 4. Experimental Result of Corpora CCF2012

methods	Micro average			Macro Average		
	Accuracy	Recall	F value	Accuracy	Recall	F value
1	0.722	0.652	0.685	0.768	0.663	0.712
2	0.810	0.767	0.788	0.812	0.78	0.796
3	0.828	0.785	0.806	0.821	0.787	0.804
4	**0.855**	0.820	0.837	**0.867**	0.845	**0.856**
5	**0.858**	0.824	0.841	**0.869**	0.846	**0.857**
6	**0.862**	0.840	**0.851**	**0.870**	0.850	**0.860**
7	0.704	0.460	0.556	0.752	0.454	0.566
8	0.850	**0.850**	0.850	0.854	**0.854**	0.854

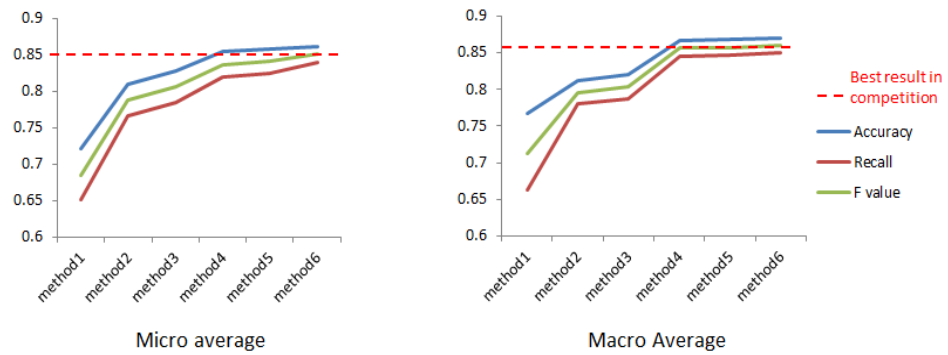


Fig. 6. Experimental result of corpora CCF2012.

Methods:

1: Bag of words model
2: Unigram training (para2vec)
3: Bigram training (para2vec)
4: (Unigram+Bigram) training (para2vec)
5: (Unigram+Bigram) training +Emotional vector (para2vec)
6: (Unigram+Bigram) training +Emotional vector +Unlabeled data (para2vec)
7: Average result in competition
8: Best result in competition

5. Conclusions

Different from most of the conventional methods for sentiment classification, para2vec focuses on the semantic features between words rather than the simple lexical or syntactic features. In this paper, we propose two categories of methods, model extension and emotional tendency vectors, to improve the model para2vec. Our methods compensate for the weakness of para2vec on microblog sentiment analysis with simple techniques successfully. But it is far from enough. Model extension is just an aspect of improvement. We will try to train paragraph vectors based on RNNLM (recurrent neural network language modeling) in the future.

Acknowledgment

The research work was supported by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant No. XDA06030200 and the National Key Technology R&D Program under Grant No. 2012BAH46B03.

References

[1] Du, Z. L., & Zhang, Y., S. (2013). Weibo short text sentiment analysis research. From: http://www.cnki.net/

[2] Hutto, C. J., & Gilbert, E. (2014). VADER: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*.

[3] Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning Sentiment-specific word embedding for Twitter sentiment classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics* (pp. 1555–1565).

[4] Xie, L., Zhou, M., & Sun, M. (2012). Hierarchical structure based hybrid approach to sentiment analysis of chinese micro blog and its feature extraction. *Journal of Chinese Information Processing*, *26(1)*, 73-83.

[5] Pak, A., & Paroubek, P. (2010, May). Twitter as a corpus for sentiment analysis and opinion mining. *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*: *Vol. 10* (pp. 1320-1326).

[6] Read, J. (2005). Using emoticons to reduce dependency in machine learning techniques for sentiment classification. *Proceedings of the ACL Student Research Workshop* (pp. 43-48).

[7] Mesnil, G., Ranzato, M. A., Mikolov, T., & Bengio, Y. (2015). Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews. *Under Reviewing of Proceedings of Workshop Contribution at ICLR*.

[8] Deng, P., & Lu, G., M. (2014). *Word2vec: Actual combat of Deep Learnin.* From: http://techblog.youdao.com/?p=915

[9] Le, Q. V., & Mikolov, T. (2014). Distributed representations of sentences and documents. *Proceedings of*

the 31st International Conference on Machine Learning. Beijing, China.

[10] Xiong, F., L., Deng, Y., H., & Tang, X., S. (2015). The architecture of Word2vec and its applications. *Journal of Nanjing Normal University(Engineering and Technology Edition).*

[11] Gao, K., Xu, H., & Wang, J. (2014, September). Emotion classification based on structured information. *Proceedings of Multisensor Fusion and Information Integration for Intelligent Systems (MFI)* (pp. 1-6).

[12] Goldberg, Y., & Levy, O. (2014). Word2vec Explained: deriving Mikolov *et al*.'s negative-sampling word-embedding method. From: http://arxiv.org/abs/1402.3722

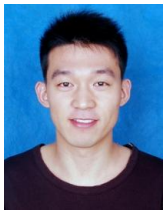
[13] Yu, M., & Dredze, M. (2014). Improving lexical embeddings with semantic knowledge. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers)* (pp. 545–550).

[14] Su, Z., Xu, H., Zhang, D., & Xu, Y. (2014, September). Chinese sentiment classification using a neural network tool — Word2vec. *Proceedings of Multisensor Fusion and Information Integration for Intelligent Systems (MFI)* (pp. 1-6).

[15] Pouransari, H., & Ghili, S. (2014). *Deep Learning for Sentiment Analysis of Movie Reviews.* From: http://cs224d.stanford.edu/reports/PouransariHadi.pdf

[16] Ghiyasian, B., & Guo, Y. F. (2014). *Sentiment Analysis Using Semi-supervised Recursive Auto Encoders and Support Vector Machines.* From: http://cs229.stanford.edu/proj2014/Bahareh.pdf

Chengcheng Hu was born in Hunan, China, in 1988. She received her B.S. degree in digital media technology in 2012 at Communication University of China, Beijing. Currently, she is a postgraduate student from Institute of Information Engineering, Chinese Academy of Sciences, China. Her research interests include natural language processing, data mining and machine learning..



Xuliang Song was born in Hebei, China, in 1989. He received his B.S. degree in information security in 2012 at Yanshan University, Qinhuangdao, China. Currently, he is a postgraduate student from Institute of Information Engineering, Chinese Academy of Sciences, China. His research interests include natural language processing, data mining and machine learning.