Sentiment Analysis of Chinese Microblogs Based on Layered Features

Dongfang Wang and Fang Li

Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering,
Department of Computer Science and Engineering,
Shanghai Jiao Tong University, China
{mickey,fli}@sjtu.edu.cn

Abstract. Microblogging currently becomes a popular communication way and detecting sentiments of microblogs has received more and more attention in recent years. In this paper, we propose a new approach to detect the sentiments of Chinese microblogs using layered features. Three layered structures in representing synonyms and highly-related words are employed as extracted features of microblogs. In the first layer, "extremely close" synonyms and highly-related words are aggregated into one set while in the second and the third layer, "very close" and "close" synonyms and highly-related words are aggregated respectively. Then in every layer, we construct a binary vector as a feature. Every dimension of a feature indicates whether there are some words in the microblog falling into that aggregated set. These three features provide perspectives from micro to macro. Three classifiers are respectively built from these three features for final prediction. Experiments demonstrate the effectiveness of our approach.

Keywords: Layered Features, Sentiment Analysis, Microblogs.

1 Introduction

As a social communication tool, microblogging is very popular among Internet users. More and more people post their opinions towards products or political views. Therefore sentiment analysis of microblogs can help social studies or marketing, and becomes a quite hot topic. The features used in microblogs sentiment analysis are of great impact to the classifying performance.

Mohammad et al. [1] implemented a number of features including Ngrams, characters in upper case, lexicons, POS (Part-of-Speech) tags, hashtags, punctuations, emoticons, negations and so on. They concluded that sentiment lexicon features along with Ngrams features led to the most gain in performance. Pak and Paroubek [2] analyzed the distribution of words frequencies in positive, negative and neutral sets. They concluded that POS tags are strong indicators therefore they use Ngrams and POS tags as features. In the work of Huang et al. [3], using sentiment words, POS tags, punctuations, adversative words, and emoticons as features achieved their best results. However, extracting these features needs a lot of computational and linguistic work.

L.C. Kiong et al. (Eds.): ICONIP 2014, Part II, LNCS 8835, pp. 361–368, 2014.

Unigrams are widely used for simplicity and excellent performance. In the experiments of Pang et al. [4], using an SVM trained on Unigrams achieved the best result. Read [5], Bora [6], Purver and Battersby [7] also adopted Unigrams as features. Go et al. [8] concluded that Unigrams are simple but useful while only using Bigrams as features will cause sparseness of the feature space. However, even for Unigrams, sparseness is very severe. Due to the large quantity of Unigrams in the corpus, one word will easily drown in the sea of Unigrams and its characteristic cannot be well detected. Saif et al. [9] used semantic feature set and sentiment-topic feature set to alleviate the sparseness.

By analyzing thousands of Chinese microblogs, we find that many synonyms and highly-related words exist among microblogs. The meanings of these synonyms are quite close and the highly-related words usually appear in the same topics. Therefore synonyms and highly-related words can be regarded as the same when detecting sentiments.

In this paper, we propose a novel approach based on the observations above. We construct three layered features by aggregating synonyms and highly-related words. These three layered features provide perspectives in different levels, from micro to macro. By combining them we get a more complete perspective and achieve our best results. For comparison, we also conduct experimental comparison with other methods. Unlike [9], our method can be applied on any corpus even though they do not contain series of product entities and topics.

The rest of this paper is organized as follows. In section 2, we introduce feature extraction. The voting method is described in section 3. Section 4 shows the experimental results and discussion. Finally we conclude this paper in section 5.

2 Feature Extraction

2.1 Preprocessing

Before extracting features, we firstly split words using the tool ICTCLAS[1] because there is no space between Chinese words. URLs, hashtags, emoticons, users' names (with a character "@" before names), stop-words and words that appear less than 3 times in the training set are removed.

2.2 Synonyms and Highly-Related Words

Synonyms. Synonyms are words expressing the same or similar meanings, like 高兴(happy) and 开心(joyful). According to how "close" the meanings of synonymous words are, synonyms are grouped into three levels respectively. The structure is from fine to coarse.

Highly-Related Words. Highly-related words are words that their meanings are different but the meanings are highly-related. For example, 棉农(cotton grower) and 茶

[1] http://www.ictclas.org/

农(tea grower) means different but they are both plant growers. According to how "close" the relationships of highly-related words are, they are grouped into three levels respectively. The structure is from fine to coarse.

We use the HIT IR-Lab Tongyici Cilin[2] to find highly-related words and synonyms in Chinese. Other dictionary like SentiWordNet[3], or formulas like PMI can also be used to find synonyms or highly-related words in different languages. Examples of synonyms and highly-related words are shown in Table 1.

Table 1. Examples of Synonyms and Highly-related Words

Notation (How close they are)	Examples of Synonyms	Examples of Highly-related Words
α (Extremely close)	1. 忧愁(worry)，犯愁 (worry);	1. 侨胞(countryman residing abroad), 港胞(countryman residing Hong Kong);
	2. 忧闷(depressed)，忧郁(gloomy);	2. 爱国同胞(patriotic fellow-countryman), 爱国者(patriot);
	3. 烦闷(anguish)，烦乱(upset);	3. 非洲人(African)，亚洲人(Asian);
	4. 委屈(grievance)，憋屈(grievance);	4. 意大利人(Italian)，美国人(American);
β (very close)	1.忧愁(worry)，犯愁(worry)，忧闷 (depressed)，忧郁(gloomy);	1. 侨胞(countryman residing abroad)，港胞 (countryman residing Hong Kong)，爱国同胞 (patriotic fellow-countryman)，爱国者(patriot);
	2.烦闷(anguish)，烦乱(upset)，委屈 (grievance)，憋屈(grievance);	2. 非洲人(African)，亚洲人(Asian), 意大利人(Italian)，美国人 (American);
γ (close)	忧愁(worry)，犯愁(worry)，忧闷 (depressed)，忧郁(gloomy)，烦闷(anguish)，烦乱(upset)，委屈(grievance)，憋屈 (grievance);	侨胞(countryman residing abroad)，港胞 (countryman residing Hong Kong)，爱国同胞 (patriotic fellow-countryman)，爱国者(patriot)，非洲人(African)，亚洲人(Asian)，意大利人(Italian)，美国人(American);

2.3 Reduction of Feature Dimension

Words in microblogs are of large quantity which leading to the sparseness of vector space of Unigrams. Compressing feature dimensions using synonyms and highly-related words will relieve this dilemma. In our method, a word and all of its synonyms and its highly-related words are all expressed in only one feature dimension. For example, the word 火星(Mars), 荧惑(Mars), and 土星(Saturn) are all represented by one feature dimension, 荧惑(Mars) is synonymous word to 火星(Mars) while 土星 (Saturn) is highly-related word to 火星(Mars). Also, the synonymous words and highly-related words of 荧惑(Mars) and 土星(Saturn) are included. That is, the words in one feature dimension consist of a closure set. We define the calculation for finding the synonyms and highly-related words of a word w_i and of a word set W in equation (1-3). And the algorithm for calculating a closure word set $W_{closure}$ for a word w_i, and for finding the set of all the word sets $\mathscr{A}$ is shown in Algorithm 1.

After all the words are split into disjoint closure sets, we use the results to construct features. Each dimension of our feature vector represents whether there are some

[2] `http://www.datatang.com/data/42306/`

[3] `http://sentiwordnet.isti.cnr.it/download.php`

words contained in the microblog falling into that set. As shown in Fig. 1, the feature vector is a binary vector and the dimension of the vector equals to the number of sets.

$$S(w_i) = \{Synonyms\ of\ w_i\}\ ,\ \ H(w_i) = \{Highly\text{-}realted\ words\ of\ w_i\} \tag{1}$$

$$f(w_i) = S(w_i) \cup H(w_i) \tag{2}$$

$$f(W) = \bigcup_{i=1}^{n} f(w_i),\ W = \{w_1, w_2, w_3, ..., w_n\} \tag{3}$$

Algorithm 1. Calculating Closure Word Sets

```
$Calculate\_Closure$ (word $w_i$){
    $W_{closure} = \{w_i\}$ ; $W_{new} = \{w_i\}$ ;
    While ( $W_{new} \neq \varnothing$ ){
    $W_{new} = f(W_{closure}) - W_{closure}$ ;
    $W_{closure} = W_{closure} \cup W_{new}$ ;
    }
    Return $W_{closure}$ ;
}
```

```
$Split\_Words\_into\_Disjoint\_Sets$ (corpus $m$){
    $M = \{m_1, m_2, ..., m_k\} = \{words\ in\ m\}$ ;
    $\mathcal{A} = \varnothing$ ;
    While ( $M \neq \varnothing$ ){
    $C = Calculate\_Closure(m_0)$ , $m_0 \in M$ ;
    $M = M - C$ ;
    $\mathcal{A} = \mathcal{A} \cup \{C\}$ ;
    }
    Return $\mathcal{A}$ ;
}
```

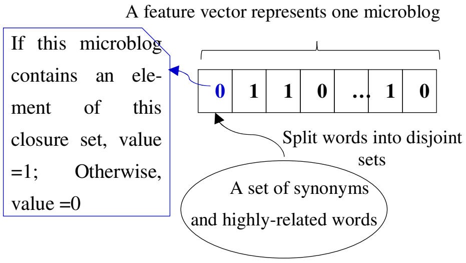
A feature vector represents one microblog
If this microblog contains an element of this closure set, value =1; Otherwise, value =0
0 1 1 0 ... 1 0
Split words into disjoint sets
A set of synonyms and highly-related words

Fig. 1. Description of One Feature Vector

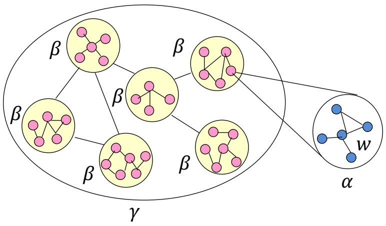

Fig. 2. Structure of the Layered Word Sets

2.4 Layered Feature

There are three layers of compressed features in our approach. The difference between these three layers is how "close" the meanings of synonymous words or how "close" the relationships of related words in one feature dimension. We group words that are "extremely close", "very close", "close" respectively and denote one aggregated set as α, β, γ respectively.

Note that one γ consists of many β and one β consists of many α. The aggregation is the most precise in the first layer while the aggregation is the least precise in the third layer, satisfying the inequality (4). The structure of the layered word sets is shown in Fig. 2. In every layer, words are split into disjoint sets. We construct a feature use the method shown in section 2.3 for every layer. These features provide perspectives from micro to macro. Therefore combing all of them, we get a more complete view of the Microblogs and it can provide more information than one single layer.

$$max\{distance_{\alpha}(w_{\alpha i},w_{\alpha j})\} \prec max\{distance_{\beta}(w_{\beta i},w_{\beta j})\} \prec max\{distance_{\gamma}(w_{\gamma i},w_{\gamma j})\} \quad (4)$$

3 Voting SVM Classifiers

Previous research [4] has shown that SVM performs excellently for sentiment analysis. Therefore we employ SVM classifier in our approach. Once we construct the three layered features, three SVM classifiers are built from each layer of features respectively. To combine the three classifiers, voting strategy is used. We consider that the features of the three layers are equally important, so we set the weights of these three classifiers to be equal when voting. The final classified result will be the majority predicted label [10]. The strategy of the classification is shown in Algorithm 2.

Algorithm 2. The Strategy of the Classification

```
Voting(microblog m){
    Classifier $C_i$ is trained from features in layer $i$ $(1 \le i \le 3)$;
    $votingResult = 0$;
    for $(i = 1; i \le 3; i{+}{+})$
    {if (Ci predicts the m is positive) $votingResult{+}{+}$;}
    if $(votingResult \ge 2)$ return positive;
    else return negative;
}
```

Table 2. Component of the Cilin

	Number of word sets
In the 1st layer	13440
In the 2nd layer	3880
In the 3rd layer	1423
In the 4th layer	95
In the 5th layer	12

4 Experiments and Result Analysis

To reveal the effectiveness of our approach, in this section we carry out a series of experiments on two datasets and compare our results with other well-known approaches. All the data used in this paper comes from Xinlang Weibo[4], a large Twitter-like Chinese microblogging website.

4.1 Dictionary and Datasets

HIT IR-Lab Tongyici Cilin. We use the cilin to extract layered synonyms and highly-related words. In the cilin, there are five layers. We only make use of the first three layers because the last two layers are too coarse which will cause over-aggregation of words, shown in Table 2.

MoodLens Dataset [11]. There are four categories of sentiments: angry, disgusted, joyful and sad in this dataset. In our experiments, we take joyful as positive sentiment

[4] http://www.weibo.com

and the other three emotions as negative sentiments. We randomly select 5000 positive microblogs and 5000 negative microblogs from this dataset.

NLP&CC2013 Dataset[5]. Eight categories of sentiments serve as the labels. We take angry, disgusted, afraid and sad as negative sentiments and take happiness and like as positive sentiments. We use 965 positive microblogs and 965 negative microblogs.

4.2 Experimental Results

In the experiments, we compare our proposed approach with methods using Unigrams, Bigrams, Unigrams + POS tags, Bigrams + POS tags as features respectively. We also compare with lexicon method and the method proposed in [2]. In lexicon method, a microblogs will be labeled according to the number (the first deciding factor) and the strength (the second deciding factor) of the positive and negative words, using Dalian Ligong lexicon[6]. In [2], Bigrams and POS tags are used as the features with salience-feature-selection, and Naïve Bayes classifier is trained for classification. The performances of three classifiers that built from features of different layers are also examined respectively. In all experiments, 5-fold cross validation is conducted. And we use the criterion (5-7) to evaluate our system. Experimental results (%) on both datasets are shown in Table 3 & 4, respectively.

$$Precision_{emotion=i} = \frac{\#system\_correct(emotion=i)}{\#system\_proposed(emotion=i)} \quad i \in \{Pos, Neg\} \tag{5}$$

$$Recall_{emotion=i} = \frac{\#system\_correct(emotion=i)}{\#manually\_label(emotion=i)} \quad i \in \{Pos, Neg\} \tag{6}$$

$$F-measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7}$$

Table 3. Results on MoodLens Dataset

Method	Positive			Negative		
	Precision	Recall	F-measure	Precision	Recall	F-measure
Unigrams	60.44	62.18	61.30	61.06	59.30	60.17
Bigrams	59.24	51.08	54.86	57.00	64.85	60.68
Unigrams + POS tags	60.32	61.87	61.09	60.86	59.30	60.07
Bigrams + POS tags	60.24	53.23	56.52	58.11	**64.87**	61.30
Lexicon Method	54.66	44.96	49.34	53.26	62.71	57.60
Method of [2]	59.62	51.90	55.49	57.41	64.85	60.91
The 1st layered classifier	61.73	61.67	61.70	61.71	61.77	61.74
The 2nd layered classifier	61.62	62.38	62.00	61.91	61.15	61.53
The 3rd layered classifier	59.72	61.87	60.78	60.45	58.27	59.34
Combination of three layered classifier	**62.94**	**62.49**	**62.71**	**62.76**	63.21	**62.98**

[5] http://tcci.ccf.org.cn/conference/2013/

[6] http://ir.dlut.edu.cn/EmotionOntologyDownload.aspx

Table 4. Results on NLP&CC2013 Dataset

Method	Positive			Negative		
	Precision	Recall	F-measure	Precision	Recall	F-measure
Unigrams	74.16	72.93	73.54	73.37	74.59	73.98
Bigrams	65.54	53.59	58.97	60.75	71.82	65.82
Unigrams + POS tags	71.88	71.38	71.63	71.58	72.08	71.83
Bigrams + POS tags	62.84	61.05	61.93	62.13	63.90	63.00
Lexicon Method	59.69	57.64	58.65	59.05	61.07	60.04
Method of [2]	60.17	73.55	66.19	65.99	51.31	57.73
The 1st layered classifier	74.59	74.59	74.59	74.59	74.59	74.59
The 2nd layered classifier	73.86	71.82	72.83	72.58	74.58	73.57
The 3rd layered classifier	69.01	65.19	67.05	67.02	70.73	68.82
Combination of three layered classifier	**75.56**	**75.14**	**75.35**	**75.28**	**75.70**	**75.49**

From Table 3 & 4, we can find that on both datasets, our proposed method combining three layered classifiers achieves the best F-measure on both positive and negative sentiments. The classifier of the first layer also largely outperforms other comparison methods. It indicates that aggregating synonyms and highly-related words in the microblogs is a feasible approach for sentiment analysis. The results also reveal that the final combined classifier outperforms all the single classifiers. This may due to features in different layers provide different perspectives, from “micro” to “macro”. Through voting, all the information gets together and improves the performance.

Observations of the classifying results indicate that layered features can handle sparseness more effectively. Examples like the microblog 他真令我们伤悲 (he makes us so sad) containing the word 伤悲 (sad) is predicted right in our approach but it is predicted wrong when using other features. And in the case that replacing the word 伤悲 (sad) with 悲伤 (sad), all the approaches predict right. The reason is, generally people use the word 悲伤 (sad) to express sad mood and the word 伤悲 (sad) is rarely used. However, these two words are grouped into one feature dimension when using layered feature. Therefore, layered features will make better use of rare words.

4.3 Further Discussion

Why Keep the Classifier of the Third Layer as One of the Voters? As shown in Table 3&4, the classifier of the third layer doesn’t perform as well as the classifiers of the first and the second layers. However, we still keep this classifier. Because words not strongly related can have similarities, and the third layer provides a “macro” perspective to express similarities of the words. Meanwhile, the third layer provides the least sparse feature space.

Why not Combine the Three Layered Features as a Single Feature Vector? We have tested and the performance is not better than the voting method. Using a single feature vector will increase the dimensions of the feature space. And the second and the third layer will have less effects because their feature dimensions are less than the first layer.

5 Conclusions

In this paper, we propose a novel approach based on layered features. In this method, we assume that synonyms and highly-related words play equal roles in sentiment analysis and can be regarded as the same. Based on this assumption, three layered features are constructed by grouping synonyms and highly-related words. The difference among features of the three layers is how close the meanings or the relationships of the words are. And three classifiers are trained from features of different layers respectively. The final results will be a major vote comes from the three classifiers. In the comparison experiments, our approach achieves the best results.

Acknowledgement. This work was supported by the National Natural Science Foundation of China (Grant No. 61375053 and No. 60873134). We thank professor Liqing Zhang greatly for his valuable comments and suggestions.

References

1. Mohammad, S.M., Kiritchenko, S., Zhu, X.D.: NRC-Canada: Building the State-of-the-Art in Sentiment Analysis of Tweets. In: 7th International Workshop on Semantic Evaluation, pp. 321–327 (2013)
2. Pak, A., Paroubek, P.: Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In: Proceedings of the International Conference on Language Resources and Evaluation, pp. 1320–1326 (2010)
3. Huang, S., You, J.P., Zhang, H.X., Zhou, W.: Sentiment Analysis of Chinese Micro-blog Using Semantic Sentiment Space Model. In: 2nd IEEE International Conference on Computer Science and Network Technology, pp. 1443–1447 (2012)
4. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification using Machine Learning Techniques. In: Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86 (2002)
5. Read, J.: Using Emoticons to Reduce Dependency in Machine Learning Techniques for Sentiment Classification. In: Proceedings of the ACL Student Research Workshop, pp. 43–48 (2005)
6. Bora, N.N.: Summarizing Public Opinions in Tweets. International Journal of Computational Intelligence and Applications, 41–55 (2012)
7. Purver, M., Battersby, S.: Experimenting with Distant Supervision for Emotion Classification. In: Proceedings of the 13th Conference of the European Chapter of the ACL, pp. 482–491 (2012)
8. Go, A., Bhayani, R., Huang, L.: Twitter Sentiment Classification Using Distant Supervision. CS224N Project Report, pp. 1-12. Stanford University (2009)
9. Saif, H., He, Y., Alani, H.: Alleviating Data Sparsity for Twitter Sentiment Analysis. In: 2nd Workshop of CEUR, pp. 2–9 (2012)
10. Tsutsumi, K., Shimada, K., Endo, T.: Movie Review Classification Based on a Multiple Classifier. In: The 21th PACLIC, pp. 481–488 (2007)
11. Zhao, J., Dong, L., Wu, J., Xu, K.: MoodLens: an Emoticon-Based Sentiment Analysis System for Chinese Tweets in Weibo. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1528–1521 (2012)