Sentiment Analysis of Name Entity for Text

Xinzhi Wang
Department of Engineering Physics
Tsinghua University
Beijing, China
wxz15@mails.tsinghua.edu.cn,

Hui Zhang
Department of Engineering Physics
Tsinghua University
Beijing, China
zhhui@mail.tsinghua.edu.cn,

Jiayue Wang
Department of Engineering Physics
Tsinghua University
Beijing, China
wangjy15@mails.tsinghua.edu.cn,

Yang Zhou
Department of Engineering Physics
Tsinghua University
Beijing, China
zhou-y15@mails.tsinghua.edu.cn

***Abstract*—Recent years, big data has attracted increasing interest. Sentiment analysis from microblog as one kind of big data also receive great attention. Some recent research works are not suitable for sentiment analysis as the result that users prefer to express their feelings in individual ways. In this paper, a framework is proposed to calculate sentiment for aspects of event. Based on some state of art technologies, we build up one flowchart to get sentiment for aspects of event. During the process, name entities with the same meaning are clustered and sentiment carrier are filtered. In this way sentiment can be got even user express feeling for the same object with different words.**

***Keywords*-sentiment analysis framework, event analysis, microblog, big data**

I. Introduction

Recently, one tends to express sentiments at anytime and anywhere with the prevalence of online social media. Sina (http://weibo.com), Tencent (http://blog.qq.com/) in China and Twitter (www.twitter.com), Facebook (www.facebook.com) in USA, spread millions of personal posts on a daily basis by millions of users. Rich and enormous volume of data propagated through social media attracted enormous attention from researchers. However, information of these platforms is sparse and various, as user have different characteristics in expressing their opinions. When users describe the same objective, they may use different words. For instance some prefer using 'typhoon', and others prefer using the name of typhoon 'Rammasun' or 'Matmo' when they discuss the same event at the same time. As a result, analyzing and managing sentiment for text has become one kind of meaningful and challenging tasks in the area of affective computing, especially in microblog.

This paper propose one practical way to get the sentiment of aspects of event even they are expressed in different ways. In this paper, introduction and related work are given in section I and section II. The framework is introduced in section III. Technologies about prominent word extraction and prominent name entity extraction are discussed in section IV. Methods of sentiment analysis for prominent name entity and experiment are denoted in section V and section VI, respectively. Finally, conclusions are made in the last section.

DOI reference number: 10.18293/SEKE2016-002

II. Related Work

Sentiment analysis is subset of semantic analysis. Semantic analysis in processing microblogs[1] and news[2] include detecting and tracing event. Semantic information is also employed in some video processing[3], [4]. Sentiment analysis reuse the technologies semantic analysis, and focus on three levels: word level, sentence level, and article level.

Opinion words are extracted mainly through three ways. 1) manual approach, 2) dictionary based approach, and 3) corpus based approach. Manual approach is very time consuming but accurate [12]. The dictionary based approach and opinion words collection have a major shortcoming, since the same word may have different sentiment orientation in different domains. So far, several opinion word lists have been generated [13]. Semantic associations in text are mined from different aspect[14], [15]. These relations have been used in large scale news analysis[16].Rules or constraints are designed for connectives, such as AND, OR, BUT, EITHEROR, and NEITHERNOR [17]. The idea of intra-sentential and inter-sentential sentiment consistency were explored in [18]. Qiu G[19] employed dependency grammar to describe relations for double propagation between features and opinions. Liu [20] adopted the same context definition but used it for sentiment analysis of comparative sentences. Breck [21] went further to study the problem of extracting any opinion expressions, which might have any number of words. The Conditional Random Fields (CRF) method [22] was used as the sequence learning technique for extraction. Most of above works are based on probability, which cannot express two strong emotions at the same time.

Machine learning methods are widely used in both sentence and article level. Zhu Y [26] employed naive Bayesian classi-

fier to discriminate objective and subjectivity classification. Pang [27] employed three machine learning methods (i.e., Naive Bayes, maximum entropy classification, and support vector machines) on sentiment classification for text. Subsequent research also used other machine learning algorithms such as pruning and pattern recognition methods [28]. Wilson [29] pointed out that a single sentence may contain multiple opinions. Automatic sentiment classification was presented to classify the clauses of sentiment of every sentence down to four levels (i.e., neutral, low, medium, and high) by the strength of the opinions being expressed in individual clauses [30]. So, lots of researchers tried to break down the word boundary of sentiment carrier.

All of the previous works focus on the methods of deterministic algorithm, which contribute to the development of sentiment analysis. But few have combine these effective method. This paper make efforts to get the goal.

III. Task Description

With the proliferation of mass media, interest in mining sentiment and opinions in text has attracted more and more attention. Sentiment or opinions expressed by authors is affected by event, products, aspect of event, or aspect of products. Such as the sentence 'The camera of my phone is dim' expresses negative sentiment about aspect 'camera' of product 'phone'. Also the sentence 'Things went well, but the results were heartbreaking' expresses positive sentiment about event 'Things' and negative sentiment about aspect 'result' of event 'Things'.

This paper focus on the problem of finding sentiment orientation for prominent word of text. To do this job, we divide the whole process into four parts. Firstly, Extracting Prominent Word. Extracting prominent word is the basic work, which include word segmentation, removing stop words, recognizing name entity. Secondly, Extracting Prominent Name Entity. Extracting prominent name entity is based on the fact that prominent word has been mined. This part contains representation of word, computing weight of word and name entity filtration. Thirdly, Extracting Sentiment Carriers For Name Entity. Sentiment carrier can be any type of word including nouns, verbs, adjectives, and adverbs, but not all word can express opinion. In this way, we design one method to recognize sentiment word and build relationship between these carriers and name entities, and details be discussed later. Fourthly, Sentiment Analysis For Text. Sentiment carriers are recognized as they express some sentiment. In this part, which kind of sentiment is expressed is analysed. Furthermore, sentiment orientation for name entity and sentiment orientation for text will be calculated.

Every part of the whole process shown in Fig.1 has been attracting a lot of researchers' attention. This paper aims to do sentiment analysis employing improved state-of-art methods, and more details will be discussed later.

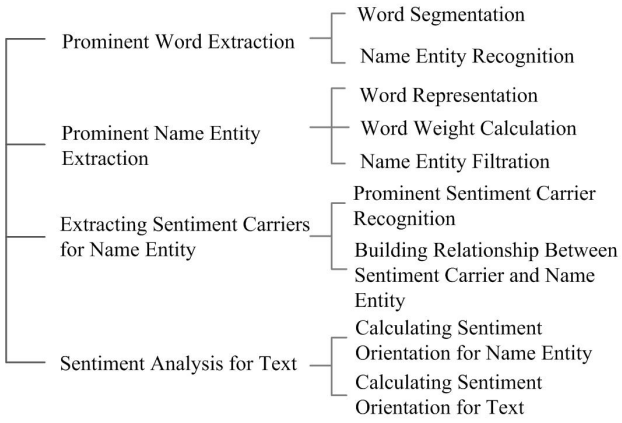

Fig. 1. Technological process of sentiment analysis for text.

IV. Prominent Word Extraction

To extract prominent word from Chinese, four main steps have to be considered. The three steps are word segmentation, name entity recognition, word representation, word weight calculation. With each part has been researched deeply, we choose some efficient method to finish our work. Details are discussed later.

A. Word Segmentation

Now, segmentation of Chinese word are mainly focus on three field including word based generative model, character based generative model, and dictionary based model. This paper employ one kind of word based generative model. Then the following three steps are token.

1) Search candidate by matching items in the word dictionary and build up wordnet like {(a,ab),(b,bcd),(c),(d)}
2) Get best candidate sequence employing viterbi algorithm using bigram dictionary.
3) Determine pos sequence using word dictionary and transfer matrix table.

MM(Markov Model) and HMM(Hidden Markov Model) are used to do segment of Chinese. In this way, word and corresponding pos can be got.

B. Name Entity Recognition

Name entity denotes name for certain specific person, location, group, or organization containing person name, location name, and organization name. Also, there are lots researches have been done, here we employ method proposed by Huaping Zhang[11], in which cascaded hidden Markov model is used for Chinese named entity recognition.

The whole process is divided into three levels, person name recognition, location recognition, group and organization name recognition. In each level, roles are employed. Furthermore, state transmission matrix and state observation emission matrix are counted. At last, patterns of roles are set to finalize which consecutive roles sequence combined to these entity name. Details can be found in [11].

After recognizing name entity, it turns to extracting prominent name entity. To achieve this goal, weight calculation method are introduced after word representation using word vector. At last, ways to filter prominent name entity are described.

C. Word Representation

One famous text representation model is VSM(Vector Space Model), in which every document is represented by a vector with every element be the IF/IDF weight of corresponding word. Also, word have always been regarded as a vector in some topic models such as LDA [7]. Here we employ GloVe [5], [6] model which has perfect performance in capturing fine-grained semantic and syntactic regularities proposed by Richard Socher.

More details can be found in the paper of Socher et al.[6]. Word vector employ one kind of unsupervised model learning word representations which have great performance on word analogy, word similarity, and named entity recognition tasks. In this way, each word was represented by a vector.

D. Word Weight Calculation

Many methods have been proposed to calculate word weight in the research domain of keyword extraction, such as classical TextRank [8] and TFIDF [9]. Both methods have been widely used in natural language processing area. TextRank functions as PageRank which is employed by Google in information retrieval. Here we employ TextRank in calculating word weight. Also, this method is often been used in abstract generation. Calculation of word weight consists of the following main steps:

1) Identify relations that connect vertexes, and use these relations to draw directed and weighted edges between vertexes. Weight of words relations are got as,

$$S(w_i, w_j) = ne(w_j, w_i) \tag{1}$$

where $S(w_i, w_j)$ means the frequency of both w_i and w_j appear in the same words window. Normally, the length of window is set to be five.

2) Calculate word weight through iteration equation as,

$$WS(w_i) = (1-d) + d * \sum_{w_j} \frac{S(w_j, w_i)}{\sum_{w_k} S(w_j, w_k)} WS(w_j) \tag{2}$$

where d is a damping factor that can be set between 0 and 1. The damping factor is often set to 0.85. In this way, words with high weight can be easily found.

3) Sort vertices based on their final score. Important words are assigned high weight.

Until now, we can get prominent word after segmenting, representing and weighting words, which facilitate following calculation including name entity extraction and sentiment orientation analysis.

E. Name Entity Filtration

Name entity filtration aims to find primary name entity from large scale of text. We have introduced textrank to calculate word weight above, which means we can set one threshold to get rid of the noises. Traditionally, two ways can be employed, namely set the minimum weight or set the qualified percentage. Which method should be chosen based on specific text. More details will be discussed in experiment.

Some of these entities may be related to specific sentiments. In next section, we will discuss one method to calculate sentiment orientation of words.

V. Sentiment analysis for Prominent Name Entity

There are both objective and subjective sentences in text, such as sentence 'MacPhee, who is a girl, would be happy to eat it!' is one compounding sentence. 'MacPhee' and 'happy' are subjective when expressing some sentiment. This section aims to mine sentiment expressed by 'happy' for 'MacPhee'.

A. Prominent Sentiment Carrier Recognition

Generally speaking, some words with specific pos (part of speech), such as adjective element, adjective, adverb element, adverb, verb, exclamation, and even some punctuation can express various kind of sentiment explicitly.

In this paper, to recognize sentiment carrier, we choose a small kernel set of 1166 terms manually as sentiment seed based on [23]. There are 6 kinds of primary sentiment including love, joy, surprise, anger, sadness, and fear. Complex emotions are various combination of the six primary sentiments. Furthermore, the six kinds of sentiment words are expanded using bootstrapping method according to one Chinese lexicon named 'HaGongDaCiLin', most of which are adjectives and adverbs. Vectors of these words are described as v_k^j, namely the kth seed word in sentiment the jth sentiment. Moreover, similarity of two words s_{v_i, v_k^j} can be got as follow:

$$s_{v_i, v_k^j} = \frac{v_i * v_k^j}{|v_i||v_k^j|}, \tag{3}$$

where v_i is the word vector of w_i from V.A. If $s_{v_i, v_k^j} > 0.7$, then we select v_i as sentiment carrier, but we can not determine which kind of sentiment v_i express as the result that some antonyms can be used in the same situation with similar word vectors. K-means is employed when it comes to clustering.

As microblog is some kind of short text, the sentiment of these short messages keeps unanimous at most situation. Then sentiment carriers which appear in the same microblog express similar sentiment.

B. Building Relationship Between Sentiment Carrier and Name Entity

Sentiment expressed by carriers belongs to some entity, formulated as $w_i \in N(w_j)$ with $N(w_j)$ be the set of sentiment carrier for word w_j, such as word 'happy' attribute to 'MacPhee', namely $'happy' \in N('Macphee')$, in sentence 'MacPhee would be happy to eat it!'. The principle of proximity within the frame of punctuation is the main criterion when

it comes to consider building relationship between sentiment carrier and name entity.

C. Calculating Sentiment Orientation for Name Entity

Actually, word weight can be got using method introduced above, which means each sentiment carrier and name entity have their values. Furthermore, sentiment carrier contribute to the sentiment of name entity in different degree. So here we using the simple average weight to deal the problem of calculating sentiment orientation for name entity.

$$se_{ij} = \frac{\sum_k WS(w_k) * se_{kj}}{\sum_k WS(w_k)}, \qquad w_k \in Ne(w_i) \tag{4}$$

where $WS(w_k)$ is the weight of word w_k, and se_{kj} is the sentiment orientation of word w_k to the jth sentiment. At the same time w_k must be element of sentiment carrier set for word w_i. In this way, the prominent sentiment of name entity depend more on distinctive high weight sentiment carriers.

D. Calculating Sentiment Orientation for Text

Text is sequence of words, including name entities and sentiment carriers. Name entities act as the major descriptive content, while sentiment carriers are more like attributes for name entity. Name entity weight differently when they appear in the text. If the most weighted object is very 'joy', then the sentiment of text tend to be 'joy'. In this way, we employ one way similar to calculation of name entity to calculate sentiment orientation for text.

$$\begin{aligned} tse_{lj} &= \frac{\sum_i WS(w_i) * se_{ij}}{\sum_i WS(w_i)}, \\ &= \frac{\sum_i \sum_k WS(w_i) WS(w_k) * se_{kj}}{\sum_k WS(w_k) \sum_i WS(w_i)} \\ w_k &\in Ne(w_i) \\ w_k &\in Ne(te_l) \end{aligned} \tag{5}$$

where $Ne(te_l)$ means the name entity set appear in text te_l. tse_{lj} is the sentiment orientation in dimension jth for corresponding text. Other variables keep the same meaning as before.

VI. Experiments

In this part, we will explain how these methods works in details. Three parts, prominent word and name entity extraction, sentiment carriers recognition, sentiment analysis for text, are included.

A. Prominent Word and Name Entity Extraction

The corpus we employed in this paper is crawled from 'weibo.sina.com' with keyword 'rainstrom' in Chinese with 339541 microblogs involved. 'weibo.sina.com' is one of the most popular social media platform just like Twitter in USA, which allows users to express opinions freely. As a result, there are large amount of noisy information mixed in the interesting contents. So, before we take steps do future study, textrank with adaptive window size(one sentence as window) is used firstly to get topic sentences while removing noisy sentences. Details are shown in Table.I. Precision, Recall and F1-Value should be higher than 90% according to [11]. Then textrank with window size five is employed to calculate word weight.

name of items	token number
number of microblog in corpus	339541
number of word in corpus	76364
number of person name in corpus	8917
number of place name in corpus	3766
number of organization name in corpus	78
number of topic sentence from corpus	33907
number of word from topic sentences	8581
number of person name from topic sentences	545
number of place name from topic sentences	670
number of organization name from topic sentences	17

TABLE I
CORPUS INFORMATION WHEN EXTRACTING WORD AND NAME ENTITY.

Name entities with high weight are left to do future analysis. Some nouns such as 'rainstorm' and 'rain' who carries high weight are also kept for future analysis.

B. Extracting Sentiment Carriers for Name Entity

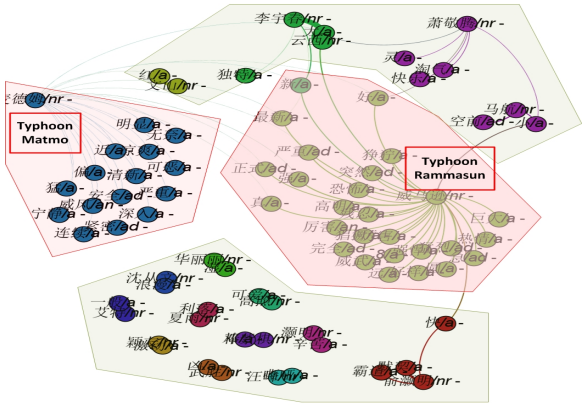

Fig. 2. Name entity (typhoon Matmo and Rammasun) and its adjectives.

As is discussed in V.A, each word is expressed by one word vector. Words with similar attributes are assigned analogous vector. Such as 'downpour' and 'rainstorm' are often decorated by 'sudden' and 'abrupt'. As a result, vector of 'downpour' and vector of 'rainstorm' is quite similar. But when it turns to adjectives, it is not that effective in finding synonyms. Distance between 'beautiful' and 'ugly' are small, as they can be used in the position such as 'the dress is ugly/beautiful'. In our practice, we manually select 1066 sentiment seeds to extract adjectives who carry sentiment. Then based on the hypothesis that sentiment stay consistent in the same microblog without negative words, we compute the sentiment orientation of these sentiment carriers. Fig.2 shows part of the results.

C. Sentiment Analysis for Text

The same object can be expressed in different ways. Such as typhoon can be express by specific typhoon name, rain can be expressed by downpour and so on. So in this part,

we cluster words with the same meaning together before calculating sentiment for them.

Then by employing sentiment carriers for cluster of name entities, we can get the sentiment orientation of aspect from text. The result show that 'fear' is strong for 'rainstorm' and 'typhoon'. Slight joy with 0.24 exists in text for 'rainstorm'. 'joy' and 'anger' both with 0.39 exist in text for 'typhoon'. Sentiment 'joy' originate from jokes produced when 'rainstorm' and 'typhoon' happen, while 'anger' and 'fear' result form their destructiveness. Follow the same steps, we can get sentiment for other entities appears in the text.

VII. Conclusion

This paper propose one practical framework in determining sentiment analysis of name entities. When we try to get short but meaningful information from big data, this framework can be used as guidance. In all, this paper have four contributions:

1.Propose one framework of new technologies. Large amount of new technologies have been spring up these years. Each of them have advantages and disadvantages. This paper combine some practical works such as word vector representation, textrank to get sentiment orientation for text.

2.Name entity clusters. By employing word vector representation, we find that noun words with similar meaning can be clustered easily, but when it comes to adjectives, it will not work that well. This is because, adjectives with different meaning may share the same context.

3.Recongnize sentiment carriers and calculate sentiment orientation for name entities. Most sentiment is carried by adjectives. By calculating their weight, prominent word can be filtered. Then sentiment seeds are used to get the sentiment orientation after building the relation of sentiment carriers and name entities.

There are still some future works needed to be completed. Such as finding practical ways to cluster adjectives which have similar meaning. Also, more experiments should be implemented to verify the performance of related technologies in the framework.

References

[1] Xu Z, Liu Y, Xuan J, et al. Crowdsourcing based social media data analysis of urban emergency events[J]. Multimedia Tools & Applications, 2015:1-18.

[2] Xu Z, Liu Y, Yen N Y, et al. Crowdsourcing based Description of Urban Emergency Events using Social Media Big Data[J]. IEEE Transactions on Cloud Computing, 2016:1-1.

[3] Xu Z, Liu Y, Mei L, et al. Semantic based representing and organizing surveillance big data using video structural description technology[J]. Journal of Systems & Software, 2015, 102(C):217-225.

[4] Xu Z, Mei L, Liu Y, et al. Semantic enhanced cloud environment for surveillance data management using video structural description[J]. Computing, 2014:1-20.

[5] Huang E H, Socher R, Manning C D, et al. Improving word representations via global context and multiple word prototypes[J]. In Proc. of the Annual Meeting of the Association for Computational Linguistics (ACL), 2012, 1:873-882.

[6] Jeffrey Pennington, Richard Socher, and Christopher D.Manning. Glove: Global vectors for word representation. In Proceedings of EMNLP. 2014.

[7] Jordan M I, Blei D M, Ng A Y. Latent Dirichlet Allocation[J]. Journal of Machine Learning Research, 2003, 3:465–473.

[8] Tarau P, Mihalcea R. TextRank: Bringing Order into Texts[J]. Unt Scholarly Works, 2004.

[9] Salton G, Buckley C. Term-weighting approaches in automatic text retrieval[J]. Information Processing & Management An International Journal, 1988, 24(5):513-523.

[10] Wu H C, Luk R W P, Wong K F, et al. Interpreting TF-IDF term weights as making relevance decisions[J]. Acm Transactions on Information Systems, 2008, 26(3):55-59.

[11] Yu H K, Zhang H P, Liu Q, et al.Chinese Named Entity Recognition Based on cascaded hidden Markov model[J]. Journal of communication, 2006, 2:87-94.

[12] S. R. Das and M. Y. Chen.Yahoo! for Amazon: Sentiment extraction from small talk on the Web, Management Science, vol. 53, pp. 1375C1388, 2007.

[13] Rao Y, Lei J, Liu W, et al. Building emotional dictionary for sentiment analysis of online news[J]. World Wide Web-internet and Web Information Systems, 2014, 17(4):723-742.

[14] Luo X, Xu Z, Yu J, et al. Building Association Link Network for Semantic Link on Web Resources[J]. IEEE Transactions on Automation Science & Engineering, 2011, 8(3):482-494.

[15] Hu C, Xu Z, Liu Y, et al. Semantic Link Network-Based Model for Organizing Multimedia Big Data[J]. IEEE Transactions on Emerging Topics in Computing, 2014, 2(3):376-387.

[16] Xu Z, Wei X, Luo X, et al. Knowle: A semantic link network based system for organizing large scale online news events[J]. Future Generation Computer Systems, 2015, s 43V44:40-50.

[17] V. Hatzivassiloglou and K. McKeown.Predicting the semantic orientation of adjectives, Proceedings of the Joint ACL/EACL Conference, pp. 174C11, 1997

[18] S. Huang, Z. Niu, C. Shi. Automatic construction of domain-specific sentiment lexicon based on constrained label propagation[J]. Knowledge-Based Systems, 2014, 56(3):191C200.

[19] G. Qiu, B. Liu, J. Bu, et al. Expanding domain sentiment lexicon through double propagation[C]// Proceedings of the 21st international jont conference on Artifical intelligenceMorgan Kaufmann Publishers Inc., 2009:1199-1204.

[20] G. Ganapathibhotla and B. Liu. Identifying Preferred Entities in Comparative Sentences, Proceedings of the International Conference on Computational Linguistics, COLING, 2008.

[21] E. Breck, Y. Choi, and C. Cardie. Identifying expressions of opinion in context, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2007.

[22] M. Zheng, Z. Lei, X. Liao, et al. Identify Sentiment-Objects from Chinese Sentences Based on Cascaded Conditional Random Fields[J]. Journal of Chinese Information Processing, 2013, 27(3):69-76.

[23] Parrott W. Emotions in social psychology: Essential readings[M]. Psychology Press, 2001.

[24] Jijian Xie, Chengping Liu. Fuzzy mathematics method and application [M]. Huazhong university of science and technology press, 2000.

[25] Salton G, Wong A, Yang C S. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11): 613-620.

[26] Zhu Y, Tian H, Ma J, et al. An Integrated Method for Micro-blog Subjective Sentence Identification Based on Three-Way Decisions and Naive Bayes[M]// Rough Sets and Knowledge TechnologySpringer International Publishing, 2014:844-855.

[27] Pang B, Lee L. Opinion Mining and Sentiment Analysis[J]. Foundations and Trends in Information Retrieval, 2008, 2(1):459-526.

[28] Karamibekr M, Ghorbani A A. Lexical-Syntactical Patterns for Subjectivity Analysis of Social Issues[M]// Active Media TechnologySpringer International Publishing, 2013:241-250.

[29] T. Wilson, J. Wiebe, and R. Hwa.Just how mad are you? Finding strong and weak opinion clauses, Proceedings of AAAI, pp. 101-109, 2004.

[30] Wilson T, Wiebe J, Hoffmann P. Recognizing contextual polarity in phrase-level sentiment analysis[J]. International Journal of Computer Applications, 2005, 7(5):347-354.