Wang *et al. EURASIP Journal on Wireless Communications and Networking* (2016) 2016:164
DOI 10.1186/s13638-016-0661-x

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Sentiment processing of social media information from both wireless and wired network

Xinzhi Wang*, Hui Zhang, Shengcheng Yuan, Jiayue Wang and Yang Zhou

Abstract

Recent years, information spreading under the environment of wireless communication has attracted increasing interest. Microblog platform on mobile terminals, as one product of wireless communication, facilitate information spreading and evolution by conveying message from peer to peer. Furthermore, sentiments from microblog reflect the attitude of peers on goods or events. Analysis of the sentiment can help in decision-making. Research work focuses on analyzing sentiment orientation for specific aspects of product with explicit names. However, it is not suitable for sentiment analysis of events using microblog data since users prefer to express their feelings in individual ways, namely the same object may be expressed in several ways. In this paper, a framework is proposed to calculate sentiment for aspects of events. First, we introduce some effective technologies in processing natural language, such as wordvec, HMM, and TextRank. Then, based on the state-of-art technologies, we build up a flowchart to get sentiment for aspects of events. At last, experiments are designed to prove these technologies on computing sentiment. During the process, name entities with the same meaning are clustered and sentiment carrier is filtered, with which sentiment can be got even users express their feeling for the same object with different words.

Keywords: Sentiment analysis, Web sematic information, Microblog, Wireless

1 Introduction

Nowadays, wireless has almost become necessities of smart mobile phone users. On the one hand, instant message through wireless has even change users' habit. Users do not have to rely on newspaper or TV to get information, and they tend to get instant message from their smart mobile phone. On the other hand, social media applications based on wireless arise frequently, such as Sina (weibo.com), Tencent (blog.qq.com) in China and Twitter (twitter.com), and Facebook (facebook.com) in the USA. One tends to express and receive sentiments at anytime and anywhere with the help of wireless. Millions of personal posts on a daily basis are spread by millions of users. Rich and enormous volume of data propagate through wireless and attract enormous attention from researchers. However, information from these wireless platforms is sparse and various, as users 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 "Haiyan" or "Sandy" when they discuss the same event at specific time. Even if they are related to the same event of the same aspect of event, the words are distinct. As a result, analyzing and managing sentiment for text has become meaningful and challenging tasks in the area of affective computing, especially in microblog, which is one typical product of wireless.

There are both objective and subjective sentences in wireless platforms. Some objective content expresses factual information about the world, such as the name entities, while some subjective content expresses personal feeling or beliefs on the events, such as adjectives. Furthermore, the subjective content is the sentiment expressed by users to objective content, so there exist a lot connections between subjective and objective contents. In other words, sentiment or opinions expressed by wireless users is activated by events, products, aspects of events,

*Correspondence: wxz15@mails.tsinghua.edu.cn
Department of Engineering Physics, Tsinghua University, Beijing, China

or aspects 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 makes efforts to analyze public opinion based on information of wireless, such as microblog. The research problem of this paper is finding sentiment orientation for prominent word of text. To do this job, we divide the whole process into four parts. The first part is extracting prominent word, which includes word segmentation, removing stop word, and recognizing name entity. The second part is extracting prominent name entity, which contains representation of word, computing weight of word, and name entity filtration. The third part is extracting sentiment carrier for name entity, which recognize sentiment word and build relationship between these carriers and name entities. The last part is sentiment analysis for text, which calculate sentiment orientation for name entity and text. The whole process can be found in Fig. 1.

This paper is organized as follows: introduction and related works are given in Sections 1 and 2. Technologies about prominent word extraction and prominent name entity extraction are discussed in Sections 3 and 4. Methods of sentiment analysis for prominent name entity and experiments are denoted in Sections 5 and 6, respectively. Finally, conclusions are made in the last section.

2 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 [5, 6]. It is usually used as the final check for automated methods. One of the simple techniques in the dictionary-based approach is based on bootstrapping using a small set of seed opinion words and an online dictionary, e.g., WordNet. So far, several opinion word lists have been generated [7, 8]. Semantic associations in text are mined from different aspects [9, 10]. These relations have been used in large-scale news analysis [11]. The dictionary-based approach and opinion word collection have a major shortcoming, since the same word may have different sentiment orientation in various domains. Corpus-based approach relies on syntactic or co-occurrence patterns and also a seed list of opinion words to find other opinion words in a large corpus. Rules or constraints are designed for connectives, such as AND, OR, BUT, EITHEROR, and NEITHERNOR [12]. The idea of intra-sentential and inter-sentential sentiment consistency was explored in [13, 14]. Qiu et al. [15] employed dependency grammar to describe the relations for double propagation between features and opinions. Ganapathibhotla and Liu [16] adopted the same context definition but used it for sentiment analysis of comparative sentences. Breck et al. [17] went further to study the problem of extracting any opinion expressions, which might have any number of words. The conditional random fields (CRF) method [18] was used as the sequence learning technique for extraction. Most of the above works were based on probability, which can not express two strong emotions at the same time.

Machine learning methods are widely used in both sentence and article levels. Zhu et al. [19] employed naive Bayesian classifier to discriminate objective and subjectivity classification. Pang and Lee [20] employed three machine learning methods (i.e., naive Bayes, maximum entropy classification, and support vector machine) on sentiment classification for text. Subsequent research also used other machine learning algorithms such as pruning and pattern recognition methods [21, 22]. Some of the above works were based on an assumption that one sentence expresses a single opinion from a single opinion holder. However, this assumption is not suitable for compound sentences. Wilson et al. [23] pointed out that a single sentence might 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 [10, 24]. A lot 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 contributes to the development of sentiment analysis. But few have combined these effective methods.

3 Prominent word extraction

Extracting prominent word is the basic work, which includes word segmentation, removing stop word, and recognizing name entity. Word segmentation is inevitable for Chinese as the result that there is no space in character sequence, which is different from English. To extract prominent word, segment of characters and all name entities including people, location, and organization names are recognized to facilitate later work. Three main steps are considered: word segmentation, word representation, and word weight calculation. With each part has been

Collecting Data	Original sentence	李昕每天都很开心。 (Chinese phoneticize : li xin mei tian dou hen kai xin.)
Prominent Word Extraction	Word Segmentation	李/ng, 昕/ng, 每天/r, 都/d, 很/d, 开心/a, 。/w
	Name Entity Recognition	**李昕/nr**, 每天/r, 都/d, 很/d, 开心/a, 。/w
Prominent Name Entity Extraction	Word Representation	李昕/nr : **[0 1 0 0.8 0]**, ⋯ , 开心/a : **[1 0 0.6 0 1]** , ⋯
	Word Weight Calculation	李昕/nr :**0.9** [0 1 0 0.8 0], ⋯ , 开心/a : **0.5** [1,0,0.6,0,1] , ⋯
	Name Entity Filtration	**李昕/nr :0.9 [0 1 0 0.8 0]**
Extracting Sentiment Carrier for Name Entity	Prominent Sentiment Carrier Recognition	**开心/a : 0.5 [1 0 0.6 0 1] [0 1 0 0 0 0]**(sentiment vector)
	Building Relationship Between Sentiment Carrier and Name Entity	李昕/nr :0.9 [0 1 0 0.8 0] ← 开心/a : 0.5 [1 0 0.6 0 1] [0 1 0 0 0 0]
Sentiment Analysis for Text	Calculating Sentiment Orientation for Name Entity	李昕/nr :0.9 [0 1 0 0.8 0] **[0 0.5 0 0 0 0]**
	Calculating Sentiment Orientation for Text	**[0 0.45 0 0 0]** for the text

Fig. 1 Technological process of sentiment analysis for text

studied in deep, some efficient methods are selected to complete our work. Details are discussed later.

3.1 Word segmentation

Now, segmentation of Chinese word is mainly focused on three fields including word-based generative model, character-based generative model, and dictionary-based model. This paper employs one kind of word-based generative model. The following three steps are taken.

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

$$p(w_0 w_1, \ldots, w_n) = \prod_{i=1}^{n} p(w_i | w_{i-1}) \tag{1}$$

where w_i is one word candidate and w_0 is the beginning of sentence. w_i is supposed to be one exclusive vector for word.

3. Determine pos(part of speech) sequence using word dictionary and transfer matrix table through

$$p(t_0, t_1, \ldots, t_n) = \prod_{i=1}^{n} p(t_i | t_{i-1}) \prod_{j=1}^{n} p(w_j | t_j) \tag{2}$$

where t_i is word pos candidate and t_0 is the beginning pos the sequence specially.

For instance, there is a Chinese sentence like "Li Xin and Fu Ximing are a couple. The location they first meet is Xueyuan road, Wulipo village, Rizhao City, Shandong Province. Later, both of them were enrolled in by school of computer science from Tsinghua University. Li Xin lived a happy life". After segmentation, the sentence is changed into some fraction in Fig. 2.

Markov model (MM) and hidden Markov model (HMM) are used to perform segment in Chinese. In this way, word and corresponding pos can be detemined.

3.2 Name entity recognition

Name entity denotes names for certain specific person, location, group, or organization. A lot of researches have studied on these. Method proposed by Yu et al. [25] is employed in this paper, in which cascaded hidden Markov model is employed for Chinese name entity recognition.

To recognize name entity employing cascaded Markov model, three levels are designed and different roles are cited [25]. The first level is person name recognition. Twelve roles in all are employed with 11 described by Yu et al. The other one is defined as the beginning of a sentence. Then, we produce some context information by half machine and half man-made. Extra training data is also added. Furthermore, state transmission matrix and state observation emission matrix are counted. At last, patterns of roles are set for person name recognition. The second level is location recognition; nine roles are used according to Yu et al., and the beginning role of sentence is added just as in the first level. Transmission matrix, observation matrix, and pattern are also mined. The last level is group and organization name recognition. Group name recognition is a little different from person and location name recognition, as the group name can be longer and may be composed of some other group names. As a result, iterations are designed to perform location name recognition until convergence. In the process, ten roles of regaining role and another nine from Yu et al. are employed. Transmission matrix, observation

“ 李昕和付希明是情侣，他们认识在山东日照市五里坡学院路，并一起进入清华大学计算机学院学习。李昕每天都很开心。”
[li xin he fu xi ming shi qing lv, ta men ren shi zai shan dong ri zhao shi wu li po xue yuan lu, bing yi qi jin ru qing hua da xue ji suan ji xue yuan xue xi. li xin mei tian dou hen kai xin.]

(a)

“李/ng, 昕/ng, 和/cc, 付/v, 希/b, 明/ag, 是/vshi, 情侣/n, ， /w, 他们/rr, 认识/v, 在/p, 山东/ns, 日照市/ns, 五/None, 里/f, 坡/n, 学院路/nz, ， /w, 并/cc, 一起/s, 进入/v, 清华大学/ntu, 计算机/n, 学院/nis, 学习/v, 。/w, 李/ng, 昕/ng, 每天/r, 都/d, 很/d, 开心/a, 。/w”

(b)

Fig. 2 a Original Chinese sentence with phonetic transcription. **b** Result of segment of sentence employing HMM

matrix, and pattern are also needed. The details are as follows.

$$P(x_1, x_2, \ldots, x_{t+1}, y_{t+1} = i|\theta) = \arg\max_j P(x_1, x_2, \ldots, x_t, y_t = j|\theta) tr_{j,i} em_{i,x_{t+1}} \quad (3)$$

where x_i is the term generated in the last step; more specifically, it is the segment result for person name entity recognition step and is the person name entity recognition result for location name entity. The same is true for group and organization name entity recognition. y_i is the corresponding potential roles of term x_i. θ is the parameter including both transmission matrix and observation matrix. $tr_{j,i}$ denotes the value of role j transmit to i, $em_{i,x_{t+1}}$ means the value of role i generating term x_{t+1} . Both $tr_{j,i}$ and $em_{i,x_{t+1}}$ belong to the parameter θ.

Using the method cited above, the sentence in Fig. 2 is transformed into Fig. 3. Here, “li xin,” “fu xi ming,” “wu li po,” and “qing hua da xue ji suan ji xue yuan” are combined as person name, location name, and organization name, respectively.

In this section, HMM-based methods to segment Chinese sentence including recognizing name entity are introduced. This is the first step in deciding sentiment of name entity.

4 Prominent name entity extraction

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. Representation for word employs word vector proposed in recent years. Word weight calculation uses TextRank which has been proven to be effective in reflecting the value of word. Finally, valuable name entity is saved for future work. To achieve this goal, word representation through word vector and weight calculation method are introduced. At last, the ways to filter prominent name entity are described.

4.1 Word representation

One famous text representation model is vector space model (VSM) [26], in which every document is represented by a vector, and every element is the IF/IDF weight of corresponding word. Also, word is usually regarded as a vector in some topic models such as LDA [27]. In this paper, GloVe [28] model is employed, which has perfect performance in capturing fine-grained semantic and syntactic regularities proposed by Richard Socher.

$$J = \sum_{i,j=1}^{V} f(X_{ij}) \left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - logX_{ij} \right)^2 \quad (4)$$

“李昕/nr, 和/cc, 付希明/nr, 是/vshi, 情侣/n, ， /w, 他们/rr, 认识/v, 在/p, 山东/ns, 日照市/ns, 五里坡/ns, 学院路/nz, ， /w, 并/cc, 一起/s, 进入/v, 清华大学计算机学院/nt, 学习/v, 。/w, 李昕/nr, 每天/r, 都/d, 很/d, 开心/a, 。/w ”

Fig. 3 Resulted name entity recognition

where V is the size of the vocabulary, X_{ij} is the number of times word j occurs in the context of word i, b_i is a bias for w_i, and $\tilde{b}_k$ is an additional bias for $\tilde{w}_k$. $f(X_{ij})$ is the weighting function obeying specific properties as

$$f(x) = \begin{cases} (x/x_{\max})^{3/4} & x < x_{\max} \\ 1 & \text{otherwise} \end{cases} \quad (5)$$

More details can be found in [28, 29]. Word vector employs one kind of unsupervised model to learn word representation, which has great performance on word analogy, word similarity, and named entity recognition tasks. In this way, each word is represented by a vector.

4.2 Word weight calculation

Various methods are proposed to calculate word weight in the area of keyword extraction, such as classical TFIDF [30, 31] and TextRank [32]. Both methods are widely used in natural language processing. TextRank functions as PageRank (PageRank is employed by Google in information retrieval). Here, TextRank is employed to calculate word weight. This method is often used in abstract generation. Calculation of word weight consists of the following steps:

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

$$S(w_i, w_j) = ne(w_j, w_i) \quad (6)$$

   where $S(w_i, w_j)$ is the frequency of both w_i and w_j appears in the same words window. Normally, the length of window is set to be five.
2. Calculate word weight through iterations

$$WS(w_i) = (1-d) + d * \sum_{w_j} \frac{S(w_j, w_i)}{\sum_{w_k} S(w_j, w_k)} WS(w_j) \quad (7)$$

   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, the prominent words are got after segmenting, representing, and weighting words, which facilitates the following calculation including name entity extraction and sentiment orientation analysis.

4.3 Name entity filtration

Name entity filtration aims to find the 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 noises. Traditionally, two ways can be employed: set the minimum weight or set the qualified percentage. Which method should be chosen based on the specific text. More details are discussed in the experiments.

If no extra context is used while processing the sentence that is discussed in Fig. 2, all the words share the same weight. Prominent entities are shown in Fig. 4. Some of these entities may be related to specific sentiments. In the next section, one method is proposed to calculate sentiment orientation of words.

5 Sentiment analysis for prominent name entity

There are both objective and subjective sentences in text, such as the sentence "She says, because her lab is used to study lots of things" is one objective sentence while "MacPhee would be happy to eat it!" is one subjective sentence. More specifically, there are both kinds, objective and subjective, of components in subjective sentence. For instance, "MacPhee" and "happy" are different when expressing some sentiments.

5.1 Prominent sentiment carrier recognition

Generally speaking, some words with specific pos (part of speech), such as adjective element (tagged as ag), adjective (tagged as a), adverb element (tagged as dg), adverb (tagged as d), verb (tagged as verb), exclamation (tagged as e), and even some punctuation (tagged as w) can express various kinds of sentiment explicitly, instantly "happy" and "!" in the sentence we mentioned earlier.

In this paper, to recognize sentiment carrier, a small kernel set of 1066 terms are selected manually as sentiment seed based on [33]. There are six kinds of primary sentiments including love, joy, surprise, anger, sadness, and fear. Complex emotions are various combination of six primary sentiments. Furthermore, 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

"李昕/nr, 付希明/nr, 山东/ns, 日照市/ns, 五里坡/ns, 学院路/nz, 清华大学计算机学院/nt 李昕/nr "

Fig. 4 Prominent name entity extracted from text

the jth sentiment set. At the same time, one hypothesis that some words can express multiple sentiments simultaneously should be bear in mind always. Fuzzy concept [34] is employed here. In the process of determining the sentiment orientation, we mine the words in corpus which has similar function as these seeds. Moreover, s_{v_i,v_k^j} can be obtained as follows:

$$s_{v_i,v_k^j} = \frac{v_i * v_k^j}{|v_i||v_k^j|}, \tag{8}$$

where v_i is the word vector of w_i from Section 4.1. If $s_{v_i,v_k^j} > 0.7$, then we select v_i as sentiment carrier. But we cannot determine which kind of sentiment v_i expresses because some antonyms have similar word vectors. This happens when antonyms are used in the same place of sentence. Some of these words carry strong sentiment, while others carry weak sentiment, but all of them contribute to sentiment expression of name entity.

As microblog is a kind of short text, the sentiment of these short messages keeps unanimous at most situation. Then sentiment carrier which appear in the same microblog expresses similar sentiment. So we can calculate sentiment orientation based on this consistency. More details will be discussed in the experiments.

5.2 Building relationship between sentiment carrier and name entity

Relationship among words in various levels has been mined in some related work such as [35]. This paper focus on relationship between sentiment words and name entities. Sentiment expressed by carrier belongs to some entity, formulated as $w_i \in N(w_j)$, such as the word "happy" attributes to "MacPhee," namely "happy$''$ $\in N($"Macphee$''$), in the 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. In other words, co-occurrence dependency is mined, not grammar dependency. Qualifying sentiment words (such as adjectives) always decorate the name entities which exist in the same sentence. Also, coreference resolution problem is ignored in this paper. If some sentiment carriers and coreference appear in the sentence, we pass this sentence and move on.

If we build relationship for the case shown in Fig. 2, the result would be "li xin—kai xin". There is only one sentiment carrier "kai xin" coexist with "li xin" in the sentence. Another case was shown in Fig. 5. First, name entity and sentiment carrier are analyzed. Second, relation of these words are mined. Sentiment carrier for name entity "MacPhee" contains "scream," "perfect," "happy," and "laugh," namely. The sentiment orientation of these words implies the attitude of person "MacPhee" to some degree. In the next section, we will discuss how to calculate sentiment orientation in detail.

5.3 Sentiment orientation analysis for prominent name entity

Given some text, name entity, sentiment carrier, sentiment orientation of carrier, and relationship between name entity and sentiment carrier can be calculated by employing the method mentioned earlier. This section turns to calculate sentiment orientation for name entity and text.

5.4 Calculating sentiment orientation for name entity

Actually, word weight can be obtained by using the method introduced by Section 4.2, which means each sentiment carrier and name entity have their values. Furthermore, sentiment carrier contributes to the sentiment of name entity in different degrees. So here, we use the simple average weight to deal the problem of calculating sentiment orientation for name entity.

$$se_{ij} = \frac{\sum_k WS(w_k) * se_{kj}}{\sum_k WS(w_k)}, \qquad w_k \in Ne(w_i) \tag{9}$$

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 the element of sentiment carrier set for word w_i. In this way, the prominent sentiment of name entity depends on distinctive high weight sentiment carrier.

5.5 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. If the most weighted object is very "joy," then the sentiment of text tends to be "joy". In this way, we employ one way similar to the calculation of name entity to calculate sentiment orientation for text.

$$\begin{aligned} tse_{lj} &= \frac{\sum_i WS(w_i) * se_{ij}}{\sum_i WS(w_i)}, \\ &= \frac{\sum_i \sum_k WS(w_i) WS(w_k) * se_{kj}}{\sum_k WS(w_k) \sum_i WS(w_i)} \\ & w_k \in Ne(w_i) \\ & w_k \in Ne(te_l) \end{aligned} \tag{10}$$

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

6 Experiments

In this part, we will explain how these methods work in details. Three parts, prominent word and name entity

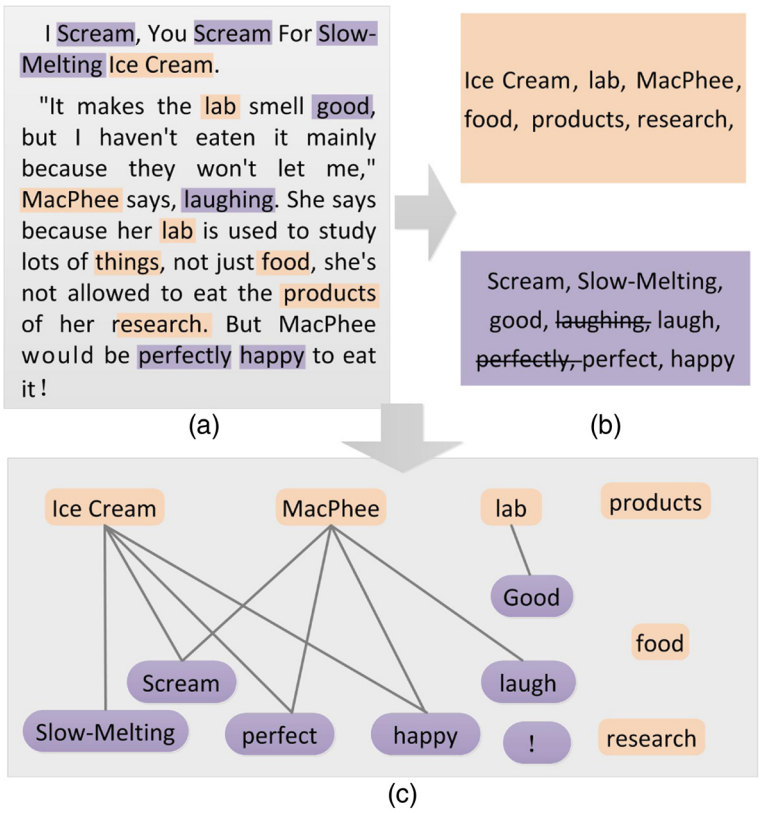


Fig. 5 Diagrammatic sketch of building relationship between sentiment carrier and name entity. **a** is the original text. **b** is the prominent information extraction. **c** is the dependence of sentiment to name entities

extraction, sentiment carriers recognition, and sentiment analysis for text, are included.

6.1 Prominent word and name entity extraction

To extract Chinese word, we use a dictionary, which are extracted from open online source, such as China Daily (2000.1–2000.3) and Sougou (pinyin.sogou.com/dict). Number of tokens are shown in Table 1. To do our work, one word dictionary (format: word, pos_1, fre_1, pos_2, fre_2, ..., pos_n, fre_n)[1], one bigram dictionary (format:$word_1$@$word_2$ fre)[1], and one pos transfer matrix table are used. Name entity dictionary (format:word, $role_1$, fre_1, $role_2$, fre_2, ..., $role_n$, fre_n)[2] and transfer matrix are employed in segment Chinese sentence. HMM model and viterbi algorithm are employed as described in Section 3.

The corpus we employed in this paper is crawled from "weibo.sina.com" with the keyword "rainstrom" in Chinese; 339,541 microblogs are found. "weibo.sina.com" is one of the most popular social media platform just like Twitter in the USA, which allows users to express opinions freely. As a result, there are large amount of noisy

Table 1 Dictionaries for prominent word and name entity extraction

Name of dictionary	Token number
Dictionary for segment	85,584
Dictionary of bigram	408,980
Dictionary for person name recognition	22,289
Dictionary for place name recognition	19,248
Dictionary for organization name recognition	19,289

Table 2 Corpus information when extracting word and name entity

Name of items	Token number
Number of microblog in corpus	339,541
Number of word in corpus	76,364
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	33,907
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

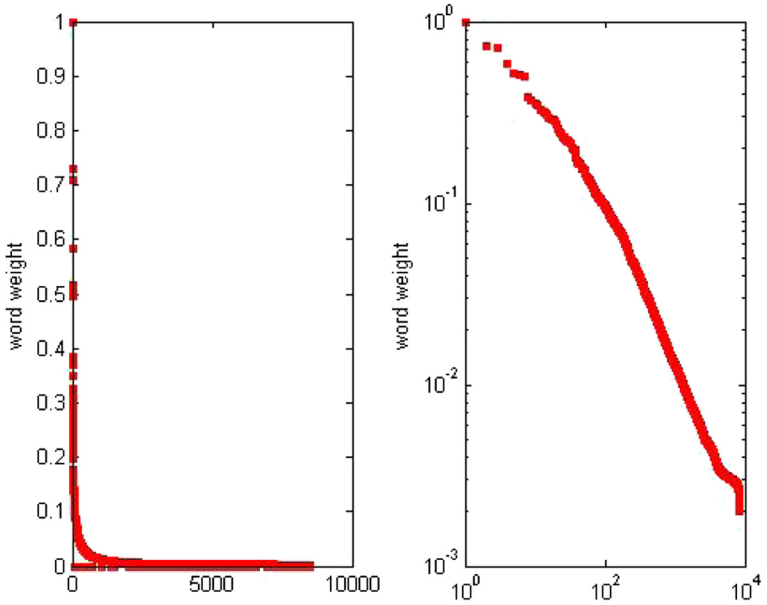


Fig. 6 Word weight distribution ordered by their weight. The second graph is transformed into log-log format

information mixed in the interesting content. Before we take steps to go in the future, 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 2. Precision, recall, and F1-value should be higher than 90 % according to [25]. Then, TextRank with window size five is employed to calculate word weight. The distribution of word weight is shown in Fig. 6 which follows the power distribution. That means only a small percentage of words are assigned with high weight; most of words carries low weight. From Fig. 7, weight distribution of name entities, such as person name (nr), organization name (nt), and place name (ns) can be found. Place name entities carry heavier weight compared with person name and organization name entities. Until now, words are sorted through their weight.

Name entities with high weight are left for future analysis. Some nouns such as "rainstorm" and "rain" which carries high weight are also kept for future analysis.

6.2 Extracting sentiment carriers for name entity

Name entities are mined using HMM. There are a large amount of adjectives around these name entities to express sentiments. But not all adjectives carry sentiment such as "latest" in the sentence "The latest news said that hazardous typhoon Rammasun would land tomorrow

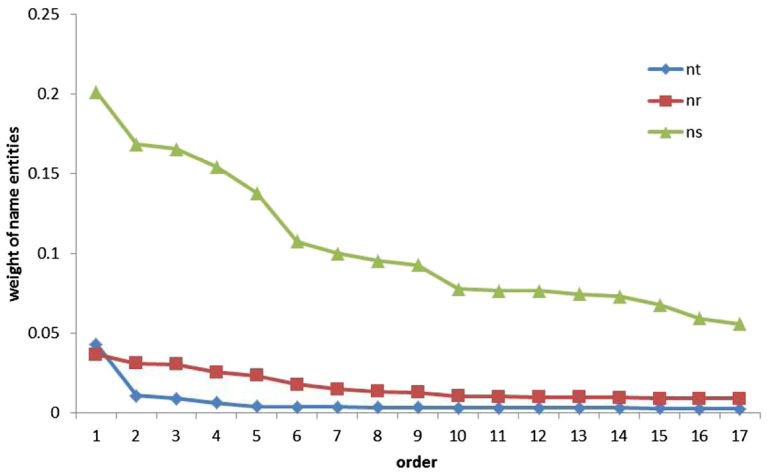


Fig. 7 Name entity weight distribution ordered by their weights. *nt* is the organization name. *nr* is the person name. *ns* is the place name

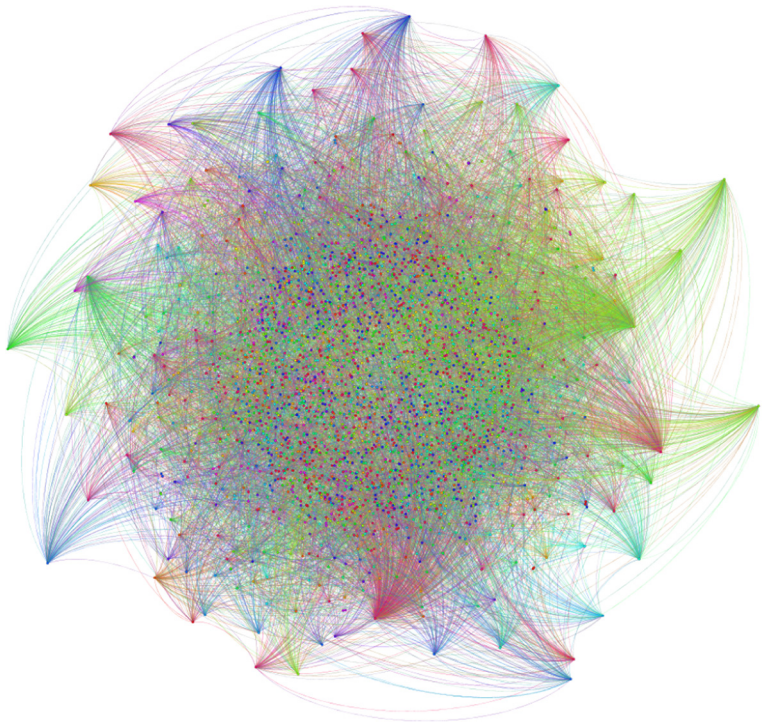


Fig. 8 Network of dependency between nouns and sentiment carriers. Sentiment carriers imply sentiment orientation of users for entities. Vertex denotes words, and edge represents dependency relations

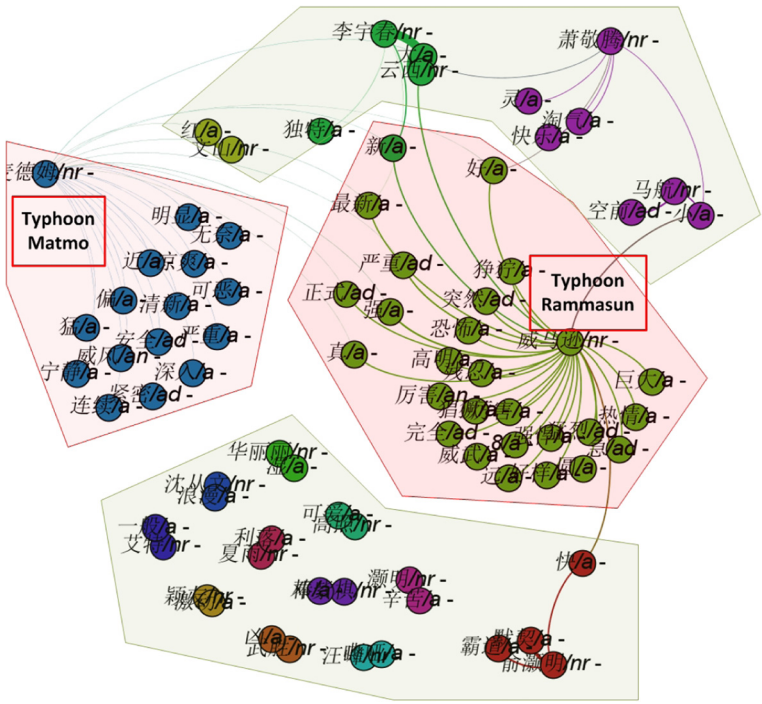
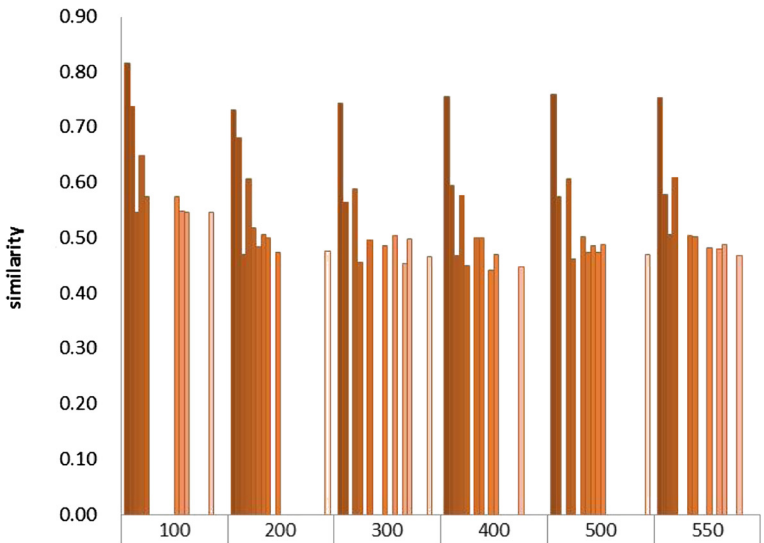


Fig. 9 Name entity (typhoon Matmo and Rammasun) and its sentiment carriers. Two centers can be found in this figure. There are also some other person name (suffixed by "/nr-") are shown with their sentiment carriers

morning". We will discuss how to distinguish these adjectives later. Nouns and adjectives construct some network, in which words are denoted by vertexes. Relations between words are edges, as shown in Fig. 8. The network is complex. However, only relations with high weight should be considered.

If we mine the network in detail, sub-network like Fig. 9 can be found. Two important name entities, typhoon Matmo and Rammasun, are highlighted in light red polygon. Some other name entities and their adjectives are shown in light green polygon. Connections between name entities and adjectives are also revealed. Adjectives are shared by different entities.

As is discussed in Section 4.1, 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" are quite similar. It is not that effective in finding synonyms when it turns to adjective. Distance between "beautiful" and "ugly" is small, as the two words 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 stays consistent in the same microblog if there is no negative words in the microblog,



	100	200	300	400	500	550
大暴雨	0.82	0.73	0.74	0.76	0.76	0.75
暴雨	0.74	0.68	0.57	0.60	0.58	0.58
倾盆大雨	0.55	0.47		0.47		0.51
大作/v	0.65	0.61	0.59	0.58	0.61	0.61
雷鸣	0.57	0.52	0.46	0.45	0.46	
电闪		0.48				
突然		0.51	0.50	0.50	0.50	0.51
萍城		0.50		0.50	0.47	0.50
四起					0.49	
紧接着		0.47	0.49	0.44	0.47	
昨晚	0.58			0.47	0.49	0.48
昨夜	0.55		0.51			
19:12分	0.55					0.48
大作/n			0.45			0.49
2:45			0.50			
day521				0.45		
7.14						0.47
听说	0.55					
纳莎			0.47			
刚才		0.48			0.47	

Fig. 10 Words with high similarity to the word "rainstorm"

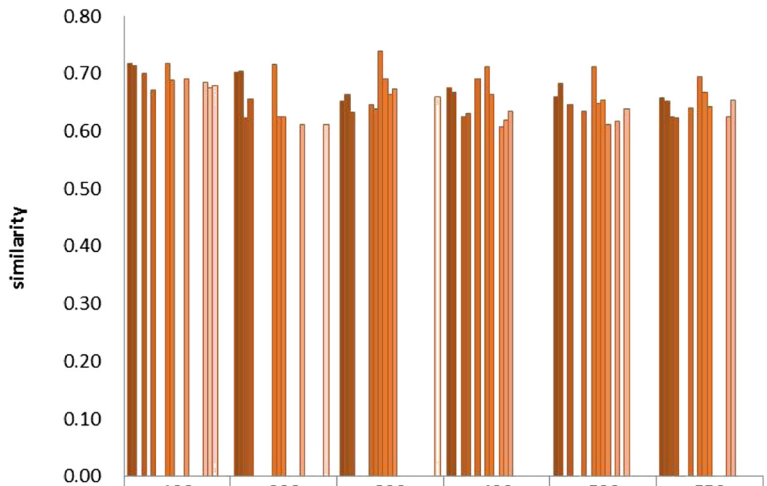
we compute the sentiment orientation of these sentiment carriers.

6.3 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. In this part, we cluster words with the same meaning together before calculating sentiment.

In the last subsection, we build the relation between entities and sentiment carriers; experiments are shown on sentiment analysis. Based on the word vector of words, we try to get similar nouns who have similar meanings. Figures 10 and 11 show two cases for words "rainstorm" and "Rammasun". The meaning of Chinese is explained in Table 3. We choose top ten most similar words for "rainstorm" and "Rammasun". In the figure, similar words are listed, with different dimensions of word vector from 100 to 550. As we have constrained the sequence of similar words, bars who clustered in the first half part are better. We can conclude that with dimensions above 400 including 400, the result stays stable and correct relatively.

By employing sentiment carriers for cluster of name entities, we can get the sentiment orientation of aspects for text. As shown in Table 4, "fear" is strong for



	100	200	300	400	500	550
Matmo	0.72	0.70	0.65	0.68	0.66	0.66
麦德姆	0.71	0.70	0.66	0.67	0.68	0.65
麦得姆		0.62	0.63			0.63
威马逊	0.70	0.66		0.63	0.65	0.62
威玛逊				0.63		
韦森特	0.67					
启德			0.65	0.69	0.63	0.64
浣熊			0.64			
超强	0.72	0.72	0.74	0.71	0.71	0.69
登录	0.69	0.62	0.69	0.66	0.65	0.67
重创		0.63	0.66		0.65	0.64
过境			0.67	0.61	0.61	
第十	0.69			0.62		
第九				0.63	0.62	
正面		0.61				0.62
8D					0.64	0.65
三级跳	0.69					
后移	0.68					
势如破竹	0.68					
41年		0.61				
保准/d			0.66			

Fig. 11 Words with high similarity to word "Rammasun"

Table 3 Maps of Chinese and English for Figs. 10 and 11

rainstorm (大暴雨,暴雨)	Matmo (Matmo, 麦德姆, 麦得姆)
downpour(倾盆大雨)	Rammasun (威马逊,威玛逊)
erupt (大作/v)	Vicente(韦森特)
thunder (雷鸣)	Kai-tak(启德)
light (电闪)	Neoguri (浣熊)
abrupt (突然)	ultrastrong (超强)
heavy rain (萍城)	landing(登录)
all together (四起)	hit(重创)
immediately (紧接着)	pass by(过境)
last night (昨晚,昨夜)	the tenth(第十)
time 19:12 (19:12分)	the ninth(第九)
script (大作/n)	frontage(正面)
time 2:45 (2:45)	8 dimension(8D)
21st May (Day521)	hop skip and jump(三级跳)
14th july (7.14)	backward(后移)
heard of (听说)	without difficulty(势如破竹)
Typhoon Nesat(纳莎)	41 years(41年)
Just now (刚才)	for sure(保准)

"rainstorm" and "typhoon". Slight joy with 0.24 is shown in text for "rainstorm". "joy" and "anger" both with 0.39 is shown in text for "typhoon". Sentiment "joy" originates 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 in the text.

The skills in this paper are efficient when used for sentiment analysis. Word features and word dependency features have been considered in the skills. Also, the skills can generate characteristic result for different corpus, which is in accordance with human cognition. However, the accuracy of the skills is somehow depending on the quality of dictionaries. Furthermore, word vector representation does not work well when it is used for finding similarities of adjectives. Word vector is more suitable for extracting hidden relations. These relations are affected by word meaning, word dependency, and some unknown elements.

7 Conclusions

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

1. Propose a framework of new technologies. A large amount of new technologies arises these years. Each of them has advantage and disadvantage. This paper combines 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 out that noun words with similar meaning can be clustered easily, but when it comes to adjective, it will not work that well. This is because, adjective with different meaning may share the same context. In other words, word vector is generated by fusing multinomial information.
3. Recognize 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 between sentiment carriers and name entities.

Table 4 Sentiment orientation for "rainstorm" and "typhoon"

	Love	Joy	Anger	Sad	Fear	Surprise
Rainstorm	0.13	0.24	0	0.09	0.91	0
Typhoon	0	0.39	0.39	0	0.71	0

Some future works are needed to complete, such as finding some 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.

Endnotes

[1] pos means part of speech; fre means frequency.

[2] Details of role can refer to [25].

Appendix

Typhoon Matmo (2014): Typhoon Matmo, known in the Philippines as Typhoon Henry, was the first tropical cyclone that directly impacted Taiwan in 2014. It was the tenth named storm and the fourth typhoon of the 2014 Pacific typhoon season. The typhoon is believed to be the main reason behind the crash of TransAsia Airways Flight 222, which occurred a day after it made landfall. More details can be seen through en.wikipedia.org/wiki/Typhoon_Matmo_(2014).

Typhoon Rammasun (2014): Typhoon Rammasun, known in the Philippines as Typhoon Glenda, was one of only two category 5 super typhoons on record in the South China Sea, with the other being Pamela in 1954. Rammasun had destructive impacts across the Philippines, South China, and Vietnam in July 2014. It was the seventh tropical cyclone of the season to be named by the Philippine Atmospheric, Geophysical and Astronomical Services Administration (PAGASA). Rammasun is a Siamese word for thunder god. More details can be seen through en.wikipedia.org/w/index.php?title=Typhoon_Rammasun&redirect=no.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This work is partially supported by the National Natural Science Foundation of China (Grant No. 91224008) and by the National Basic Research Program of China (973 Program No. 2012CB719705).

Authors' information
Xinzhi Wang is a PhD student in Tsinghua University. She received the Master degree and bachelor degree from School of Computer Engineering and Science of Shanghai University, Shanghai, China, in 2015 and 2012, respectively. Her main research field includes text analysis, topic detection and tracking, sentimental analysis, and web mining.
Hui Zhang is the vice director of Institute of Public Safety Research in Tsinghua University, Beijing, China. His recent researches most focused on public safety and emergency management, including preparedness and response to emergencies, decision-making support, applications of social media and big data in emergency management, etc. He received the National Science Foundation CAREER Award (USA) in 1999 when he was in SUNY Stony Brook, USA, and Changjiang Scholar (China) in 2008.
Shengcheng Yuan is a Ph.D. student of the Institute of Public Safety Research at Tsinghua University, People's Republic of China, and a visiting researcher of Center for Information Management, Integration and Connectivity (CIMIC) at Rutgers, the State University of New Jersey, USA. His research interests include agent-based traffic simulation, emergency evacuation, and decision support system on emergency management.
Jiayue Wang is a PhD student in Tsinghua University. Her main research directions are crowd dynamics and abnormal behavior detection technology based on CCTV (closed circuit television). She received the Master's degree from Tsinghua University, Beijing, China, in 2015.
Yang Zhou is a PhD student in Tsinghua University. His main research directions is integration testing of UAV. He received the Master's degree from Tsinghua University, Beijing, China, in 2015.

Received: 2 March 2016 Accepted: 25 June 2016
Published online: 11 July 2016

References
1. Z Xu, Y Liu, J Xuan, et al., Crowdsourcing based social media data analysis of urban emergency events. Multimed. Tools Appl., 1–18 (2015)
2. Z Xu, Y Liu, NY Yen, et al., Crowdsourcing based description of urban emergency events using social media big data. IEEE Trans. Cloud Comput, 1 (2016)
3. Z Xu, Y Liu, L Mei, et al., Semantic based representing and organizing surveillance big data using video structural description technology. J. Syst. Softw. **102**(C), 217–225 (2015)
4. Z Xu, L Mei, Y Liu, et al., Semantic enhanced cloud environment for surveillance data management using video structural description. Computing, 1–20 (2014)
5. SR Das, MY Chen, Yahoo! for Amazon: sentiment extraction from small talk on the Web. Manag. Sci. **53**, 1375–1388 (2007)
6. S Morinaga, K Yamanishi, K Tateishi, T Fukushima, in *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. Mining product reputations on the Web (Industry track, 2002), pp. 341–349
7. Y Rao, J Lei, W Liu, et al., Building emotional dictionary for sentiment analysis of online news. World Wide Web Internet Web Inform. Syst. **17**(4), 723–742 (2014)
8. X Ding, B Liu, PS Yu, in *International Conference on Web Search and Data Mining*. A holistic lexicon-based approach to opinion mining (ACM, 2008), pp. 231–240
9. C Hu, Z Xu, Y Liu, et al., Semantic link network-based model for organizing multimedia big data. IEEE Trans. Emerg. Top. Comput. **2**(3), 376–387 (2014)
10. L Polanyi, A Zaenen, in *Computing Attitude and Affect in Text: Theory and Applications*. Contextual Valence Shifter (Springer Netherlands, 2006), pp. 1–10
11. Z Xu, X Wei, X Luo, et al., Knowle: a semantic link network based system for organizing large scale online news events. Futur. Gener. Comput. Syst. **43–44**, 40–50 (2015)
12. V Hatzivassiloglou, KR Mckeown, in *Proceedings of the Acl*. Predicting the Semantic Orientation of Adjectives, (2002), pp. 174–181
13. L Liu, M Lei, H Wang, Combining domain-specific sentiment Lexicon with Hownet for Chinese sentiment analysis. J. Comput. **8**(4) (2013)
14. S Huang, Z Niu, C Shi, Automatic construction of domain-specific sentiment lexicon based on constrained label propagation. Knowl.-Based Syst. **56**(3), 191–200 (2014)
15. G Qiu, B Liu, J Bu, et al., in *Proceedings of the 21st international jont conference on Artifical intelligence*. Expanding domain sentiment lexicon through double propagation (Morgan Kaufmann Publishers Inc., 2009), pp. 1199–1204
16. G Ganapathibhotla, B Liu, in *International Conference on Computational Linguistics*. Identifying preferred entities in comparative sentences, (2012). To appear
17. E Breck, Y Choi, Cardie C, in *International Joint Conference on Artificial Intelligence*. Identifying expressions of opinion in context, (2007), pp. 2683–2688
18. M Zheng, Z Lei, X Liao, et al., Identify sentiment-objects from Chinese sentences based on cascaded conditional random fields. J. Chin. Inf. Process. **27**(3), 69–76 (2013)
19. Y Zhu, H Tian, J Ma, et al., *An integrated method for micro-blog subjective sentence identification based on three-way decisions and naive Bayes*. (Rough Sets and Knowledge TechnologySpringer International Publishing, 2014), pp. 844–855
20. B Pang, L Lee, Opinion mining and sentiment analysis. Found. Trends Inform. Retr. **2**(1), 459–526 (2008)
21. Y Rao, J Lei, W Liu, et al., Building emotional dictionary for sentiment analysis of online news. World Wide Web Internet Web Inform. Syst. **17**(4), 723–742 (2014)
22. M Karamibekr, AA Ghorbani, *Lexical-syntactical patterns for subjectivity analysis of social issues*. (Active Media TechnologySpringer International Publishing, 2013), pp. 241–250
23. T Wilson, J Wiebe, R Hwa, in *Proceedings of AAAI*. Just how mad are you? Finding strong and weak opinion clauses, (2004), pp. 101–109
24. T Wilson, J Wiebe, P Hoffmann, in *Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP)*. Recognizing contextual polarity in phraselevel sentiment analysis, (2005), pp. 347–34
25. HK Yu, HP Zhang, Q Liu, et al., Chinese named entity recognition based on cascaded hidden Markov model. J. Commun. **2**, 87–94 (2006)
26. G Salton, A Wong, CS Yang, A vector space model for automatic indexing. Commun. ACM. **18**(11), 613–620 (1975)
27. MI Jordan, DM Blei, AY Ng, Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 465–473 (2003)
28. J Pennington, R Socher, CD Manning, in *Empirical Methods in Natural Language Processing*. Glove: global vectors for word representation, (2014)
29. EH Huang, R Socher, CD Manning, et al., in *Proc. Meeting of the Association for Computational Linguistics: Long Papers (ACL), vol. 1*. Improving word representations via global context and multiple word prototypes, (2012), pp. 873–882
30. G Salton, C Buckley, Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. Int. J. **24**(5), 513–523 (1988)
31. HC Wu, RWP Luk, KF Wong, et al., Interpreting TF-IDF term weights as making relevance decisions. Acm Trans. Inf. Syst. **26**(3), 55–59 (2008)
32. R Mihalcea, P Tarau, TextRank: bringing order into texts. Unt Scholarly Works. (2004), pp. 404–411
33. W Parrott, *Emotions in social psychology: essential readings*. (Psychology Press, 2001)
34. J Xie, C Liu, *Fuzzy mathematics method and application*. (Huazhong university of science and technology press, 2000)
35. X Luo, Z Xu, J Yu, et al., Building association link network for semantic link on Web resources. IEEE Trans. Autom. Sci. Eng. **8**(3), 482–494 (2011)