Trending Sentiment-Topic Detection on Twitter

Baolin Peng[1,2], Jing Li[1,2], Junwen Chen[1,2], Xu Han[1,3], Ruifeng Xu[4], and Kam-Fai Wong[1,2,3]

[1] Dept. of Systems Engineering & Engineering Management, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[2] MoE Key Laboratory of High Confidence Software Technologies, China
[3] Shenzhen Research Institute, The Chinese University of Hong Kong, Hong Kong
[4] Harbin Institute of Technology, Shenzhen Graduate School, China
{blpeng,lijing,jwchen,kfwong}@se.cuhk.edu.hk, xuruifeng@hitsz.edu.cn

Abstract. Twitter plays a significant role in information diffusion and has evolved to an important information resource as well as news feed. People wonder and care about what is happening on Twitter and what news it is bringing to us every moment. However, with huge amount of data, it is impossible to tell what topic is trending on time manually, which makes real-time topic detection attractive and significant. Furthermore, Twitter provides a platform of opinion sharing and sentiment expression for events, news, products etc. Users intend to tell what they are really thinking about on Twitter thus makes Twitter a valuable source of opinions. Nevertheless, most works about trending topic detection fail to take sentiment into consideration. This work is based on a non-parametric supervised real-time trending topic detection model with sentimental feature. Experiment shows our model successfully detects trending sentimental topic in the shortest time. After a combination of multiple features, e.g. tweet volume and user volume, it demonstrates impressive effectiveness with 82.3% recall and surpasses all the competitors.

Keywords: Twitter, Online social network, Trending topic detection, Sentiment analysis.

1 Introduction

Twitter, one of the most popular microblogging websites in the world, is regarded more as a new type of information source as well as news feed than an online communication platform[9]. It not only brings strong influence to our daily life, but also servers as a mirror reflecting real-life-events. On Twitter, people are now more concerned about what is happening on the world instead of what their friends or family are doing. Is Justin Bieber having a new girlfriend? How terrible the damage brought by Typhoon Haiyan to Philippines? Did Portugal or Sweden win the qualification for World Cup finals? A case study given by Sakaki[19] shows that the Twitter can act as a bursting event monitor and enable the information propagates even faster than other news media.

A. Gelbukh (Ed.): CICLing 2015, Part II, LNCS 9042, pp. 66–77, 2015.
DOI: 10.1007/978-3-319-18117-2_5

With more than 500 million registered users posting 340 million tweets per day[1], it is impossible to detect big news or important events in time manually. Additionally, Twitter provides a platform for opinion sharing and sentiment expression. On Twitter, people tends to express their true feelings and tells what they are really thinking about. One of the reasons why Twitter is regarded as valuable data resource is that it contains hundreds and thousands of opinions including discussions about social events, praises or complains about products, and so on[1]. By knowing how sentiment changes from time to time, we will have a better understanding about the evolution of topics, thus helps us determine whether a topic is trending or not. However, little real-time models for trending topic detection has taken sentiment features into consideration.

As such, we propose a real-time non-parametric trending topic detection model combining sentiment analysis methods. In experiment, the topics detected by our model are compared to the topics selected as trending topics by Twitter's own system. And it demonstrates that our system is able to detect trending topics using less time compared to current state-of-the-art real-time trending topic detection systems. Moreover, the topics detected by our model has stronger sentiment, some of which has brought strong influence to society although they maybe ignored by Twitter company at that time. Furthermore, we make a combination of multiple features and yield a compound model, which achieves highest effectiveness.

To sum up, the main contribution of this paper are mainly two folds :

- Our proposed approach utilizes time-varied sentiment to enhance the state-of-the-art real-time trending topic detection model.
- We propose a novel compound model combining multiple features, i.e. tweet volumes, sentiment and user volumes, which obtains best effectiveness and quick response.

2 Related Work

As to topic detection, topic model is one of the most well-known method. Diao et al.[5] processes a LDA model and made some adaptations for the diversity and noise in microblog. Gao et al. [6] proposed an incremental Gibbs sampling algorithm based on HDP to incrementally derive and rene the labels of clusters, thus helps to reveal the connections between topics. Though these works claims to have amazing result compared to their baseline topic model, they are not suitable to deal with streaming data thus can not be applied for real-time topic detection task. For real-time topic detection, there are three kinds of approaches[13]. The most popular one is to analyze the deviation of topics' activity relative to some baseline. Twitter monitor [12] cluster words to form topics and combine features like temporal features and social authority as an accurate description of each topic, adding to user interaction they successfully detect real-time topics over Twitter Streams. Similar methods are also given by Becker et al.[3] and Cataldi et al.[4].

[1] http://techcrunch.com/2012/07/30/analyst-twitter-passed-500m-users-in-june-2012-140m-of-them-in-us-jakarta-biggest-tweeting-city/

Nikolov[13] proposed a model based on time series classification and argue that from pattern of tweets with respect to timeline, we could determine which topic is trending. His work is attractive and claimed to be state-of-the-art but only takes tweets' amount variation into consideration and ignores the importance of sentiment. Whereas, Twitter is a platform for sentiment expression and opinion sharing, sentiment analysis and opinion mining draws particular attention in Twitter analysis. Among all the research topics in sentiment analysis, sentiment classification is perhaps most extensively studied[16] one. Its goal is to classify a subjective document as positive or negative with the help of some machine learning methods. Classical sentiment classification works mainly focus on two aspects.

One is to refine machine learning approaches and apply them to classify documents. Pang and Lee[17] are pioneers to use these classification algorithms to classify sentiment of movie reviews. In their work, Naive Bayes, Maximum Entropy model and Support Vector Machine (SVM) are applied to determine polarity of reviews as positive or negative. Back to the year of 1999, Wiebe et al.[23] came up with a method using Naive Bayes to determine whether a document is subjective or objective. Inspired by this, Pang and Lee[15] used a hierarchical classification model that treated subjectivity classification as the first step of sentiment analysis before polarity classification. Subjectivity classification with no doubt make sense based on the assumption that objective sentence imply little opinion though it maybe not correct under some special circumstances.

Another important aspect is feature selection. Six types of features are summarized as the most common features appeared in previous works [11]. Features based on terms and their frequencies are the most simple kind of feature but shown in Pang and Lee[17] to be effective using Naive Bayes and SVM as sentiment classifier. Part of speech of each word is also of importance, which is already proved effective in sentiment tagging task by Riloff et al.[18] and Wiebe et al.[22]. Sentiment words and phrases expressing positive or negative sentiment, are regarded as efficient sentiment indicators. Specifically, adjectives, adverbs, verbs and nouns are probable to be considered as sentiment words. Sentiment shifters can change sentiment orientations so that they should not be ignored.

3 Methodology

In this section, we describe how our model detect real-time sentimental topics. Firstly, sentiment scores indicating both polarity and intensity are given to tweets based on SVM sentiment classifier. Furthermore, sentimental temporal series are determined by estimating distribution of sentiment scores. Then we build a real-time trending topic detection model with sentimental features. A coarse-grained problem definition is given in the following part in Section 3.1

3.1 Problem Definition

Discovering topics from the massive social network information is a worthy task when we faces the big data scenario in social computing domain. But we still want to focus on a more specific scenario, which is sentiment sensitive and dynamic.

Taking a further considering, besides the newest events, we are arguing that the topic received heavier sentiment from users would be more tend to become a trending one. The intuition is directed, think about our everyday life, even though some news might get large volume of mentioned for it is formality (like a government document get published), but it is hardly to say it would be a hot topic if nobody have strong emotion or sentiment on it; Conversely, most of the trending topic is the exciting or controversial ones.

Taking such issues into account, and concerning the data gathering, we decide to define our problem on the Twitter platform, as Twitter provide sufficient API to manipulate the data, as well as an official streaming API, which would be stated in detail in Section 3.2

3.2 Data Collection and Preprocessing

Twitter Streaming API has accessing rate limitations for common developers, so it is impossible to reach a relative higher sampling rate. The sampling rate from the API is guaranteed to be no lower than 1% of the whole server streams according to Twitter documentation. We collected the raw tweets from Streaming API through more than 20 days, and around 10GB data was returned in every 24-hour. We totally gained more than nine hundreds of millions tweets, which are divided uniformly into three dataset for training, developing and testing respectively [2].

The trending hashtags in each hour are also provided by Twitter API, we collected them within the same period with the streaming data, and totally more than 1400 hot, or trending hashtags are retrieved. Each hashtag has a timestamp to indicate in which hour of which day it was trending.

Our goal is to decide whether a topic is trending or not. Generally speaking, the traditional way to defining a topic or event in social computing and information retrieval domain is using the bag-of-word model, which is the most simple but also most effective one. It defines the topic as a set of keywords, most of which is verb or noun, without considering the ordering of them, and practically this model could roughly depict an event already. On the other hand, Classical topic models such as LDA are not applicable here since we aim at detecting real-time topics from streaming data.

In this work, we regard hashtags consist of a word or a phrase with a hash symbol as a corse-grained topic, e.g. #londonriot, #TwitterParty,#nowplaying. Hashtags are created by Twitter users as a way to categorize tweets thus illustrates the connection among the topic. For one thing, hashtags are popular among Twitter users. Wang et al.[20] measured on a dataset with around 0.6 million randomly selected tweets and found that around 14.6% tweets have at least one hashtag. For another, real-time trending topics can be detected via trend analysis of hashtags and avoid delay brought by topic detection processing like clustering. In addition, we are not so care how the topic is described but how to find a trending one. Thus we would only choose a hashtag in Twitter to be a

[2] All the codes and data will be published and uploaded to google drive after reviewing.

topic, instead of using bag-of-word, and this setting could erase the complication on clustering keywords into a single topic.

Trending hashtags were selected from the hot topics received from Twitter as training set. We filtered out the hashtags existing for too short or too long period (less than 4 hours or longer than 24 hours), as these kinds of topics usually brought many noise and might be unstable along the time line. Also, to limit the topics lasting period within 24 hours is to avoid the periodic patterns. For the non-trending hashtags which is testing set, we would just randomly pick from the tags which were not existing in the trending list, and lasting for enough time (at least 4 hours) meanwhile. The time series would be generated with respect to each of these filtered hashtags.

To accelerate hashtag extraction and analysis, MapReduce is applied here. Each hashtag could be a key in MapReduce framework. The combination of tweet list relevant to a topic could be referred to the Reduce procedure. So we conclude that our preprocessing could be perfectly fit in the MapReduce framework. For each tweet, the mapper produces a pair of hashtag and twitter ID and the reducer combines twitter IDs relevant to each hashtag into a list.

As sentiment analysis is an important step in our work, NLP related preprocessing work such as tokenization, lemmatization, and part-of-speech annotation is required, which is implemented by Stanford Core NLP[3].

3.3 Sentimental Time Series

Before sentimental topic detection, we give sentiment scores to each tweet to indicate both the polarity and sentiment intensity. Sentiment analysis in this paper is based on SVM classifier with unigram features. According to Pang and Lee[15], unigram feature model is simple but has a good performance and when fed to SVM classifier, it achieves impressive performance and surpasses all other competitors[7]. A bit tweet set with 1600000 subjective tweets labeled by emoticons as positive or negative release by Go. et al. [7] is utilized as training data.

In many previous works, sentiment words are chosen as features in sentiment classification. However, most of these works are based on subjectivity lexicon, such as MPQA[4]. Unfortunately, subjective words in these lexicons hardly appear in tweets because Twitter users tend to use very casual language. So feature words are selected from training set, a huge corpus of tweets with emoticons, most of which can somehow regarded as subjective microblogs.

However, such a huge training set contains 794876 words and phrase but most of them are noise for sentiment analysis. According to Go et al.[7], usernames, links and repeated letters can be eliminated with the help of regular expression, thus shrinks the feature set down to 45.85%. Though they perform a good reduction to feature space, there are some other properties we can take advantage of for further feature reduction.

[3] http://nlp.stanford.edu/downloads/corenlp.shtml
[4] http://mpqa.cs.pitt.edu

Part-of-speech (POS) is considered of big significance in sentiment classification. Nouns, verbs, adjectives and adverbs are all probable sentiment indicators. Barboca and Feng[2] also mentioned top 5 features as positive polarity, negative polarity, verbs, good emoticons and upper cases based on training data. So we only keep emoticons and upper cases as well as words annotated as noun, verb, adjective or adverb.

Furthermore, word frequency in different polarity helps to evaluate its subjectivity. For example, words like happy, great, wonderful, will appear a lot more frequently in positive tweets. It also acts as a polarity indicator for Naive Bayes. In the last step of feature selection, we filter out words equally appearing in both polarity determined by the following indicator:

$$i(f) = \begin{cases} 1 & \text{if } 1\text{-}\theta < P(f|POS) < \theta \\ 0 & \text{otherwise} \end{cases}$$

and a feature f will be removed if $i(f) = 1$.With all aforementioned steps, we finally shrink features down to 8.65% of the original size.

Moreover, we'd like to define *sentiment score* in Definition 1 to illustrate both the polarity and sentiment intensity of a tweet.

Definition 1. *Sentiment score is a real value in [-1,1]: negative score indicates negative polarity while positive score denotes positive polarity. And a larger absolute value means higher degree of sentiment.*

Furthermore, SVM classifier not only determine the polarity but also helps to evaluate how subjective a tweet is. In fact, the farther the distance between tweet vector and SVM hyperplane, the more words as positive or negative sentiment indicators it contains, thus it is more subjective, or in other words, it expresses stronger sentiment. As a result, we use the score returned by $SVMlight$[8] as a sentiment score.

Specifically, we should point out that sentiment scores are relative instead of absolute. In other words, it only makes sense when sentiment of different tweet are comparing with each other. In fact, it is impossible to define exactly how strong the sentiment of a tweet is but sentiment comparison is much easier. For example, different person will have different evaluation in sentiment of the tweet 'I've done a good job'. But for tweet 'I've done a good job ' and 'I've done a great job', almost everyone will agree with the statement that the latter one has stronger positive sentiment than the previous one. As a result, we scale our sentiment scores into the interval of $[-1, 1]$ so as to fit Definition 1.

With sentiment scores given to Twitter topics, sentimental time series can be thus given. Considering that sentiment scores are relative values, a few steps mush be made in case of sentiment score's spiky or bursty distribution. Firstly, Histogram Equalization, which is a well-known method in image processing to adjust contrast using image's histogram, is applied for sentiment score smoothing due to the property of sentiment relativity. This method usually gains a higher global contrast particularly when the distribution of sentiment score is represented by close contrast values. Though it may reduce local contrast, through this adjustment, the intensities can be better distributed on the histogram and it effectively spreads out the most frequent intensity values thus increase the global contrast.

In order to calculate the temporal sentiment series, we propose a method to give a score to each time window based on the sentiment scores of tweets. First of all, we estimate the sentiment distribution in each time window by Parzen Window method with Gaussian Kernel given in the following:

$$p_n(s) = \frac{1}{nh_n} \sum_{i=1}^{n} K(\frac{s - s_i}{n}) \tag{1}$$

where $K(x) = \frac{1}{\sqrt{2\pi}} exp^{-\frac{x^2}{2}}$, n is the number of tweets, s_i indicate the sentiment score given to the i-th tweet in this time window, and h_n is the window width for sentiment score. Parzen window is non-parametric way to estimate the probability density function of a random variable inspired by histogram estimation method. And Gaussian Kernel is fundamental in data smoothing problem where inferences about the population are made, based on a finite data sample.

As a result, by calculate the expectation of sentiment in a time window, i.e. $e(t_i) = \int_{-1}^{1} sp_n(s)ds$ and we give a value $e(t_i)$ to time window t_i thus get a sequence $e(t_1), e(t_2), \cdots, e(t_n)$ as sentimental time series of a hashtag.

3.4 Real-Time Trending Topic Detection

The original time series is sharp due to rapid change on Twitter as shown in the left figure in Figure 1. In order to lower down noises and cover higher order, i.e. second order information, we applied the normalizing method mentioned in [13] to time series before Trending Topic Detection, then we can smooth the spiky time series as shown in the right figure in Figure 1. Specifically speaking, assume the $s[n]$ is the value of time series s at the index (time stamp) n and the method consists of the following steps:

- Baseline normalization: $s[n] = \frac{s[n]}{\sum_i^N s[i]}$, N is the total length of the series
- Spike normalization: $s[n] = |s[n] - s[n-1]|^\alpha$.
- Smoothing: $s[n] = \sum_(m = n - N_smooth + 1)^n s[m]$, where the N_smooth is a parameter that controlling how smooth we want.
- Logarithmic: $s[n] = logs[n]$.

Then we use a supervised model proposed by Nikolov[13] with sentimental time series to detect trending sentimental topics.

In the model, we need to label some positive and negative instances as indicating signals. Positive instances are those hashtags known as trending ones. We use Twitter API to retrieve the trending hashtags that are suggested by Twitter system, which are partially labeled by people from Twitter, and apply them as positive sample of trending topic. In addition, some negative instances, i.e. non-trending hashtags, are also required. In addition, we randomly select some

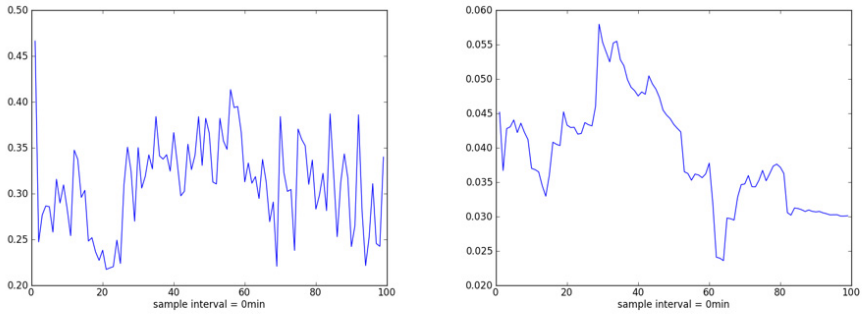

Fig. 1. Time Series Before and After Normalization

not trending hashtags to be negative samples. With positive set denoted by R_+ and negative set indicated as R_-, we are able to calculate the voting of each of these signals for an input time-series as a evaluation for trending topics:

$$R(s) = \frac{\sum_{c \in R_+} exp(-r \cdot d(s,c))}{\sum_{c \in R_-} exp(-r \cdot d(s,c))} \tag{2}$$

where r is a parameter that used to constrain system's sensitivity and $d(s_1, s_2)$ defines distance between two time series s_1 and s_2, and specifically speaking the Euclidean distance defined as following:

$$d(s_1, s_2) = \sqrt{\sum_i (s_{1i} - s_{2i})^2} \tag{3}$$

Then the value of $R(s)$ is used to determine which class the time series S belongs to, thus can tell whether a hashtag time series is trending or not.

4 Experiments

We tested our system on the test set with streaming data as previous mentioned, and would evaluate the results with two aspects: The trending prediction effectiveness and the trending detection time.

4.1 Effectiveness Evaluation

To evaluate the effectiveness of our system, we apply the (precision, recall, F-score) schema. We point out that in the context of topic detection, more emphasize should be put on recall rather than precision. As even if the system yields many actually non-trending topics, people don't want to miss the ones that are really trending. So we expect our system to guarantee high recall, which is just the case in the later experiments.

The test set contains around 400 trending topics and 300 non-trending topics. They are passed through the candidate systems, and the predicting results would be used to compare with the true values. Besides trending topic detection model with sentimental feature denoted by S-Model, we also implement model with other kinds of features like tweet volume (V-Model), which is implemented in the work of Nikolov[13] and user volume (UV-Model). Moreover, we make a linear combination of various features given in equation 4 as a compound one and integrate it into our model. Specifically in the equation below, $R_v(s)$, $R_s(s)$ and $R_u(s)$ are voting score of time series s given by our model with tweet V-Model, S-Model and UV-Model respectively.

$$R_v(s) = w_1 R_v(s) + w_2 R_s(s) + w_3 Ru(s) \tag{4}$$

We evaluate each model with precision, recall and F-score and furthermore compare it with baseline Bursting Model, i.e. Twitter Monitor[12]. The evaluating results are shown in Table 1.

Table 1. Effectiveness Evaluation of Different Models

Model	Precision	Recall	F-score
Bursting Model	60.5%	48.1%	53.6%
V-Model	65.3%	80.1%	71.9%
S-Model	57.5%	63.7%	60.4%
V+UV-Model	63.5%	75.1%	68.8%
V+S-Model	65.2%	**82.3%**	**73.3%**
V+U+S-Model	**68.3%**	78.1%	72.9%

Bursting model's performance just reaches the basic requirement line, but one thing interesting is even though the recall is low in this model, the precision is relatively good enough comparing with other models. The intuitive interpretation is that in the bursting model is stricter on judging whether a time series is trending or not, due to the bursting degree might not be spike enough to trigger the model for many actually trending time series.

Obviously, the volume feature is the most indicative features among all. Even with the classification model trained only on it, the recall is very high, just no more than 3 percentages lower than the highest recall among all. One could interpret this as the trending of time series are generally regarding the volume variation as indicator. This is also why some classic topic detection models, e.g. the bursting model, would take tweet volume series as their inputs.

The overall performance of our sentimental model is better than the baseline, with a relative higher recall, which is what we expected. We measured the hashtag set detected only by S-Model and only by V-Model with some sampled hashtags shown in table 2.

We can see that our sentimental model will pay more attention to topics with strong sentiment expression although they may not be selected as trending topics by Twitter company. For example, #TellAFeministThankYou is started by Melissa McEwan of Shakesville with the purpose to response to harassment

Table 2. Comparison of Sampled hashtags detected by V-Model and S-Model

V-Model	S-Model
#LoQueMasDeseoEs	#HappyBirthdayHarryFromLatinas
#mbv	#WaysToPissOffYourValentine
#NXZEROnoEncontro	#giornatadellamemoria
#EresLittleMonsterSi	#TuCaraMeSuena15
#PraSempreNossoEncantoPF	#WeWantMarcoGopezFor5Minutes
#RANHariBaru	#10TheBestMoviesEver
#QueremosBandaCineNoEncontro	#TellAFeministThankYou
#MJ50	#NationalSigningDay

of feminists on Twitter. It stroke American society and there were many news reports talking about this event.

When we combine multiple features together, interesting things happen. Even though user volume should have indicating function, when it was combined with the volume features the system performance even dropped; this might due to the correlation between user and tweet volume is relatively high, and when they are applied together even more noise would be created. Nevertheless, the compound model with volume and sentiment features, yields the highest F-score among all models, when the precision almost stay the same with only V-model. Yet the highest precision is achieved not by V-only or V-S model, but the compound-all model, namely applied all the V, S and U. So in general, from the overall aspect, one could expect the V-S model to be the best system, unless higher precision is required.

4.2 Detecting Time Evaluation

Since our model is a real-time model dealing with streaming data, how fast a trending topic is detected plays an important role. We also make an evaluation on the time trending topics are detected, which is illustrated by *detecting time* and defined as the following:

Since detecting time may vary from topic to topic so we calculate the average detecting time of different models and the results are shown in Table 3

What we can conclude from above table is that, the baseline, bursting model make the decision even after the Twitter company reports the trending topics. And the model our system based on can successfully detect trending topics before the report of Twitter company with whatever kind of feature. And among all kinds of features, V+S+U feature acts most slowly. And V+S-Model, which performs the best in the evaluation of effect has a not-bad performance, even faster than V-Model.

The model with the best detecting time is our S-Model. We interpret it as the sentiment information is more fierce at the beginning, and also have more stable patterns, which might give the system more confidence when it tries to give prediction when the topics start to spread. Also this property affects the performance of V+S-Mode which has better detecting time than V-Model.

Table 3. Detecting Time Evaluation of Different Models

Model	Detecting Time
Bursting Model	10.5 min
V-Model	-18 min
S-Model	**-33 min**
V+U-Model	-15 min
V+S-Model	-22 min
V+U+S-Model	-5 min

5 Conclusions

In this paper, we propose a real-time non-parametric model to detect trending topic. This work is based on a state-of-the-art supervised trending topic detection model and takes sentiment variation into consideration. In experiments, our model has the fastest response while making a descent decision. Although it does not have the best performance in effectiveness evaluation, i.e. precision, recall and F-Score, it successfully detect topics with stronger sentiment. Moreover, by a linear combination of multiple features, especially the tweet volume feature and sentimental feature, we get a model with best effectiveness and quick response.

However, there exists other kinds of features that are likely to make a contribution to the dynamic models. In addition, we could define different kinds of topics in the future, rather than the traditional concept which majorly based on the keyword bursting intuition. Moreover, the dynamic model, i.e. time series classification model might be applied to other social computing domains[14], e.g. rumour detection[10,21], along with a compound model taking several dynamic features into account at the same time.

Acknowledgement. This research is partially supported by General Research Fund of Hong Kong (417112), Shenzhen Fundamental Research Program (JCYJ20130401172046450, JCYJ20120613152557576), RGC Direct Grant (417613), National Natural Science Foundation of China No. 61370165 and Shenzhen Peacock Plan Research Grant KQCX20140521144507925.

References

1. Agarwal, A., Xie, B., Vovsha, I., Rambow, O., Passonneau, R.: Sentiment analysis of twitter data. In: LSM, pp. 30–38. Association for Computational Linguistics (2011)
2. Barbosa, L., Feng, J.: Robust sentiment detection on twitter from biased and noisy data. In: COLING(Poster), pp. 36–44. Association for Computational Linguistics (2010)
3. Becker, H., Naaman, M., Gravano, L.: Beyond trending topics: Real-world event identification on twitter. In: ICWSM (2011)
4. Cataldi, M., Di Caro, L., Schifanella, C.: Emerging topic detection on twitter based on temporal and social terms evaluation. In: MDMKDD, p. 4. ACM (2010)

5. Diao, Q., Jiang, J., Zhu, F., Lim, E.P.: Finding bursty topics from microblogs. In: ACL, pp. 536–544. Association for Computational Linguistics (2012)
6. Gao, Z.J., Song, Y., Liu, S., Wang, H., Wei, H., Chen, Y., Cui, W.: Tracking and connecting topics via incremental hierarchical dirichlet processes. In: ICDM, pp. 1056–1061. IEEE (2011)
7. Go, A., Bhayani, R., Huang, L.: Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, pp. 1–12 (2009)
8. Joachims, T.: Svmlight: Support vector machine. SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund 19(4) (1999)
9. Kwak, H., Lee, C., Park, H., Moon, S.: What is twitter, a social network or a news media? In: WWW, pp. 591–600. ACM (2010)
10. Li, B., Zhou, L., Wei, Z., Wong, K., Xu, R., Xia, Y.: Web information mining and decision support platform for the modern service industry. In: ACL, pp. 97–102 (2014), http://aclweb.org/anthology/P/P14/P14-5017.pdf
11. Liu, B.: Sentiment analysis and opinion mining. Synthesis Lectures on Human Language Technologies 5(1), 1–167 (2012)
12. Mathioudakis, M., Koudas, N.: Twittermonitor: Trend detection over the twitter stream. In: SIGMOD, pp. 1155–1158. ACM (2010)
13. Nikolov, S.: Trend or No Trend: A Novel Nonparametric Method for Classifying Time Series. Ph.D. thesis, Massachusetts Institute of Technology (2012)
14. Ou, G., Chen, W., Wang, T., Wei, Z., Li, B., Yang, D., Wong, K.: Exploiting community emotion for microblog event detection. In: EMNLP, pp. 1159–1168 (2014)
15. Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: ACL, p. 271. ACL (2004)
16. Pang, B., Lee, L.: Using very simple statistics for review search: An exploration. In: COLING (Posters), pp. 75–78 (2008)
17. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: Sentiment classification using machine learning techniques. In: ACL, pp. 79–86. Association for Computational Linguistics (2002)
18. Riloff, E., Wiebe, J., Wilson, T.: Learning subjective nouns using extraction pattern bootstrapping. In: HLT-NAACL, pp. 25–32. Association for Computational Linguistics (2003)
19. Sakaki, T., Okazaki, M., Matsuo, Y.: Earthquake shakes twitter users: Real-time event detection by social sensors. In: WWW, pp. 851–860. ACM (2010)
20. Wang, X., Wei, F., Liu, X., Zhou, M., Zhang, M.: Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In: CIKM, pp. 1031–1040. ACM (2011)
21. Wei, Z., Chen, J., Gao, W., Li, B., Zhou, L., He, Y., Wong, K.: An empirical study on uncertainty identification in social media context. In: ACL, pp. 58–62 (2013)
22. Wiebe, J., Riloff, E.: Creating subjective and objective sentence classifiers from unannotated texts. In: Gelbukh, A. (ed.) CICLing 2005. LNCS, vol. 3406, pp. 486–497. Springer, Heidelberg (2005)
23. Wiebe, J.M., Bruce, R.F., O'Hara, T.P.: Development and use of a gold-standard data set for subjectivity classifications. In: ACL, pp. 246–253. Association for Computational Linguistics (1999)