Subtopic-level Sentiment Analysis of Emergencies

Kunmei Wen, Zhijiang Liu, Shuai Xu,
Ruixuan Li, Yuhua Li, Xiwu Gu, and Jie Zan

Huazhong University of Science and Technology, Wuhan, China
{kmwen,rxli,idcliyuhua,guxiwu}@hust.edu.cn
{liuzhijiang_123,xushuaixu_shuai}@126.com,zanjie@outlook.com

Abstract. With the rapid development of microblog, millions of Internet users share their opinions on different aspects of daily life. By analyzing and monitoring sentiment information extracting from tweets related to an important event, we are able to gain insights into variation trends of users' sentiment. In this paper, we focus on extracting public sentiment of microblog emergencies. A subtopic-level opinion mining method is proposed based on two-phase optimization. Different subtopics of emergencies are extracted based on retweets. Opinion tweets are classified to different subtopics. The sentiment score of opinion holders is calculated. The above results are optimized based on users and endorsement interactions between users. Experimental results validate the effectiveness of the proposed method.

Keywords: sina microblog, opinion, sentiment analysis

1 Introduction

More and more people post tweets by using microblog platforms such as Twitter[1] and Chinese Sina Weibo[2]. These opinion-based data which cover the most diverse topics enable the creation of valuable real-time applications that monitor public opinion and summarize the aggregated sentiment of online society. This problem can be addressed as Sentiment Analysis or Opining Mining. One method is based on opinion words in context by using an opinion dictionary to identify and determine sentiment orientation[1]. The other method applies machine learning techniques and treats sentiment analysis as a classification problem[2].

Recently, the existing methods of sentiment analysis are employed into the scenario of microblog. Davidov et al.[3] utilized Twitter characteristics and language conventions as features to train sentiment classifier. Silva et al.[4] proposed an augmentation train procedure. Tweets were classified into positive, negative and neutral according to the classifier. O'Connor et al.[5] found that surveys of consumer condence and political opinion correlate with sentiment word frequencies in tweets. Barbosa et al.[6] investigated a two-stage SVM classifier with two

[1] http://www.twitter.com
[2] http://www.weibo.com

sets of features: meta-information about the words of tweets and the written style of tweets. Instead of learning textual models to predict content polarity, Guerra et al.[7] proposed a transfer-learning approach which utilized the user bias to analysis the sentiment orientation of tweets. However, it selected a few users named attractors who had clearly bias toward one or more sides of a discussion, based on prior knowledge.

In this paper, we propose a subtopic-level sentiment analysis method to extract microblog users' opinions toward different subtopics of an emergency. The proposed approach is different from the methods mentioned above. It is not necessary to select attractors for each discussion side. Instead of assuming one's opinion keeps stable in the whole discussion, we assume users' opinions only keep stable in the period of a time parameter.

2 The proposed method

2.1 Subtopic extraction based on retweets

A subtopic is defined as a set of noun words or noun phrases that appear explicitly in tweets. The frequent noun words and phrases are extracted as candidate subtopics. A term-tweet matrix TW, as shown in Formula (1), is defined to represent the frequency of each candidate subtopic.

$$[TW]_{m\times n} = \begin{bmatrix} t_0w_0 & t_0w_1 & \dots & t_0w_{n-1} \\ t_1w_0 & t_1w_1 & \dots & t_1w_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ t_{m-1}w_0 & t_{m-1}w_1 & \dots & t_{m-1}w_{n-1} \end{bmatrix} \tag{1}$$

Here, m is the number of candidate subtopics, n is the number of tweets. t_iw_j is the frequency of keyword t_i which appears in tweet w_j. A matrix WW, as shown in Formula (2), is defined to represent the retweet relationship between tweets.

$$[WW]_{n\times n} = \begin{bmatrix} w_0w_0 & w_0w_1 & \dots & w_0w_{n-1} \\ w_1w_0 & w_1w_1 & \dots & w_1w_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n-1}w_0 & w_{n-1}w_1 & \dots & w_{n-1}w_{n-1} \end{bmatrix} \tag{2}$$

Here, n is the number of tweets. w_iw_j equals 1 if one of the following conditions is satisfied: 1) $i = j$; 2) tweet w_i is a retweet of tweet w_j; 3) tweet w_j is a retweet of tweet w_i. Otherwise, w_iw_j equals 0. A multiplication is executed on the above two matrixes. A new term-tweet matrix TW=$TW \times WW$ is generated. By using the multiplication operation, the candidate subtopics appear in tweets and retweets can be reinforced mutually.

2.2 Two-Phase Sentiment Optimization

User-Based Optimization A time decay function is designed to model tweets over time posted by the same user. The probability of tweets sharing the same sentiment is calculated. The time decay function is normally defined as Formula (3) based on time-stamps, which is a monotonic decreasing function.

$$f(x) = \frac{1}{1 + \alpha \times \mathrm{e}^{(x/c_1 - c_2)}} \tag{3}$$

The value of the function is in the range $(0, 1)$ and will be reduced within a time interval. Where α is a parameter that adjusts the curve shape of the function. c_1 and c_2 are used to determine the time duration. x represents duration between tweets' time-stamp.

All tweets have been divided into positive or negative class according to their sentiment scores. We don't consider the neutral class which sentiment score is zero. A graph $G(V, E)$ is constructed as shown in Figure 1. V represents all tweets which have been divided into two classes(positive or negative). An edge in E connects two tweets which are posted by the same user toward a same sub-topic, as shown in Figure 1. A solid line rectangular is drawn around the tweets which posted by a same user, for example tweet (w_1, w_2, w_3, w_4) and (w_5, w_6, w_7, w_8). Inside this rectangular, tweets that surrounded by dot line rectangular and connected by solid line are the same sub-topic, such as (w_1, w_2, w_3, w_4), (w_5, w_7) and (w_6, w_8). The weight of each edge means the probability of tweets sharing the same sentiment measured by the time-decay function based on the duration between two tweets, as in Formula (4).

$$f(\Delta ts) = \begin{cases} 1 & if \quad \Delta ts = 0 \\ \frac{1}{1+\alpha \times \mathrm{e}^{(\Delta ts/c_1 - c_2)}} & otherwise \end{cases} \tag{4}$$

Where Δts is the duration between two tweets. We set the time-stamp unit as one day, for example, the weight of $E(w_1, w_2)$ is 1.0 means tweet w_1 and w_2 are posted by the same user about one sub-topic in one day, as show in Figure 1 .

Definition 1. *Each class $C\{positive, negative\}$ has an attraction to each tweet w calculated by the probability of all the tweets which have connections with it in class C. It is defined as $attract\langle w, C\rangle$, as Formula* (5).

$$attract\langle w_i, C\rangle = \sum_{w_j \in C \cap E(w_i, w_j) \in E} f(\Delta ts_{ij}) \tag{5}$$

Retweet-Based Optimization Retweet means users can repost a tweet and append some comments or do nothing. Through retweeting users just want to share the original tweet with more users. Therefore, it is difficult to mine the sentiment of these tweets if only considering the content of tweets. Retweets represent the endorsement interactions, through which a user explicitly agrees

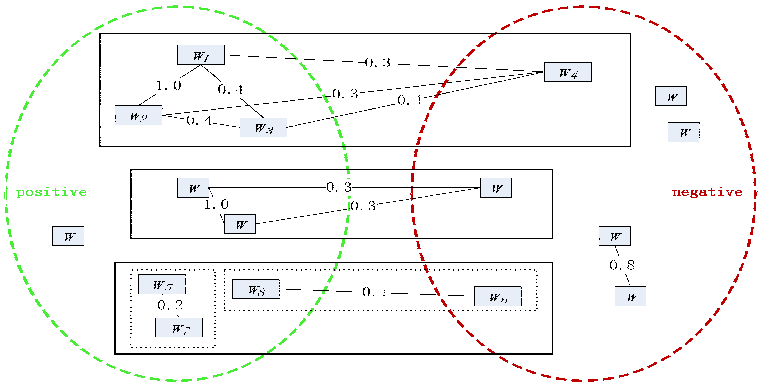

Fig. 1: An Example of User-Based Optimization Graph

with other users. A retweet-tree is completed for each subtopic. These trees can be called retweet-forest. As illustrated in Figure 2, it is an example of retweet-forest about a certain subtopic. A node indicates a tweet. In each retweet-tree, except the root node, any node is a retweet that repost its parents.

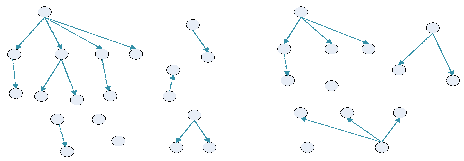

Fig. 2: An Example Retweet-Forest About A Certain Sub-Topic

If the sentiment score of a tweet is lower than the predefined threshold β, it will be optimized by the function as follow Formula (6).

$$op(w_c) = op(w_c) + (1 - \alpha) * op(w_p)$$
$$\alpha = \begin{cases} 0 & if \quad |OP(w_c)| \leq \beta \\ 1 & otherwise \end{cases} \tag{6}$$

Where w_p represents one tweet in the retweet-tree and w_c is the child node of w_p, β is the predefined threshold, α is 0 if the sentiment score of w_c is smaller than β, otherwise α is 1.

3 Experiment

3.1 Dataset description

We obtained the dataset from *Sina Weibo* by searching keywords *"7.23bullet train collision"*. About 424,090 tweets were posted by 322,523 users.

3.2 Experimental results and evaluation

Based on the relationship between original tweets and retweets, we run *K-means* clustering algorithm. Table 1 indicates that the accident can be clustered into three subtopics: Ministry of Railways, passengers and reaction after the outbreak.

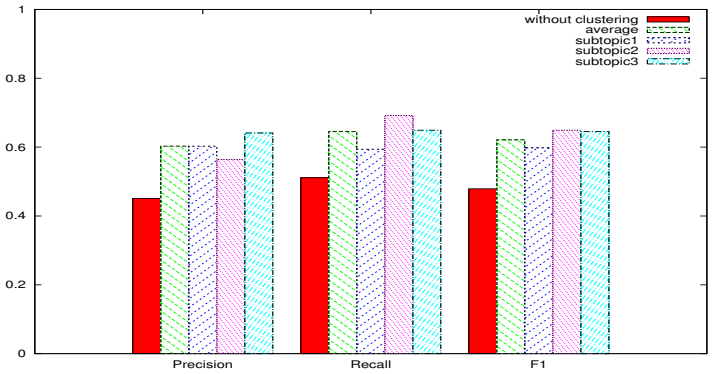

Fig. 3: Performance of Opinion Identifier with Retweet-based Optimization

Table 1: Subtopics of "7.23 bullet train collision"

Subtopic	keywords
subtopic1	Ministry of Railways, Shinkansen, real-name system, transport, EMU
subtopic2	victims, survivors, natural disasters, compensation
subtopic3	press conference, female reporter, spokesman, Central Propaganda Department

1000 tweets are randomly selected for each subtopic. These tweets are manually labeled by three annotators. The performance of subjective identifier, i.e., whether a tweet is opinionated, is evaluated. We use the standard evaluate measures: precision, recall and F-score. The proposed method is mainly compared with the one only based on lexicon without clustering subtopics. The results shown in Figure 3 indicate that the retweet-based method outperforms the baseline method. The average measure value is better than the one without cluttering subtopic, exceeding 11.5%, 11.4% and 18.4%, respectively. The results shown in Figure 4 indicate that the method with user-based optimization outperforms the baseline method. The average measure value is better than the one without cluttering subtopic, exceeding 12.3%, 13.8% and 21.2%, respectively.

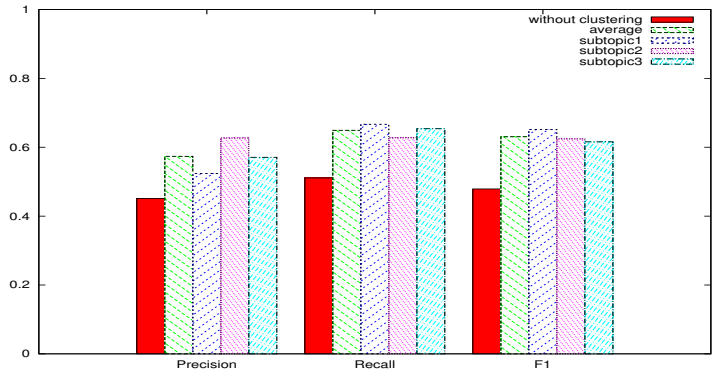

Fig. 4: Performance of Opinion Identifier with user-based optimization

4 acknowledgements

This work is supported by National Natural Science Foundation of China under grants 61173170, 61300222, 61433006, and U1401258, and Innovation Fund of Huazhong University of Science and Technology under grants 2015TS071.

References

1. M. Hu and B. Liu. Mining and summarizing customer reviews. In: 10th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 168-177, ACM Press, New York (2004)
2. B. Pang and L. Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: 42nd Annual Meeting of the Association for Computational Linguistics, pp. 271-278 (2004)
3. D. Davidov, O. Tsur, and A. Rappoport. Enhanced Sentiment Learning using Twitter Hashtags and Smileys. In: 23rd International Conference on Computational Linguistics , pp. 241- 249 (2010)
4. I. S. Silva, J. Gomide, A. Veloso, W. M. Jr. and R. Ferreira. Effective sentiment stream analysis with self-augmenting training and demand-driven projection. In: 34th Annual ACM SIGIR Conference, pp. 475484 (2011)
5. B. OConnor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In: 24th International AAAI Conference on Weblogs and Social Media, pp. 122-129 (2010)
6. L. Barbosa and J. Feng. Robust sentiment detection on twitter from biased and noisy data. In: 23rd International Conference on Computational Linguistics: Posters, pp. 3644 (2010)
7. P. H. C. Guerra, A. Veloso, W.M. Jr and V. Almeida. From bias to opinion: A transfer-learning approach to real-time sentiment analysis. In: 17th ACM SIGKDD international conference on Knowledge discovery and data mining,, pp. 150-158 (2011)