

Exploiting Community Emotion for Microblog Event Detection

Gaoyan Ou^{1,2}, Wei Chen^{1,2}, Tengjiao Wang^{1,2}, Zhongyu Wei^{1,3},
Binyang Li⁴, Dongqing Yang^{1,2} and Kam-Fai Wong^{1,3}

¹Key Laboratory of High Confidence Software Technologies, Ministry of Education, China

²School of Electronics Engineering and Computer Science, Peking University, China

³Shenzhen Research Institute, The Chinese University of Hong Kong, China

⁴Dept. of Information Science & Technology, University of International Relations, China

pekingchenwei@pku.edu.cn

Abstract

Microblog has become a major platform for information about real-world events. Automatically discovering real-world events from microblog has attracted the attention of many researchers. However, most of existing work ignore the importance of emotion information for event detection. We argue that people's emotional reactions immediately reflect the occurring of real-world events and should be important for event detection. In this study, we focus on the problem of community-related event detection by community emotions. To address the problem, we propose a novel framework which include the following three key components: *microblog emotion classification*, *community emotion aggregation* and *community emotion burst detection*. We evaluate our approach on real microblog data sets. Experimental results demonstrate the effectiveness of the proposed framework.

1 Introduction

Microblog has become a popular and convenient platform for people to share information about social events in real time. When an external event occurs, it will be quickly propagated between microblog users. During propagation process of an event, sufficient amount of users will express their emotions. Taking Sina Weibo¹ as an example, more than 12 percent of users use emoticons² when reposting an event-related microblog message.

The emotion information can not only help us better understand a given event, but also be utilized to discover new events. Figure 1 shows

¹<http://weibo.com/>

²An icon to indicate user's emotion, as shown in Table 1.

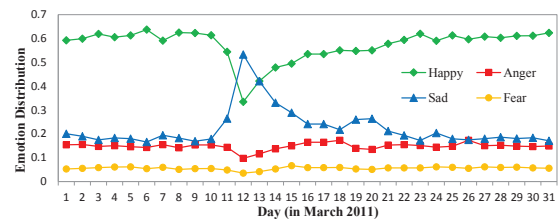


Figure 1: The global emotion dynamics

the emotional distribution dynamics of the overall microblog messages in March 2011. The sudden change of the public emotion distribution on March 12 indicates a public event: *3.11 Earthquake in Japan*. We can see that emotional changes immediately reflect the occurring of real-world events, thus it is reasonable to use them to perform event detection.

Most existing research on microblog event detection (Weng and Lee, 2011; Sakaki et al., 2010; Becker et al., 2010) only account for the factual information (e.g., burstness of topic keywords). They usually ignore the importance of emotion information for event detection. Although there have recently been a few papers (Zhao et al., 2012a; Nguyen et al., 2013; Akcora et al., 2010) in this direction, they have a number of disadvantages. Firstly, they can not detect community-related events. Since they all aggregate emotion at global level, they can only discover national attention events, such as public holidays (“Christmas” and “Spring Festival”) or natural disasters. In many applications, discovering events related to a certain group of users or a certain topic is more meaningful. Consider the following questions: “*what happened in the football community last week?*” and “*what are the most significant events in the lawyer community last month?*” Secondly, they assign equal weight to each microblog message or user when aggregating them to a global emotion score. Such approaches may lead to incorrect results when a user posts emotion spam

in the community. Thirdly, there is a lack of quantitative evaluation using real-world events in existing work. For example, there is only case study in (Zhao et al., 2012a) and (Akcora et al., 2010).

In this study, we focus on a new task of detecting community-related events via community emotion. Our intuition is inspired from the social psychology theory of *group emotion*. In social psychology, *group emotion* refers to the moods, emotions and dispositional affects of a group of people³. It can arise as a result of group-relevant events (Smith et al., 2007). For example, people will feel happy when their favorite football team wins a game and feel sad when their team loses a game. Thus we can use the community emotion as signals to detect community-related events. To achieve good performance of this new task, the following two factors must be considered: 1) *how to measure the community emotion based on microblog message or user emotion* and 2) *how to perform event detection based on the sequence of community emotion*.

To measure community emotion, we argue that in a given community, different users have different emotional authorities. The emotion of highly influential people in the community may be more important in determining the community emotion (Barsäde and Gibson, 1998). We propose to use the user's emotion authority computed by the social network of the community to weight the user when aggregating community emotion. Since spam user is unlikely to have high emotion authority in the community, our method can reduce the effect of emotion spam. Based on the community emotion, we present an emotion burst detection algorithm for the community emotion distribution sequence. We identify two emotional states of the community: "emotion stable" state and "emotion burst" state. We use the Dirichlet distribution to model the generation process of the community emotion distribution. An efficient emotion burst detection algorithm is presented to detect community-related events.

We evaluate our approach on large-scale microblog data sets by using real-world event list for each community. Experimental results demonstrate that the community emotion is an effective signal for community-related event detection. In comparison with several baseline methods, our emotion burst detection algorithm also improves

the event detection performance in terms of precision, recall and F-measure.

2 Related Work

In this section, we review the related work on sentiment analysis and event detection in microblog, respectively.

Sentiment Analysis in Microblog: Sentiment analysis (Pang and Lee, 2008; Liu, 2012) is mainly about analyzing people's opinions, sentiments and emotions towards a given event, topic, product, etc. Microblog platforms like Twitter and Weibo, provide people a convenient way to post their emotional reactions towards social events in almost real time. This leads to increasing number of interests on sentiment analysis in microblog data (Davidov et al., 2010; Liu et al., 2012; Go et al., 2009; Agarwal et al., 2011; Pak and Paroubek, 2010; Jiang et al., 2011; Speriosu et al., 2011; Birmingham and Smeaton, 2010). The training data for microblog sentiment analysis are usually obtained in an automatic manner by utilizing emoticons, hashtags and smileys. Davidov et al. (2010) propose an approach to automatically obtain labeled training examples by exploiting hashtags and smileys. Liu et al. (2012) proposed an emoticon smoothed method to integrate both manually labeled data and noisy labeled data for Twitter sentiment classification.

Different from existing microblog sentiment analysis work, which aims at discovering sentiments and emotions for an event or topic given in advance, we are interested in utilizing the emotion information in microblog messages for real-world event detection. Our work use sentiment analysis as a tool to perform microblog emotion classification. Then we propose an event detection algorithm based on the sequence of community level emotion distribution.

Event Detection in Microblog: Event detection from microblog data has attracted increasing attention recently. We divide existing work into the following two categories: *non-emotion-based* methods and *emotion-based* methods. *Non-emotion-based* methods try to exploit keyword or activity patterns to discover events (Weng and Lee, 2011; Sakaki et al., 2010; Becker et al., 2010; Mathioudakis and Koudas, 2010). Mathioudakis and Koudas (2010) first identify "bursty" keywords and then discover events by grouping bursty keywords together. Zhao et al. (2012b) focuses on

³http://en.wikipedia.org/wiki/Group_emotion

identifying event-related burst from social media based on multiple activities. They propose a unified optimization model to integrate multiple correlated activity streams based on the state machine in (Kleinberg, 2003).

Emotion-based methods try to exploit the emotional reactions to discover events (Zhao et al., 2012a; Akcora et al., 2010). Akcora et al. (2010) model public emotion as an emotion vector computed by counting emotion words and then use a rule-based method to identify public breakpoints. Zhao et al. (2012a) use a simple top-k method to detect abnormal events based on sequence of sentiment variance.

Our method is also an emotion-based method. However, our approach is different from existing emotion-based methods in the following aspects. Firstly, while existing work aggregates emotion for all users, we focus on emotion for a certain community to discover community-related events. Secondly, existing methods assume that the emotions of different users are of equal importance. We distinguish user’s emotional authority based on the community structure of users.

3 Preliminary Definitions

In this section, we first give some basic concepts before formally defining our problem.

Topical Community: A group of connected users that are interested in a specific topic. A topical community can be denoted as $C = \{V, E, M\}$, where V is the set of community users, E is the set of relationships of between users. $M = \{M_1, M_2, \dots, M_T\}$ is the microblog message collection in the community, which is segmented according to time. M_t is the microblog message collection in time interval t .

Emotion Distribution: An emotion distribution e_t is a real-value N -dimension vector sampled from the emotion space, satisfying the constraint $\sum_{i=1}^N e_{ti} = 1$. It indicates the emotional state of a microblog message, a user or a community at time t . At a given time interval t , a user emotion distribution $e_t(u)$ is computed by an aggregation function over the emotion distribution of the microblog messages posted by u in time interval t . Community emotion distribution $e_t(c)$ is computed by an aggregation function over the emotion distribution of the community users in c at time interval t .

Community Emotion Burst: Given an emotion distribution sequence e_t for community c , an

emotion burst is defined as period $[t_s, t_e]$ in which some event relevant to c takes places. During the time period, the emotion distribution of c is :1) significantly different from its average emotion distribution, or 2) extremely uneven distributed.

Given the above definitions, our object is then to detect community-related events from the emotion distribution sequence of the community.

4 The Proposed Framework

In this section, we describe our microblog event detection framework. The framework aims to detect community-related events based on the community emotion. The overview of our framework is shown in Figure 2. In particular, we define the following four main components:

1) **Microblog emotion classification:** We train emotion classification model by automatically acquiring training data using the emoticons.

2) **Community emotion aggregation:** We assume that in a given community, different users have different weights when aggregating community emotion. Thus we propose a novel weighting approach based on the user’s authority.

3) **Community emotion burst detection:** Given the community emotion, we propose an emotion burst detection algorithm to detect community emotion bursts.

4) **Event extraction:** The function of this component is to extract event keywords for each community emotion burst. We count the document frequency (DF) of each term contained in the microblog messages in the burst period. Then the top-5 DF terms are used to describe the event occurring during the burst period, although there exist alternative techniques (Ritter et al., 2012; Li et al., 2010).

Since the last component is not the main focus of this work, we only introduce the first three components in detail in the following subsections.

4.1 Microblog Emotion Classification

We build an emotion classifier using the multinomial Naïve Bayes classifier for each community to classify microblog messages into different emotion classes. Here we are interested in the setting where the microblog messages arrive as stream, thus it is not appropriate to use a constant set of manually labeled messages. To avoid manual annotation cost, we adopt a distant supervision approach (Go et al., 2009; Davidov et al., 2010; Hu

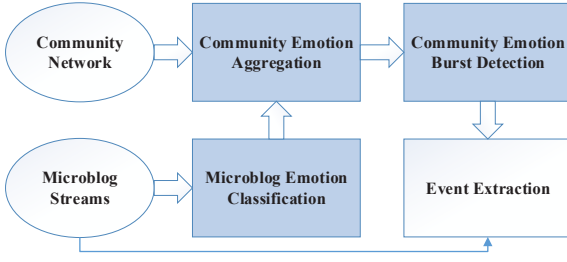


Figure 2: Overview of our community-related event detection framework

Table 1: List of emoticons

Happy (26)	good 👍	laugh 😄	love 😍	smile 😊	cute 😘
Anger (19)	anger 😡	hate 😠	curse 😡	despise 😏	crazy 😜
Sad (13)	sad 😞	disappoint 😞	cry 😭	unhappy 😞	unwell 😷
Fear (11)	fear 😨	surprise 😮	shame 😞	doubt 😟	fright 😱

et al., 2013) to acquire labeled microblog messages automatically by using the emoticons. We first select a set of emoticons which are most frequently used to express emotion in microblog. Then we manually map each emoticon into four emotion classes (26 for *happy*, 19 for *anger*, 13 for *sad* and 11 for *fear*). We only show the top five emoticons for each emotion type in Table 1. The labeled emoticons can then be used to acquire training data to train an emotion classifier for a community in any time period.

We combine the features which have been proven effective by previous work, such as punctuation and emotion lexicons. Specifically, we use the following features : 1) Words appearing in the microblog message serve as word features, 2) Number of “!” characters and “?” in the microblog message, and 3) Each term in a general emotion lexicon serves as an emotion lexicon feature.

The emotion distribution $e(m)$ of a microblog message m is represented by a N -dimension distribution vector. For example, if a microblog message m is classified as *happy*, then its emotion distribution $e(m)$ is $[1, 0, 0, 0]$. The emotion distribution of $e(u)$ of user u at time t is average emotion distribution of the microblog messages posted by him during time t .

$$e_t(u) = \frac{1}{N_{ut}} \sum_{m=1}^{N_{ut}} e(m) \quad (1)$$

where N_{ut} is the number of microblog messages

posted by user u at time t .

4.2 Community-level Emotion Aggregation

A common strategy to measure community emotion is to compute the average emotion of the community users. It is based on the assumption that the emotion of different community users are with equal importance. This is implemented by employing an *average* aggregation function on the individual emotion distribution $e_t(u)$:

$$e_t(c) = \frac{1}{N_c} \sum_{u \in c} e_t(u) \quad (2)$$

where N_c is the number of users in community c .

Intuitively, the emotion of user with higher authority in community should be more important in determining community emotion. Thus we estimate community emotion by taking into account user’s emotional authority, which is based on the assumption that different users in a community have different emotional authorities.

We employ HITS algorithm(Kleinberg, 1999) to compute the user authority $auth(u)$ based on the user network $\{V, E\}$ of the community. Then $auth(u)$ is used to represent the emotional authority of user u . This authority-based community emotion aggregation approach can also reduce the effect of spam users, since they usually have low authorities in the community network. For simplicity and computation efficiency, we assume that $auth(u)$ is time independent, which means that we only need to run the HITS algorithm once for each community. Given the user emotion distributions and the user emotional authorities, the emotion distribution for a community c in time interval t can be measured as:

$$e_t(c) = \frac{1}{A_{ct}} \sum_{u \in c} auth(u) e_t(u) \quad (3)$$

where $A_{ct} = \sum_{u \in c_t} auth(u)$ is the normalization term.

4.3 Community Emotion Burst Detection

We formulate our problem into the binary state model framework (Kleinberg, 2003). For a given community c , there are T time intervals in total, with community emotion distribution sequence $e = (e_1, e_2, \dots, e_T)$ and state sequence $q = (q_1, q_2, \dots, q_T)$. Each q_t can be one of the following two states: $q_t = 0$ (“emotion stable” state) and $q_t = 1$ (“emotion burst” state). Our objective

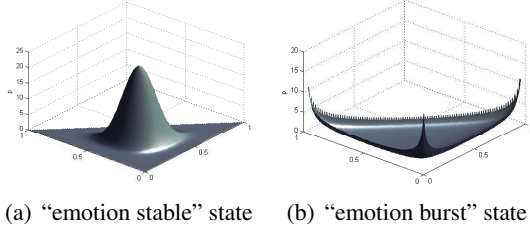


Figure 3: Emotion probability density for “emotion stable” state and “emotion burst” state

is to find an optimal state sequence q^* given the emotion distribution sequence e .

Since each e_t is a distribution rather than positive integer, the emotion generation process can no longer be modeled by a Poisson distribution. We choose to model the emotion generation process by the Dirichlet distribution. This process is analogous to the document-level topic generation in the LDA topic model (Blei et al., 2003).

If community c is in an “emotion stable” state in time interval t , its emotion distribution e_t should be close to the average emotion distribution e_a . The density function is defined as $f(e_t, e_a, s_t = 0) = \text{Dirichlet}(\alpha_0 e_a)$, where $s_t = 0$ indicates that the community is in an “emotion stable” state and α_0 is a positive parameter. To ensure that e_t closer to e_a will get higher probability, α_0 should satisfy the constraint $\alpha_0 \cdot \min e_a > 1$. An example of the probability density function of $f(e_t, e_a, s_t = 0)$ in a three dimension emotion space is shown in Figure 3(a).

If community c is in an “emotion burst” state in time interval t , the emotion distribution of c is : 1) significantly different from its average emotion distribution, or 2) extremely uneven distributed. For example, assume $e_a = [0.25, 0.25, 0.25, 0.25]$, if community is in burst in time interval t , its emotion distribution is more likely to be $[0.1, 0.4, 0.1, 0.4]$ (significantly different from e_a) or $[0.1, 0.7, 0.1, 0.1]$ (extremely *anger*). The density function is then defined as $f(e_t, e_a, s_t = 1) = \text{Dirichlet}(\alpha_1 e_a)$. $s_t = 1$ indicates that the community is in an “emotion burst” state and α_1 should satisfy the constraint: $\alpha_1 \cdot \max e_a < 1$. An example of the probability density function of $f(e_t, e_a, s_t = 1)$ in a three dimension emotion space is shown in Figure 3(b).

Based on above discussion, the cost function for

an emotion state sequence q can be defined as:

$$\text{cost}(q|e) = \sum_{t=1}^T -\ln(f_t(e_t, e_a, q_t)) + b \ln\left(\frac{1-\pi}{\pi}\right) \quad (4)$$

where π is the probability the community will change the emotion state in two consecutive time intervals t and $t + 1$. b denotes the number of emotion state changes in the whole time intervals $[1, T]$.

Equation 4 is exactly the objective function we need to optimize. This optimization problem can be efficiently solved by using a dynamic programming procedure, as summarized in Algorithm 1. Algorithm 1 mainly consists of two phases: a forward phase (line 5 - line 10) which calculates the minimal cost for emotion distribution sub-sequence and a backward phase to construct the optimal emotion state sequence (line 11 - line 14).

For convenience, we use $f_t(s)$ to denote $f_t(e_t, e_a, s)$. $\pi_{s's} = \pi$ if $s' \neq s$, otherwise $\pi_{s's} = (1 - \pi)$. $c_t(s)$ denotes the minimal cost of generating the emotion distribution sub-sequence $\{e_1, \dots, e_t\}$ with $q_t = s$. $q'_t(s)$ stores the state of time interval $t - 1$ for the most likely state sub-sequence so far with $q_t = s$.

Algorithm 1 Emotion Burst Extraction

Input: The emotion distribution sequence $e = (e_1, e_2, \dots, e_T)$, the state transition cost π and the parameters α_0 and α_1

```

1: for each  $s \in \{0, 1\}$  do
2:   Initialize  $c_1(s) = -\ln f_1(s)$ 
3:   Initialize  $q'_1(s) = 0$ 
4: end for
5: for each  $t = 2, \dots, T$  do
6:   for each  $s \in \{0, 1\}$  do
7:      $c_t(s) = \min_{s'} (c_{t-1}(s') - \ln f_t(s) - \ln \pi_{s's})$ 
8:      $q'_t(s) = \arg \min_{s'} (c_{t-1}(s') - \ln \pi_{s's})$ 
9:   end for
10: end for
11:  $q^*(T) = \arg \min_s c_T(s)$ 
12: for each  $t = T - 1, \dots, 2$  do
13:    $q^*(t) = q'_{t+1}(q^*(t + 1))$ 
14: end for

```

Output: The optimal emotion state sequence q^*

Table 2: Basic statistics of data sets

Community	#User	#Link	#Microblog	#Event
legal cases	4937	97639	269871	31
football	9928	105483	416631	75
economy	2657	65403	179584	46
singer	3759	79458	478265	53

5 Experimental Setup

5.1 Data Set

We use a large microblog data set crawled from Sina Weibo, which is the most popular microblog service in China. This data set contains 212,859,466 Chinese microblog messages posted by 916,126 users in a period of about 8 months (1/12/2010~ 20/7/2011). We also crawled the *following* network, which resulted in 15,681,296 *following* relationships for the 916,126 users.

We choose four communities: “legal cases”, “football”, “economy” and “singer”. To obtain the members for each topical community, we manually selected several keywords and input them as queries to the user search page⁴. After filtering out the users whose microblog messages are not collected in our corpora, we extract the sub-network of the users from the whole *following* network.

We use a simple but efficient method to extract microblog messages for each topical community: 1) If a microblog message is posted by the community members and also contains keywords of the community, it belongs to the community; 2) If a microblog message is posted by community member u and it is *reposting*, *commenting on* or *replying to* another microblog posted by community member v , then it belongs to the community. The basic statistics of our data sets are shown in Table 2.

5.2 Ground Truth Generation

Algorithm 1 generates a list of emotion bursts for each community. Since our goal is to identify community-related events, we need to evaluate how well the generated emotion bursts correspond to the ground truth real-world events. To generate the ground truth for evaluation, we utilize the news website Sina News⁵. Two PhD students are invited to manually generate a list of real events for each community by referring to the annual top-

⁴<http://s.weibo.com/user/&tag=keyword>

⁵<http://news.sina.com.cn>

ic summary page⁶ of Sina News. The annotation agreement is higher than 90%. Each event is also associated with its occurred date. The number of events for each community is shown in the last column of Table 2. For each community, the event list is then used to evaluate the performance of different event detection models.

5.3 Evaluation Metric

We use precision, recall and F-measure as evaluation metric. For each community, we compare the event list \mathcal{E}^c and the generated burst list \mathcal{B}^c to compute the above metric. Specifically, the precision, recall and F-measure for a community c are defined as follows:

$$P = \frac{\sum_{j=1}^{|\mathcal{E}^c|} \sum_{k=1}^{|\mathcal{B}^c|} I(\mathcal{E}_j^c \in \mathcal{B}_k^c)}{|\mathcal{B}^c|} \quad (5)$$

$$R = \frac{\sum_{j=1}^{|\mathcal{E}^c|} \sum_{k=1}^{|\mathcal{B}^c|} I(\mathcal{E}_j^c \in \mathcal{B}_k^c)}{|\mathcal{E}^c|} \quad (6)$$

$$F = \frac{2 \times P \times R}{P + R} \quad (7)$$

where \mathcal{E}_j^c is the occurring time of the j -th event in community c , \mathcal{B}_k^c is the k -th identified burst for community c . $I(\cdot)$ is the indicator function (which equals 1 if its argument is true and 0 otherwise).

The final precision, recall and F-measure are averaged over different communities.

5.4 Compared Methods

We now introduce four methods used for comparison as follows:

EmoPeakFind: The method proposed in (Akorca et al., 2010), which aims at discovering breakpoints from public emotion. They use the following simple rule to find breakpoints from emotion sequences:

$$Sim(e_{t-1}, e_t) < Sim(e_{t-2}, e_{t-1}) \quad (8)$$

$$Sim(e_{t-1}, e_t) < Sim(e_t, e_{t+1}) \quad (9)$$

where Sim is a similarity function. We use the cosine similarity function in our evaluation.

TopKEmoVar: The method used in (Zhao et al., 2012a). They first derive a sequence of relative variation V^n for each single emotion sequence e^n . Then they define a sequence of emotion variation as $(\sum_{n=1}^4 |V_t^n|)$. This sequence is sorted in descending order and the top- k t is selected as burst

⁶<http://news.sina.com.cn/zt/>

Table 3: Event detection performance of different methods

Method	Precision	Recall	F-measure
<i>EmoPeakFind</i>	0.313	0.625	0.417
<i>TopKEmoVari</i>	0.423	0.423	0.423
<i>KLB</i>	0.575	0.702	0.632
<i>MBurst</i>	0.534	0.497	0.515
<i>Our Method</i>	0.654	0.715	0.683

state points, where k is set to be the size of the event list for each community.

KLB: The method proposed in (Kleinberg, 2003). Note that *KLB* can only deal with a single sequence $e^n = (e_1^n, \dots, e_t^n, \dots, e_T^n)$ for emotion type n . We first apply *KLB* to find the optimal state sequences for each emotion type. Then we perform an OR function to merge the N state sequences to a global emotion state sequence.

MBurst: The method proposed in (Zhao et al., 2012b) for multi-stream burst detection. *MBurst* is evaluated on three activity streams in (Zhao et al., 2012b). Here we apply *MBurst* to the N emotion streams. Then we perform an OR function to merge the N state sequences to a global emotion state sequence.

6 Experimental Results

6.1 Performance Comparison Results

In this experiment, we compare our method with different baseline methods as introduced in Section 5.4. We use Equation (3) to aggregate community emotion for all the compared methods. The parameter α_0 and α_1 are empirically set to be $\frac{5}{\min_n e_a^n}$ and $\frac{0.5}{\max_n e_a^n}$, respectively. The experimental results are shown in Table 3.

Table 3 shows that *EmoPeakFind* and *TopKEmoVari* are less effective than other methods. The simple rule used in *EmoPeakFind* produced many noisy bursts, leading to low precision. *TopKEmoVari* only considers the relative variation of two consecutive time intervals. The choice of k is also nontrivial since it is hard to know the number of events before the events are identified. Note that *EmoPeakFind* and *TopKEmoVari* are both rule-based methods, while *KLB*, *MBurst* and *Our Method* are state machine based methods. This demonstrates that for community-related emotion burst detection, it is more appropriate to use a state machine based model.

It looks surprising that *MBurst* performs worse than *KLB*, since *MBurst* is specifically designed

Table 5: Performance of different weighting schemes in terms of F-measure

Community	Weighting schema	
	<i>equal</i>	<i>HITS-based</i>
legal cases	0.517	0.590
football	0.674	0.765
economy	0.642	0.712
singer	0.589	0.665
avg	0.605	0.683

for multiple streams. However, *MBurst* is based on the assumption that the states of multiple streams in the same time interval tend to be the same (i.e., there is positive correlation between two different streams). This assumption no longer holds in the context of different emotion streams. For example, if a negative event occurs in the community, while *sad* emotion is likely to be in a burst state, *happy* emotion is not likely to be in a burst state.

We can see from Table 3 that our method outperforms the four baselines. The main reason is that our burst detection method is based on the sequence of community emotion distribution. Modeling community emotion as a distribution is more suitable than modeling several different emotion types.

We further show some example events detected by our model in Table 4. Since the event keywords are manually translated from Chinese, one keyword may include more than one English words. We can see that community emotion can help to detect emotionally significant events for different communities. For example, the “legal cases” community is in a strong *anger* emotional state on December 25, 2010, which indicates an important event “Qian Yunhui’s case”.

6.2 Effect of Emotion Aggregation Functions

In this experiment, we show the importance of correctly aggregation community level emotion for community-related event detection. We compare the two aggregation approaches introduced in section 4.2. The first approach assigns equal weight to each community users, while the second approach assigns weights to users based on their authorities in the community.

The performance in terms of F-measure is shown in Table 5. It is obvious that, for all commu-

Table 4: Examples of events and the corresponding community emotions for two communities. The four emotion types (*happy*, *anger*, *sad* and *fear*) are mapped to green, red, blue and yellow color, respectively.

Community	Date	Emotion	Event keywords
legal cases	25/12/2010	—■—■—■—■	Qian Yunhui, village head, Yueqing, run over, Zhejiang
	22/4/2011	■—■—■—■	Yao Jiaxin, death penalty, first instance, Xi’an, intentional killing
	9/5/2011	■—■—■—■	Xia Junfeng, final judgment, death penalty, pedlar, Shenyang
football	2/12/2010	■—■—■—■	Qatar, 2022, world cup bid, win, FIFA
	14/2/2011	—■—■—■—■	Ronaldo, retire, legend, football, Brazil
	29/5/2011	■—■—■—■	Barcelona, 2011, Champions League final, win, UEFA

nities, HITS-based weighting approach performs better than equal weighting approach. Thus we can conclude that user authority is important when aggregating community emotion.

We further perform an empirically analysis of the events that successfully identified by HITS-based approach but failed by equal weighting approach. By manually analyzing the microblog messages of the corresponding time intervals, we found that most of the errors of equal weighting approach were caused by emotion spam. Users of low authority post many microblog messages with extreme emotion to claim the attention of the community. Since there is no significant community-related events at that time interval, we do not observe emotional changes of the high authority users. In the equal weighting method, the existence of emotion spam lead to wrong result of community emotion. Since the weights of users who post emotion spam are small in the HITS-based approach, they have little effect on the community emotion. This is the main reason why HITS-based weighting method is more effective than equal weighting method.

6.3 Parameter Sensitivity

In this experiment, we test the sensitivity of our model by different choices of the parameters α_0 and α_1 . $\alpha_0 \min_n e_a^n$ is set to be [2, 3, 4, 5, 6, 7, 8] and $\alpha_1 \max_n e_a^n$ is set to be [0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8]. The event detection results of different parameter settings are shown in Figure 4.

It can be seen from Figure 4 that: 1) The performance is relatively good in a particular range of the parameters. When $\frac{2}{\min_n e_a^n} \leq \alpha_0 \leq \frac{7}{\min_n e_a^n}$ and $\frac{0.3}{\max_n e_a^n} \leq \alpha_1 \leq \frac{0.7}{\max_n e_a^n}$, the F-measure is

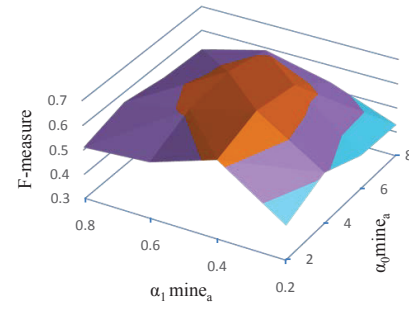


Figure 4: F-measure of our method for event detection in different parameter settings

consistently larger than 0.55. 2) In general, the performance is more sensitive to α_1 than to α_0 . Note that α_1 controls the generation process of the “emotion burst” state, thus it is relatively more important to tune α_1 than α_0 . The experimental results demonstrates that when $\frac{0.3}{\max_n e_a^n} \leq \alpha_1 \leq \frac{0.7}{\max_n e_a^n}$, the performance can be relatively good.

7 Conclusion

Microblog has become a popular and convenient platform for people to share information about social events in real time. In this paper, we focus on the problem of community-related event detection by community emotions. We propose a novel method to compute community-level emotion by considering the user authority in the community network. Then we present an effective emotion burst detection algorithm for the community emotion distribution sequence.

We evaluate our approach on real microblog data sets. Experimental results demonstrate that it is important to take into account the user authority when aggregating community emotion for

community-related event detection. Our emotion burst detection algorithm also achieves better performance than several baseline methods.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments and suggestions. This research is supported by the National High Technology Research and Development Program of China (Grant No. 2012AA011002), Natural Science Foundation of China (Grant No. 61300003). This research is partially supported by General Research Fund of Hong Kong (417112), Shenzhen Fundamental Research Program (JCYJ20130401172046450) and Shenzhen International Cooperation Funding GJHZ20120613110641217. This work is partially supported by Huawei Noah's Ark Lab, Hong Kong. The contact author of this paper, according to the meaning given to this role by Peking University, is Wei Chen.

References

- Apoorv Agarwal, Boyi Xie, Iliia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In *Proceedings of the Workshop on Languages in Social Media*, pages 30–38. Association for Computational Linguistics.
- Cuneyt Gurcan Akcora, Murat Ali Bayir, Murat Demiras, and Hakan Ferhatosmanoglu. 2010. Identifying breakpoints in public opinion. In *Proceedings of the First Workshop on Social Media Analytics*, pages 62–66. ACM.
- Sigal G Barsade and Donald E Gibson. 1998. Group emotion: A view from top and bottom. *Research on Managing Groups and Teams*, 1:81–102.
- Hila Becker, Mor Naaman, and Luis Gravano. 2010. Learning similarity metrics for event identification in social media. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 291–300. ACM.
- Adam Birmingham and Alan F. Smeaton. 2010. Classifying sentiment in microblogs: is brevity an advantage? In *CIKM*, pages 1833–1836.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.
- Dmitry Davidov, Oren Tsur, and Ari Rappoport. 2010. Enhanced sentiment learning using twitter hashtags and smileys. In *COLING (Posters)*, pages 241–249.
- Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, pages 1–12.
- Xia Hu, Jiliang Tang, Huiji Gao, and Huan Liu. 2013. Unsupervised sentiment analysis with emotional signals. In *WWW*, pages 607–618.
- Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *ACL*, pages 151–160.
- Jon M Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632.
- Jon Kleinberg. 2003. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397.
- Zhenhui Li, Ding Zhou, Yun-Fang Juan, and Jiawei Han. 2010. Keyword extraction for social snippets. In *Proceedings of the 19th international conference on World wide web*, pages 1143–1144. ACM.
- Kun-Lin Liu, Wu-Jun Li, and Minyi Guo. 2012. Emoticon smoothed language models for twitter sentiment analysis. In *AAAI*.
- Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167.
- Michael Mathioudakis and Nick Koudas. 2010. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pages 1155–1158. ACM.
- Thin Nguyen, Dinh Q. Phung, Brett Adams, and Svetha Venkatesh. 2013. Event extraction using behaviors of sentiment signals and burst structure in social media. *Knowl. Inf. Syst.*, 37(2):279–304.
- Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*.
- Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, 2(1-2):1–135.
- Alan Ritter, Oren Etzioni, Sam Clark, et al. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1104–1112. ACM.
- Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860.
- Eliot R Smith, Charles R Seger, and Diane M Mackie. 2007. Can emotions be truly group level? evidence regarding four conceptual criteria. *Journal of personality and social psychology*, 93(3):431.

- Michael Speriosu, Nikita Sudan, Sid Upadhyay, and Jason Baldridge. 2011. Twitter polarity classification with label propagation over lexical links and the follower graph. In *EMNLP*, pages 53–63.
- Jianshu Weng and Bu-Sung Lee. 2011. Event detection in twitter. In *ICWSM*.
- Jichang Zhao, Li Dong, Junjie Wu, and Ke Xu. 2012a. Moodlens: an emoticon-based sentiment analysis system for chinese tweets. In *KDD*, pages 1528–1531.
- Wayne Xin Zhao, Baihan Shu, Jing Jiang, Yang Song, Hongfei Yan, and Xiaoming Li. 2012b. Identifying event-related bursts via social media activities. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1466–1477. Association for Computational Linguistics.