Emerging Topic Detection for Organizations from Microblogs

Yan Chen*
State Key Laboratory of Software Development Environment
Beihang University, China
chenyan@cse.buaa.edu.cn

Hadi Amiri
School of Computing
National University of Singapore, Singapore
hadi@comp.nus.edu.sg

Zhoujun Li
State Key Laboratory of Software Development Environment
Beihang University, China
lizj@buaa.edu.cn

Tat-Seng Chua
School of Computing
National University of Singapore, Singapore
chuats@comp.nus.edu.sg

ABSTRACT

Microblog services have emerged as an essential way to strengthen the communications among individuals and organizations. These services promote timely and active discussions and comments towards products, markets as well as public events, and have attracted a lot of attentions from organizations. In particular, emerging topics are of immediate concerns to organizations since they signal current concerns of, and feedback by their users. Two challenges must be tackled for effective emerging topic detection. One is the problem of real-time relevant data collection and the other is the ability to model the emerging characteristics of detected topics and identify them before they become hot topics. To tackle these challenges, we first design a novel scheme to crawl the relevant messages related to the designated organization by monitoring multi-aspects of microblog content, including users, the evolving keywords and their temporal sequence. We then develop an incremental clustering framework to detect new topics, and employ a range of content and temporal features to help in promptly detecting hot emerging topics. Extensive evaluations on a representative real-world dataset based on Twitter data demonstrate that our scheme is able to characterize emerging topics well and detect them before they become hot topics.

Categories and Subject Descriptors

H.0 [**Information Systems**]: General; H.3.5 [**Information Storage and Retrieval**]: Online Information Services

General Terms

Algorithms, Experimentation

Keywords

Microblog Service, Emerging Topic Detection, Brand Monitoring, Organization Monitoring

1. INTRODUCTION

Microblog services such as Twitter and Weibo provide an essential platforms for users to convey their thoughts, exchange their opinions and share their experiences. One key reason leading to their popularity is their real-time nature. On these platforms, individuals update their status regarding various topics, spanning from "what are they doing" (Twitter), to " what are on their mind" (Facebook), and this is conveyed instantaneously to their friends. This greatly strengthens inter-personal exchange and cooperation.

Besides facilitating communications among individuals, microblog services also explicitly or implicitly contain rich information towards organizations, such as banks, universities, and government organizations, etc. Many organizations are keen on continually mining and analyzing these user-generated social data due to the following reasons. First, social data contains the interests, concerns and criticisms of their users, and provides pointers for organizations to improve their products or services. Second, social data implicitly contains invaluable market insights for the organizations. The primary foundation of these high-level applications is based on topic monitoring and tracking. Specifically, organizations would like to: (1) track the evolution of any identified relevant topics about them; and (2) be informed of any new emerging topics which are fast gathering momentum in microblogs.

*This work was done when the first author was a visiting scholar in the National University of Singapore.

SIGIR'13, July 28–August 1, 2013, Dublin, Ireland.

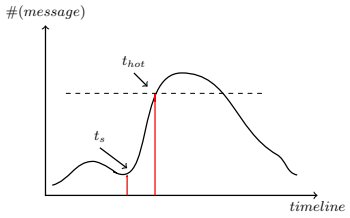


Figure 1: The key phases of an hot emerging topic.

Several threads of research have been done on emerging topic detection. They can roughly be grouped into three categories. The first is based on traditional topic detection methods to identify new terms [12, 6, 21]. The second is to utilize topic model to learn a dynamic word distribution or topic transition [24]. The last is to detect novel topic-related words by dictionary learning methods [11, 19]. However, the definition of "emerging" in previous literature focuses only on the novelty of topics, and they mainly model the novel words based on word co-occurrences within the topics. In this work, we extend the definition of "emerging" to incorporate temporal aspect of timeliness. In other words, we want to detect emerging topics that are not only novel, but also those that will become hot and viral in the near future. This presents an additional challenge to model the temporal characteristics of topics in real-time.

Figure 1 presents the evolution of a hot topic starting from the time that it is detected (t_s), to the time that it becomes hot (t_{hot}). The period from t_s to t_{hot} is known as the emerging phase. We expect to identify this topic as hot and emerging before t_{hot}.

However, emerging topic detection for organizations from microblog context faces several challenges. The first challenge is the dilemma of relevant data collection. This is a challenging issue as most live microblog services impose limits on the amount and frequency of data that can be crawled[1]. This, in conjunction with the low ratio of relevant data in microblog content, results in missing relevant data about the organizations. The second challenge is on modeling of topics with effective features to facilitate the detection of hot topics during the emerging phase.

To address these two challenges, we design a novel scheme to monitor and collect microblog messages (tweets) about organizations as well as detect the hot emerging topics promptly. It comprises two stages as illustrated in Figure 2. The first stage aims to gather rich social data with good coverage for a designated organization. Specifically, given a specific organization, we collect data in multiple aspects from four sources including fixed keywords, emerging keywords, known accounts and key users of the organization. All the crawled data is then sent to a binary SVM classifier, which discriminates the relevant tweets from the huge amount of irrelevant ones. During this process, an organization user network is also maintained based upon the existing relationships[2] between users within the organization. The second stage first employs the well-known incremental clustering algorithms [25, 2] to discover topics in real time. It then analyzes the emerging topic-related features including user authority, tweets influence, and organization attributes such as the

[1]This limit is, for each request, "up to 1%" of Firehose tweets for the streaming API of Twitter.

[2]The existing relationship includes the follower, followee and friends, etc.

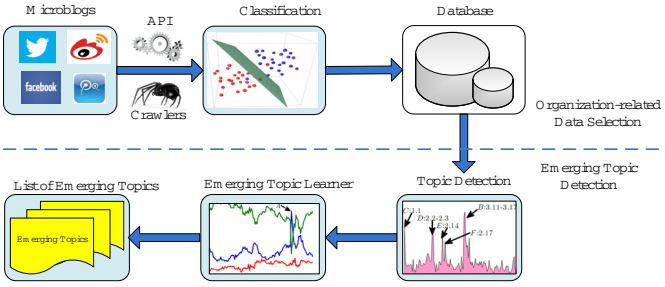


Figure 2: The overview of our framework.

emerging keywords. These features are then incorporated into the topic learner to identify hot emerging topics in a timely manner.

The main contributions of this paper are as follows:

- We design a crawling strategy that incorporates multiple aspects of microblog data to obtain more comprehensive organization related data. Based on the collected data, we detect organization related topics.
- We build two effective semi-supervised models to detect hot emerging topics for an organization in a timely manner. To the best of our knowledge, this is the first that targets effective feature extraction for hot emerging topics detection.

The rest of this paper is organized as follows: Section 2 introduces the related work of emerging topic detection. Sections 3 and 4 respectively detail organization-related data selection and emerging topic detection. Our experimental results are presented in Section 5, followed by concluding remarks in Sections 6.

2. RELATED WORK

The popularity of microblog portals like Twitter has enabled hot topics to be quickly propagated to a large number of users over wide geographical regions. Research on detecting emerging and evolving [8, 25] topics[3] of live tweet streams has gained much interest in recent years, and has been applied to a wide variety of applications [7], such as detecting emergencies like earthquakes [20], predicting political election outcomes [22], mining topic and evolution [9], discovering controversial topics from twitter [17], and so on. There are several lines of research in this direction.

One line of research is based on the traditional topic detection approach. From the feature pivot aspects, some keywords based approaches [12] work well on mining tweets about specific topics. While high frequency of terms may be a good indicator for hot topics or trends, it does not identify new of emerging trends. Cataldi et al. [6] defined emerging keywords as those which are frequently used in a given time period, but not in previous ones. They presented an approach to identify emergent keywords and utilized them together with frequently co-occurring words as emerging topics. From the document-pivot aspects, Sayyadi et al. [21] created a keyword graph, and used it to cluster tweets based on various distance and similarity metrics.

Probabilistic latent semantic analysis (PLSA) and Latent Dirichlet Allocation (LDA) [5] are probabilistic methods

[3]Here, topic can be exchangeable with event.

that have found remarkable success in building topic models of static text. Variants of PLSA and LDA have been proposed for online and dynamic topic modeling [4].Wang et al. [23] took advantage of temporal information, and tried to model the topics continuously over time. They learned the dynamic word distributions and trends of topics over time. Wang et al. [24] proposed a Temporal-LDA or TM-LDA method to mine streams of social text such as the Twitter stream for an author, by modeling the topics and topic transitions that naturally arise in such data. Different from the work of [23], TM-LDA focuses more on learning the relationship among topics.

Another line of related research is on dictionary learning and non-probabilistic matrix factorizations based methods. Kasiviswanathan et al. [11] proposed a two stage approach based on the detection and clustering of novel user-generated content. They derived a scalable approach by using the alternating directions method to effectively solve the resulting optimization problems. By extending the above work, Saha and Sindhwani [19] adapted Non-negative Factorization to learn trending topics in the context of social media. They showed that better topic modeling performance can be achieved, when the continuity between topics matrices in consecutive time stamps is taken into account.

Previous research on emerging topic [18, 1] detection mainly focused on keywords and textual content, whereas we aim to find emerging topics with respect to an organization. The major difference is that for entities like organizations, in addition to textual content, user association to the organization and social relations among users of the organization will greatly affect the detection of emerging topics for the organization. These features have not been utilized before due to the focus on general emerging topics.

3. ORGANIZATION-RELATED DATA SELECTION

To perform high-order analytics, it is desirable to collect a relatively complete set of relevant data for the target organization in an effective way. However, such a task is often overwhelmed by the tremendous amount of relevant as well as irrelevant data. To ensure comprehensive data collection, two interconnected observations can be made: (a) users related to organizations are more likely to post tweets related to the organization; and (b) tweets on organization often contain organization related keywords. These two observations enable us to generate descriptive keywords and cues, such as fixed keywords, dynamic keywords, known accounts, and organization keyusers. Accordingly, we design four intelligent crawlers to comprehensively crawl organization relevant data from multiple aspects, as shown in Figure 3. The generation of four aspects of sources are detailed in the following subsections.

3.1 Fixed and Dynamic Keywords Sources

Given the name of the organization, we first manually select a few fixed keywords that uniquely identify the organization, such as the name of the organization, the key terms of its brands, and the name of its CEO, etc. These fixed keywords are used in the streaming based *Fixed Keyword Crawler*.

To elicit a live and more diverse set of relevant tweets about the organization, typically those that do not contain

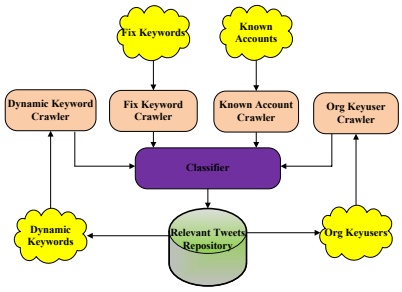


Figure 3: Crawling of data from multi-aspect sources.

the organization keywords, we extract a list of temporally relevant emerging terms about the organizations at each time point t. Emerging terms are defined as those newly introduced terms that are able to represent emerging topics about the organization. To identify these terms, two sets of foreground (S^t_{for}) and background (S^t_{bak}) tweets at each time point t are constructed. All the relevant tweets in the current time slot $[t-T, t]$ are regarded as S^t_{for} tweets, where T is the time interval. While S^t_{bak} covers all the tweets sent during the three time slots of: $[t-2T, t-T]$ of current day, $[t-T, t]$ of previous day, as well as $[t-T, t]$ of one week ago. This definition aims to filter out the time related periodical terms, such as morning, Friday, etc. The vocabulary sets of these two sets of tweets are respectively denoted as, $W^{for} = \{w^{for}_1, w^{for}_2, ...\}$ and $W^{bak} = \{w^{bak}_1, w^{bak}_2, ...\}$. Our goal is to identify the terms that have distinctively different distributions in S^t_{for} and S^t_{bak}. This signals that the term is changing its behavior over time. Specifically, in terms of statistics, given two distributions of a term w_i in S^t_{for} and S^t_{bak}, we expect to disprove the null hypothesis that the two distributions are drawn from the same distribution function to a certain level of significance. Those terms with rising frequencies are the potential emerging keywords, whereas those with decreasing frequencies will gradually disappear. In this work, we use the chi-square test [14] to compare two distributions due to its efficiency and ability to handle rapidly evolving microblog contents. Formally, for each word $w_i \in W^{for}$, its chi-square distribution is:

$$\chi^2_i = \begin{cases} \frac{(f_i - b_i)^2}{b_i} + \frac{[(100-f_i)-(100-b_i)]^2}{100} & \text{if} \quad f_i > b_i; \\ 1 & \text{if} \quad otherwise. \end{cases} \tag{1}$$

where f_i and b_i are the normalized word frequency values of w_i in S^t_{for} and S^t_{bak} respectively, defined as:

$$f_i = 100 \times \frac{w^{for}_i}{\sum_{\forall i} w^{for}_i}, \quad and \quad b_i = 100 \times \frac{w^{bak}_i}{\sum_{\forall i} w^{bak}_i}.$$

Based on the value of χ^2, a list of ordered terms is generated, and the terms in the top N positions are truncated as the dynamic keywords. Each dynamic keyword will be concatenated with the acronym of the target organization to form a specific query for data collection.

3.2 Known Accounts Source

Similar to fixed keywords, we manually identify a set of organization related accounts. These are typically official accounts of the organization on microblog platform that often post relevant tweets about the target organization, such

as news about the target organization. These known accounts are monitored by the streaming based *Known Accounts Crawler*.

3.3 Org Keyusers Source

The above mentioned three kinds of crawlers explicitly identify and collect data about the target organization. However, they overlook some important tweets posted by the users related to the organization. The tweets are relevant to the target organization, but in implicit form, i.e., not containing the fixed or emerging keywords. *Org Keyusers Crawler* intends to plug this gap by exploring the sources from the perspective of users. At time t, given a time interval T (such as 24 hours), we obtain a subset of users $U_{\Delta t}$ who post at least one relevant tweets in the time window $\Delta t = [t-T, t]$. Here, we regard org keyusers as those people who post more about the organization and have many followers (a larger influence) in the time window Δt. As mentioned before, an organization user relationship graph G_0 is constructed in real time. Nodes of G_0 are obtained from the known accounts, and contain all the users who posted at least one organization relevant tweets, as well as their friends and followers. Edges of G_0 are obtained by crawling the social relationships between them. In addition, we want to incorporate their activity degree during Δt. The activity degree of a user is proportional to the number of tweets the user sent during Δt. We can compute the authority score of user u_i by incorporating the activity degree of user into graph G_0 as follows.

$$auth^k(u_i) = \alpha \sum_{u_j \in follower(u_i)} \frac{auth^{k-1}(u_j)}{|following(u_j)|} + (1-\alpha)\frac{|Tw_{\Delta t}^{u_i}|}{|Tw_{\Delta t}|}. \quad (2)$$

where $\alpha \in (0,1)$ is a damping factor, $following(u_j)$ stands for a set of users that u_j follows, $Tw_{\Delta t}$ represents the relevant tweets in Δt time slot, and $Tw_{\Delta t}^{u_i}$ is the relevant tweets of u_i in Δt. We calculate users' authority score in an iterative manner, i.e., when we calculate the authority value of u_i in the kth iteration, we utilize u_j's $(k-1)$th authority score. We then rank users in $U_{\Delta t}$ by their authority scores and the top N users are selected as the org keyusers.

3.4 Two-class SVM Classification

The data collected from the four sources are a mix of relevant and irrelevant tweets to the organization. In order to filter out the noisy data, we utilize a standard two-class SVM classifier. For the training data, we regard all the tweets from known account source and fixed keyword (rule based to select) source as relevant.

4. EMERGING TOPIC DETECTION

We first present an incremental clustering method to discover topic collections. Through extracting features from topic and organization views, we train two semi-supervised hot emerging topic learners.

4.1 Topic Detection

For our real-time scenario, we need to handle live and large volume of tweets about an organization to detect topics without any prior knowledge of the number of topics, since the topics are constantly evolving and growing in size. Online or incremental clustering algorithms, which are able to handle a constant stream of new tweets, are desirable in our setting, where new tweets are continually being produced. We employ a single-pass incremental clustering algorithm [3] with a threshold τ. At each time t, and within a time interval T, we obtain all tweets D during [t-T, t] in a time order. Such a clustering algorithm considers each tweet in turn and determines the suitable cluster assignment based on a similarity function. The algorithm considers each tweet d_i in order, and computes its similarity $(d_i, Center_j)$ against each existing cluster C_j. If the maximum similarity value is greater than τ, the tweet will be distributed to the cluster, meanwhile the clustering center will be updated. Otherwise, we will generate a new cluster and cluster center. The details of this algorithm are shown in Algorithm 1.

Algorithm 1 Incremental Clustering for Topic Discovery

```
Input: tweet sets D, topic cluster set C, cluster center set Center, and threshold τ.
Output: update topic clusters C, and update cluster centers Center.
Process:
if C = ∅ then
    random select N tweets from D and add into C and Center.
end if
initialize max, tmp_C, tmp_center.
for d_i ∈ D do
    for center_j ∈ Center do
        compute Cosine Similarity sim between center_j and d_i.
        if sim > max then
            max = sim, tmp_C = C_j, tmp_center = center_j.
        end if
    end for
    if max > τ then
        distribute d_i to cluster tmp_C, and update tmp_center.
    else
        new cluster and centroid and add to C and Center.
    end if
end for
return C and Center.
```

4.2 Topic Related Features

In Section 3, we have extracted emerging keywords and org keyusers for the target organization from a global view point. To infer the importance of a topic within an organization, we need to examine also the influence of tweets and users at the local (topic) level.

4.2.1 Topical User Authority

Given a topic tp, there is a set of users related to tp, which is denoted as $U_{tp} = \{u_1, u_2, ..., u_i, ..., u_m\}$. We observe that the authority score of a user with respect to a specific topic $u_i \in U_{tp}$ is related to three factors. u_i will have a larger influence on topic tp if u_i has: (a) posted many tweets about topic tp; (b) posted more tweets retweeted by other users in U_{tp}; and (c) more followers in U_{tp}. Based on this observation, at time t, we can compute the authority score for each topic user $u_i \in U_{tp}$ as follows.

$$auth_{tp}(u_i) = \beta \frac{r_{u_i}}{\sum_j r_{u_j}} + \varphi \frac{f_{u_i}+1}{\sum_j f_{u_j}} + \omega \frac{q_{u_i}+1}{\sum_j q_{u_j}}, \quad (3)$$

where r_{u_i} is the total number of relevant tweets posted by u_i; f_{u_i} is the total number of u_i's followers who exist in U_{tp}; q_{u_i} is the total number of u_i's relevant tweets that has been retweeted by other users; and β, φ and ω are weighting parameters i.e., $\beta + \varphi + \omega = 1$.

4.2.2 Topical Tweet Influence

We can also find a set of tweets related to a topic tp, which are defined as $Tw_{tp} = \{tw_1, tw_2, ..., tw_i, ..., tw_n\}$. These tweets are posted by users U_{tp}. There are two intuitions: (a) if a tweet tw_i has a strong influence, it should be propagated to a large scope and be retweeted by a relatively higher number of times; and (b) if the tweet is posted by a topic authority user, it should also have the potential to influence more users. Thus, given a tweet tw_i in topic tp, we employ the number of retweets and authority of users to evaluate its influence, defined as follows.

$$auth_{tp}(tw_i) = \log(1 + auth_{tp}(u_{tw_i})) + \sum_{u \in U_{rtw_i}} \log(1 + auth_{tp}(u)), \tag{4}$$

where $auth_{tp}(tw_i)$ is the influence of tweet tw_i; $auth_{tp}(u_{tw_i})$ is the author of tw_i's authority; and U_{rtw_i} represents the user group that retweets tw_i.

Let $W_{tp} = \{w_1, ..., w_i, ..., w_r\}$ be the set of words that appear in topic tp. For each word w_i in topic tp, we compute its weight $Weight_{tp}(w_i)$ through the influence of tweets that it appears in, as:

$$Weight_{tp}(w_i) = \frac{\sum_{\forall tw_j \in Tw_{tp} \wedge w_i \in tw_j} auth_{tp}(tw_j)}{\sum_{\forall w \in W_{tp}} \sum_{\forall tw \in Tw_{tp} \wedge w \in tw} auth_{tp}(tw)}. \tag{5}$$

We use $Weight_{tp}(w_i)$ to rank the list of topic-related keywords.

4.3 Hot Emerging Topic Learner

In order to identify the hot emerging topics from the topic collection at each time t, we should analyze and extract the emerging features of topics. Given a target organization at time t, from an organization view point, we extract key users and emerging keywords for the target organization, and from a local topic view point, we calculate the authority of users, tweets and keywords for a specific topic. By combining these two views, we extract six representative features for each topic at time t to train the emerging topic learner. The six features with respect to topic tp are defined as follows.

- f_1 is the rate of increase of user number,

$$f_1 = \frac{|U^t|}{\sum_{x=0}^{t} \frac{1}{t-x+1}|U^x|}. \tag{6}$$

- f_2 is the rate of increase of tweets number,

$$f_2 = \frac{|Tw^t|}{\sum_{x=0}^{t} \frac{1}{t-x+1}|Tw^x|}. \tag{7}$$

- f_3 is the rate of increase of re-tweets number,

$$f_3 = \frac{|Rtw^t|}{\sum_{x=0}^{t} \frac{1}{t-x+1}|Rtw^x|}. \tag{8}$$

- f_4 is the overlap between org keyusers and top N influential topic users,

$$f_4 = \frac{\#(ku_{tp} \cap ku)}{\#ku_{tp}}. \tag{9}$$

- f_5 is the overlap between org keywords and top N influential topic keywords, and

$$f_5 = \frac{\#(kw_{tp} \cap kw)}{\#kw_{tp}}. \tag{10}$$

- f_6 represents the rate of increase of influence of the accumulated weight of tweets,

$$f_6 = \frac{|A^t|}{\sum_{x=0}^{t} \frac{1}{t-x+1}|A^x|}, \tag{11}$$

where $A = \frac{\sum_{tw \in Tw_{tp}} auth_{tp}(tw)}{|Tw_{tp}|}$.

These six features were chosen to discriminate an hot emerging topic from the topic collection. For an emerging topic, the number of participating users, the increasing rate of tweets, and the number of retweets are expected to be distinctively higher than the normal topics and that of previous time. Moreover, there are some clues that the key topic users and keywords would have a large overlap with the current org keyusers and dynamic keywords. As tweets are likely to be retweeted, the increasing rate of accumulated weight of tweets are comparatively higher than those in previous time period too.

The design of our learners considers two factors in the microblog domain. (1) Since there are hundreds of clusters for an organization at each time t, the labeling process can be time consuming and labor intensive for all training data. Hence there will be the problem of insufficient training data. (2) There is a problem of imbalance of positive (emerging phase) and negative (not emerging topics or not emerging phase for emerging topics) data, since the vast majority of data are negative instances. Hence the learner must be able to achieve good accuracy in the face of imbalance training data. The above two factors require that the chosen learners should work well under the conditions of sparse and imbalance training data scenario. Thus, we adopt a co-training learner and a semi-supervised ensemble learner for our learning task. We define the training data as $TrainSet = \{(X_l, Y), X_u\}$, where X_l and X_u denote the labeled and unlabeled dataset respectively; and Y represents the label set ($positive(p)$ and $negative(n)$). Our aim is to learn a learner $c : X \mapsto Y$.

4.3.1 Co-training Learner

Algorithm 2 Co-training Learner

1: **Input:** $TrainSet = \{(X_l, Y), X_u\}$, let $L_1 = L_2 = \{\{(X_l, Y_p)\}, \{(X_l, Y_n)\}\}$ and $|\{(X_l, Y_p)\}| = |\{(X_l, Y_n)\}|$.
2: **Output:** $TrainSet = \{(X_l, Y), (X_u, Y)\}$ and two learners c_1 and c_2.
3: **Process:**
4: **while** Unlabeled dataset $\{X_u\} \neq \emptyset$ **do**
5: Train two classifiers: v(1) c_1 from L_1, and v(2) c_2 from L_2.
6: Classify unlabeled data with c_1 and c_2 separately.
7: Add c_1's k-most-confident (x, c_1(x)) to labeled data L_2.
8: Add c_2's k-most-confident (x, c_2(x)) to labeled data L_1.
9: Remove these from the unlabeled dataset $\{X_u\}$.
10: Update L_1, sampling removing negative instances to keep $|\{(X_{l:L_1}, Y_p)\}| = |\{(X_{l:L_1}, Y_n)\}|$.
11: Update L_2, sampling removing negative instances to keep $|\{(X_{l:L_2}, Y_p)\}| = |\{(X_{l:L_2}, Y_n)\}|$.
12: **end while**
13: return c_1 and c_2.

Features of topic tp at time t are divided into two orthogonal views: v(1) the number increasing rate features (f_1, f_2, and f_3); and v(2) the overlap features (f_4 and f_5) and the accumulated weights of increasing rate feature (f_6). We assume that both v(1) and v(2) are orthogonal to each other, and they are sufficient to train reasonably strong classifiers. We then train two basic SVM classifiers based on these two views. The training process is described in Algorithm 2. It is worth noting that in order to account for the imbalance data scenario, the construction of training instance set L (L_1 and L_2) are different from the previous co-training approaches. Here we keep all the positive instances in L, while sample only an equal number of labeled negative instances from the training data into L at each iteration.

4.3.2 Semi-supervised Ensemble Learner

As an alternative to co-training, we employ the voting based ensemble learning to train a semi-supervised classifier. Three classifiers (Decision Tree, SVM, and Naive Bayesian) are chosen to learn from the labeled training data to predict the unlabeled training data independently. Their results are used to vote for each unlabeled data instance. The consistent data will be added to the next training iteration, until convergence. Algorithm 3 describes the training process.

Algorithm 3 Ensemble Learner

Input: $TrainSet = \{(X_l, Y), X_u\}$, $L_1 = \{\{(X_l, Y_p)\}, \{X_l : L_1, Y_n\}\}$, $L_2 = \{\{(X_l, Y_p)\}, \{X_l : L_2, Y_n\}\}$, $L_3 = \{\{(X_l, Y_p)\}, \{X_l : L_3, Y_n\}\}$ and if $|\{(X_l, Y_p)\}| = N$, $|\{(X_l : L_1, Y_n)\}| = |\{(X_l : L_2, Y_n)\}| = |\{(X_l : L_3, Y_n)\}| = N$.
Output: $TrainSet = \{(X_l, Y), (X_u, Y)\}$ and ensemble learner c.
Process:
while Unlabeled dataset X_u is used up **do**
  Train three classifiers: c_1 from L_1, c_2 from L_2 and c_3 from L_3.
  Classify unlabeled data with c_1, c_2 and c_3 separately.
  Voting.
  Add consistent results set to L_1, L_2 and L_3.
  Remove these from the unlabeled dataset X_u.
  Update L_1, sampling removing negative instances to keep $|\{(X_{l:L_1}, Y_p)\}| = |\{(X_{l:L_1}, Y_n)\}|$.
  Update L_2, sampling removing negative instances to keep $|\{(X_{l:L_2}, Y_p)\}| = |\{(X_{l:L_2}, Y_n)\}|$.
  Update L_3, sampling removing negative instances to keep $|\{(X_{l:L_3}, Y_p)\}| = |\{(X_{l:L_3}, Y_n)\}|$.
end while
return c.

4.4 Overall Algorithm for Hot Emerging Topic Detection

The overall process of hot emerging topic detection schema is detailed in Algorithm 4. At each time t, topics are discovered by Algorithm 1. For each topic, we extract the desired features and classify it using algorithm 2 or 3. If it is an emerging topic and the cluster id does not exist in $ETSet$, we will record the id and t into $ETSet$, which holds the list of emerging topic candidates.

Algorithm 4 Overall Algorithm of Emerging Topic Detection

Input: Start time point t_0.
Output: Emerging topic list $ETSet$.
Process:
$t = t_0$, and get tweets during $t - \Delta t$ as D.
while $D \neq \emptyset$ **do**
  Detect topic using algorithm 1, and get topic cluster set C at t.
  for each topic $C_i, i = 1, ...$ **do**
    Extract features of C_i.
    Label C_i using algorithm 2 or 3 into p or n.
    if C_i labeled with p and $id \notin ETSet$ **then**
      Add id and time point t of C_i into $ETSet$.
    end if
  end for
  $t = t + \Delta t$.
  Get tweets as D from $t - \Delta t$.
end while
return $ETSet$.

5. EXPERIMENTS

5.1 Datasets, Ground Truth and Settings

In order to evaluate our approach, experiments were conducted on a real life dataset crawled for three organizations of different nature in Singapore. They are *StarHub*[4], Development Bank of Singapore (*DBS*)[5] and National University of Singapore (*NUS*)[6]. These three organizations cover local telecommunication company (*StarHub*), the university (*NUS*), and a cross-border bank (*DBS*). We use twitter API[7] to collect the datasets that contain 51K, 130K, and 142K tweets from 15K, 44K and 36K users for the above three organizations respectively. Because there is no benchmark for our emerging topic detection task, we generate the ground truth of hot emerging topics by adopting the following procedures.

(1) We manually align the topics with online news and labeled the topic as hot emerging topic if its relevant tweets number at least doubled in the future 24 hours after its detection. Finally we obtain 24, 17 and 5 hot emerging topics respectively for the three organizations.

(2) For each hot emerging topic, we label two time points: t_s and t_{hot}, the start of the topic and the time when topic becomes hot, respectively (see Figure 1). t_s is the first time slot in which the emerging topic is detected. t_{hot} represents a time slot in which tweets number exceeds a threshold. A middle point t_{mid} between t_s and t_{hot} is also computed automatically.

The statistics about the datasets for the three organizations are detailed in Table 1. We list the time duration of data collection as well as hot emerging topic numbers. Table 1 also lists the initial period of data that we used for training. In our system, the numbers of org keyusers and dynamic keywords are set to 100 and 50 respectively, in order to limit the crawling resources required to monitor the keyusers and keywords. We empirically set the thresholds $\tau = 0.7$, $\alpha = 0.6$ and $\beta = \varphi = \omega = 0.33$. The two-class SVM and dynamic keywords mining are performed at intervals of every half an hour, while the org keyusers mining are performed at every 24 hour intervals. The time interval used for topic and emerging topic detection (Algorithm 4) is 1-hour.

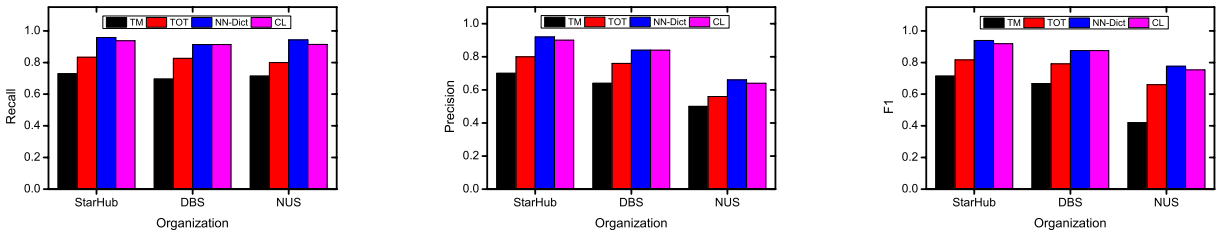
5.2 Results and Analysis

Because of the lack of space, we only detail our experi-

[4] http://www.starhub.com/
[5] http://www.dbs.com.sg/
[6] http://www.nus.edu.sg/
[7] https://dev.twitter.com/docs/api

Table 1: Statistics for organizations

Organization	Time Duration	#Tweets	#Users	#Emerging Topic	Training Time Duration	#Training Emerging Topic
StarHub	10 Oct-9 Nov, 2012	51,708	15,792	24	10-22 Oct, 2012	10
DBS	15 Oct-14 Nov, 2012	130,791	44,454	17	15-28 Oct, 2012	8
NUS	14-27 Oct, 2012	142,091	36,973	5	14-20 Oct, 2012	2



(a) *Recall* (b) *Precision* (c) F_1

Figure 4: Performance of topic Detection

ments on topic detection and emerging topic detection only. The effectiveness of our data collection strategy will not be discussed here. It is tested indirectly through the accuracy of hot emerging topic detection experiments.

5.2.1 Performance of Topic Detection

The performance of emerging topic detection model is based on the premise that the approach can fetch more organization related topics. In this work, we compare our topic detection approach with several baselines to demonstrate its effectiveness:

- **TwitterMonitor (TM)** [15]: It is a system that performs trending topic detection over the Twitter stream. We implemented two core algorithms of this paper (finding and grouping bursty keywords). Here we set the value of k in this method to 50, and the window size is set to $W = 1h$.
- **Topics over Time (TOT)** [23]: This is a LDA-based topic model that explicitly models the time jointly with word co-occurrence patterns. Here we set the topic number $|T| = 50$, hyper-parameter $\alpha = 1$, and $\beta = 0.1$.
- **NN-Dict** [11]: This is a dictionary learning based scheme. It utilizes the nearest neighbor approach to detect novel documents and then create emerging topic clusters by a dictionary learning technique. Here we set the topic numbers to 50, and the parameter $\lambda = \frac{1}{100}$.

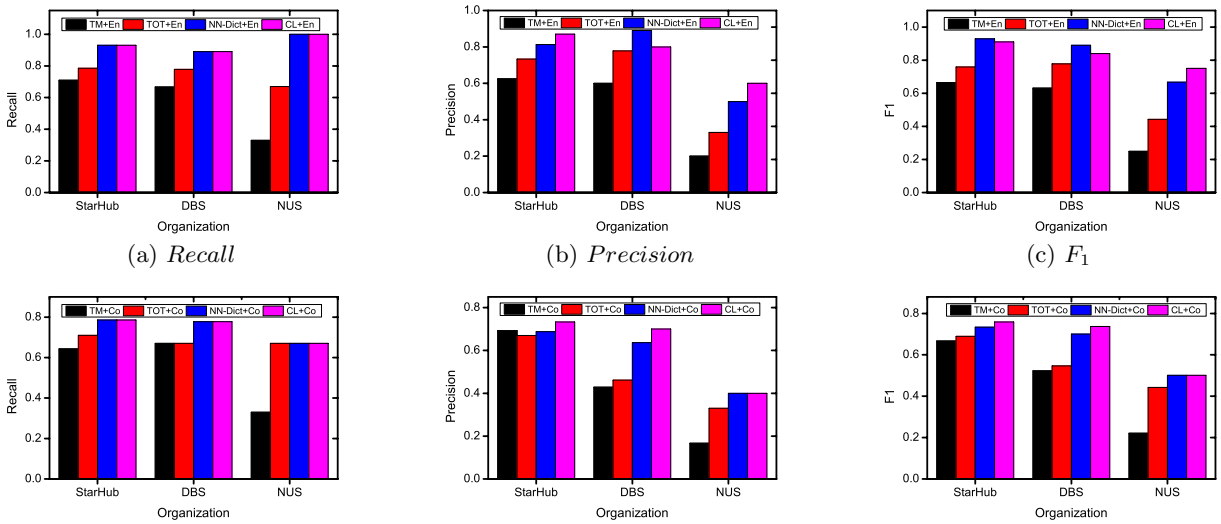
For ease in evaluation, we utilize two sources of information to construct the ground truth for topic detection. The first is the emerging topics as shown in Table 1. The second is based on the idea of pooling by aggregating all the new topics detected by the four methods (CL, TM, TOT, and NN-Dict). We asked three people to evaluate the new topics detected and come up with the final set as the ground truth. The ground truth contain 34, 31 and 24 topics for *StarHub*, *DBS*, and *NUS*. In this work, we use several widely-used classifier performance metrics for evaluation [13]: recall, precision and F_1. Figure 4 presents the performance of CL and three baselines for three organizations.

It can be seen from the Figure that NN-Dict and CL achieve better performance than TOT and TM across all evaluation metrics for all organizations. The main reason for the poor performance of TM is that it constructs an undirected graph and groups bursty keywords by maximum connected components. This probably leads to multiple semantically separate topics being merged together and therefore produces very large topics that reflect very little about the real world topics. For TOT, it discovers topics clusters and their evolution over time. However, it often identifies old or non-informative topics as compared to the previously appearing ones. Overall, our method and NN-Dict can detect over 90% of topics with a F_1 measure of 70%. Finally, it is worth noting that the higher recall but lower F_1 of *NUS* suggests that there are not many topics happened in *NUS* as compared to *StarHub* and *DBS*.

5.2.2 Performance of Emerging Hot Topic Detection

This subsection aims to detect whether a newly found topic is emerging and will become a hot topic at a later stage. Hence there is a temporal dimension to this task, that the hot topic should be detected during the emerging phase as shown in Figure 1. Here we want to test the ability of the method to identify a topic as "hot topic" before a time T_L. For evaluation purpose, we select two time limits: a stringent one with $T_L = t_{mid}$, and a more relax one with $T_L = t_{hot}$ (see Figure 1). If the topic is identified as hot topic before T_L, it is considered a positive detection; otherwise it is considered a failure.

As our baselines are designed for novel topic detection, but not emerging topic detection, we need to incorporate time dimension into these methods. Here, we incorporate our two emerging topic learners, Co (Co-training learner) and En (Semi-supervised ensemble learner) with our proposed incremental learning method (CL) and the three baselines, giving rise to 8 combination of methods as shown in Figure 5, which list the F_1 measure of hot emerging topic detection when $T_L = t_{hot}$. Three main observations can be drawn from Figure 5. First, the topic detection methods incorporating semi-supervised ensemble learner (CL+En, TM+En, TOT+En, and NN-Dict+En) generally perform much better than those incorporating the co-training learner (CL+Co, TM+Co, TOT+Co, and NN-Dict+Co). The reason for the poor performance of Co is because the emerging features are split into two views with each view being weaker than the overall combined feature set. Second, NN-Dict and our proposed CL incorporating the emerging topic learners perform much better than the other two methods. This shows that



(a) *Recall* (b) *Precision* (c) F_1

(d) *Recall* (e) *Precision* (f) F_1

Figure 5: Performance of emerging topic Detection when $T_L = t_{hot}$

Table 2: Performance of emerging topic detection when $T_L = t_{hot}$

Methods	Organization	recall	precision	F_1
CL+En	*StarHub*	**0.93**	**0.87**	**0.90**
CL+TSVM		0.86	0.75	0.80
CL+Semi-NB		0.86	0.71	0.77
CL+En	*DBS*	**0.89**	**0.80**	**0.84**
CL+TSVM		**0.89**	0.73	0.80
CL+Semi-NB		**0.89**	0.67	0.70
CL+En	*NUS*	**1.00**	**0.60**	**0.75**
CL+TSVM		**1.00**	0.50	0.67
CL+Semi-NB		**1.00**	0.42	0.73

Table 3: Performance of emerging topic detection when $T_L = t_{mid}$

Methods	Organization	recall	precision	F_1
CL+En	*StarHub*	**0.71**	**0.83**	**0.77**
CL+TSVM		**0.71**	0.71	0.71
CL+Semi-NB		**0.71**	0.67	0.69
CL+En	*DBS*	**0.78**	**0.78**	**0.78**
CL+TSVM		**0.78**	0.70	0.74
CL+Semi-NB		**0.78**	0.64	0.70
CL+En	*NUS*	**0.67**	**0.50**	**0.57**
CL+TSVM		**0.67**	0.40	0.50
CL+Semi-NB		**0.67**	0.40	0.50

a strong topic detection baseline is needed to achieve good performance in hot emerging topic detection. Here we also observe that the recall performance is better than precision for most cases. This is important as in real applications, the ability to flag all possible hot topics is essential for organizations to handle all possible eventualities. Third, it is observed that our CL performs comparably in recall as compared to NN-Dict method but much better in precision, resulting in superior F_1 measure. In general, our method can detect close to 90% of hot topics with a precision of over 70%. This is an encouraging results for hot emerging topic detection.

5.2.3 *Efficiency Analysis*

The efficiency problem is very important for a real-time schema. In this subsection, we evaluate the average run time

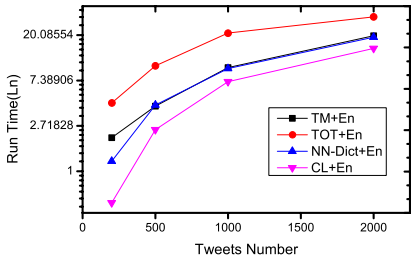


Figure 6: The impact of incoming tweets size on Run Time

with different sizes of incoming tweets for different methods. We run the algorithms on a computer with 2.83GHz Intel(R) Core 2 Quad CPU, and 4GB memory. The experimental result of logarithm deviation of run time varying with increasing tweets size is shown in Figure 6. It can be seen from the figure that the run time increases with rising incoming tweet number. TOT performs poorly as it needs to sample the tweets and iterate to convergence in several steps in each iteration. For TM, it spends much time on bursty keyword graph mining. The main cost of NN-Dict is to maintain and update the term dictionaries and to find novel clusters by clustering methods. Overall, incremental clustering is the most efficient for real-time scenarios, as it explicitly maintains topics and related tweets for further analysis of emerging topic features.

5.2.4 *Comparison with State-of-the Arts Methods*

Because of the poor performance of CL+Co, it will not be included in further experiments. Here we conduct further experiments to compare the performance of CL+En against two start-of-the-arts methods: Transductive SVM (TSVM) [10] and Semi-supervised Naive Bayesian classifiers (Semi-NB) [16]. As with the previous approaches, the parameters of the methods are carefully tuned. And we test the ability of the methods to detect emerging topic at $T_L = t_{hot}$ and $T_L = t_{mid}$.

Tables 2 and 3 show the precision, recall and $F1$ measure of the methods for $T_L = t_{hot}$ and $T_L = t_{mid}$ respectively.

From the Tables, it is observed that our proposed incremental clustering with ensemble learner (CL+En) performs the best against all the other methods, with a high F_1 of 0.90. The results show that our semi-supervised ensemble learner can inherit the advantages of each learner to make up for the shortcomings of a single learner. It is also observed that the recall is higher than precision generally, which is an important attribute of emerging topic detection methods. Table 3 presents the comparison results when $T_L = t_{mid}$, which has a much more stringent criterion than that in Table 2. It naturally shows a performance reduction for all the learners. Finally, it is observed that performance of *StarHub* and *DBS* is better than that of *NUS* for the classification based metrics shown in Tables 2 and 3. The poorer performance of *NUS* is mainly due to the low number of tweets and lack of labeled data during the emerging phase.

5.2.5 Emerging Feature Analysis

In this subsection, experiments are carried out to investigate the influence of various emerging features on F_1. We progressively remove one feature for our semi-supervised ensemble learner. The experimental results on *StarHub*, *DBS*, and *NUS* are illustrated in Figure 7. Here, "f" on x-axis means all the features are used for the learner, and "-f*" means that all the features except feature "f*" are used in the ensemble learner. It is observed that the performance of F_1 on the three organizations are all degraded to a certain degree with the absence of some features. This observation verifies that all features have some contributions to the learners towards achieving good performance. However, it can be observed that the absence of features f_2, f_3 and f_6 cause the most degradation in performance. This shows that the rates of increasing tweets number (f_2), re-tweets number (f_3), and overall accumulated influence of tweets (f_6) are very important, and comparatively, user factors are less influential. This could be due to the fact that topic has only a small number of users. More testing needs to be done for future work. Finally, as compared to the other two organizations, *NUS* shows a much larger performance degradation with the absence of the above three factors. This is again due to the small training size for *NUS*.

5.2.6 Imbalance Impact

As mentioned in Section 4.3, training data imbalance is also a challenging problem in our task. In this section, we briefly look into the impact of imbalance training data for the two semi-supervised learners. The F_1 performance of two learners for the balance and imbalance settings is shown in Figure 8. It can be seen from the Figure that the F_1 performance of the two learners for the imbalance data degrades greatly as compared to that with balance data. As imbalance data makes our learners bias towards negative instances, which will increase the false positive errors for the methods. This is worse for the co-training learner. As false positive error will be accumulated into the next iteration, the co-training classifier is more easily affected by the noisy data.

6. CONCLUSIONS

In this paper, we proposed a real-time framework for detecting hot emerging topics for organizations in social media context. First, we introduced four sources of crawling organization data from multiple perspectives to ensure a more complete set of dataset for the target organization. Second, we discovered emerging topics and extracted emerging features from both the organization and topic perspectives. Thirdly, we developed semi-supervised learners to facilitate timely identification of hot emerging topics for organizations. We demonstrated the effectiveness of our proposed framework by comparing them with the state-of-the-arts methods. Empirical evaluation on the Twitter datasets on three organizations (*StarHub*, *DBS* and *NUS*) illustrated the effectiveness of the proposed emerging topic detection framework.

One can envision several directions for future work. While the current work is based on organizations to detect emerging and evolving topics, we can extend our framework to more general entities, such as the people and location, etc. The other important direction is to build human readable emerging topic summarization for organization users.

7. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (60973105, 90718017, 61170189, and 61202239), the Research Fund for the Doctoral Program of Higher Education (20111102130003), the Fund of the State Key Laboratory of Software Development Environment (SKLSDE-2013ZX-19), and the Innovation Foundation of Beihang University for Ph.D. Graduates (YWF-13-T-YJSY-024). This research was also supported by the Singapore National Research Foundation under its International Research Center @ Singapore Funding Initiative and administered by the IDM Programme Office.

8. REFERENCES

[1] C. G. Akcora, M. A. Bayir, M. Demirbas, and H. Ferhatosmanoglu. Identifying breakpoints in public opinion. In *Proceedings of the 1st Workshp on Social Media Analytics*, 2010.

[2] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st annual International ACM SIGIR Conference on Research and development in information retrieval*, 1998.

[3] H. Becker, M. Naaman, and L. Gravano. Learning similarity metrics for event identification in social media. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, 2010.

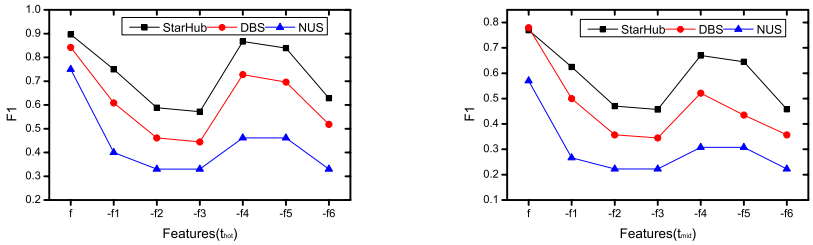
[4] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, 2006.

[5] D. M. Blei, A. Y. NG, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, pages 993–1022, 2003.

[6] M. Cataldi, L. D. Caro, and C. Schifanella. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the 10th International Workshop on Multimedia Data Mining*, 2010.

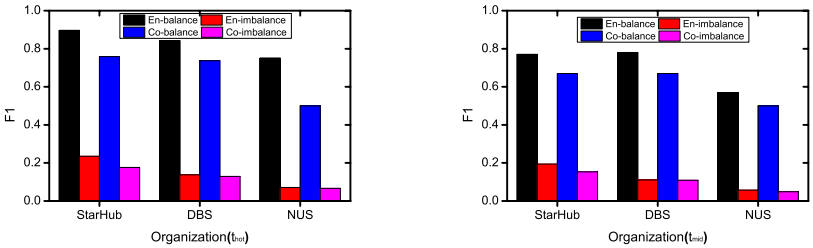
[7] Y. Chen, Z. Li, L. Nie, X. Hu, X. Wang, T. S. Chua, and X. Zhang. A semi-supervised bayesian network model for microblog topic classification. In *Proceedings of the 24th International Conference on Computational Linguistics*, 2012.

[8] Q. He, K. Chang, and E.-P. Lim. Analyzing feature trajectories for event detection. In *Proceedings of the*



(a) $T_L = t_{hot}$ (b) $T_L = t_{mid}$

Figure 7: The impact of emerging features on F_1 measure



(a) $T_L = t_{hot}$ (b) $T_L = t_{mid}$

Figure 8: The impact of imbalance data on F_1 measure

30th annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2007.

[9] L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the 1st workshop on Social Media Analytics*, 2010.

[10] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the 16th International Conference on Machine Learning*, 1999.

[11] S. P. Kasiviswanathan, P. Melville, A. Banerjee, and V. Sindhwani. Emerging topic detection using dictionary learning. In *Proceedings of the 20th ACM International Conference on Information and knowledge management*, 2011.

[12] A. Kotov, C. Zhai, and R. Sproat. Mining named entities with temporally correlated bursts from multilingual web news streams. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 2011.

[13] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, pages 249–268, 2007.

[14] H. O. Lancaster and E. Seneta. Chi-square distribution. *Encyclopedia of Biostatistics*, 2005.

[15] M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, 2010.

[16] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. In *Machine Learning - Special issue on information retrieval*, pages 103–134, 2000.

[17] A.-M. Popescu and M. Pennacchiotti. Detecting controverisal events from twitter. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 2010.

[18] K. Randinsky, S. Davidovich, and S. Markovitch. Learning causality for news events prediction. In *Proceedings of the 21st International Conference on World Wide Web*, 2012.

[19] A. Saha and V. Sindhwani. Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*, 2012.

[20] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web*, 2010.

[21] H. Sayyadi, M. Hurst, and A. Maykov. Event detection and tracking in social streams. In *Proceedings of the 3rd International AAAI Conference on Weblogs and Social Media*, 2009.

[22] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the 4th International AAAI Conference on Weblogs and Social Media*, 2011.

[23] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

[24] Y. Wang, E. Agichtein, and M. Benzi. Tm-lda: efficient online modeling of latent topic transitions in social media. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

[25] Y. Yang, T. Pierce, and J. Carbonell. A study on retrospective and online event detection. In *Proceedings of the 21st annual International ACM SIGIR Conference on Research and development in information retrieval*, 1998.