Expert Systems With Applications 114 (2018) 516–523

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Social event detection with retweeting behavior correlation

Xi Chen[a], Xiangmin Zhou[a,*], Timos Sellis[b], Xue Li[c]

[a] School of Science, RMIT University, Melbourne, VIC 3000, Australia
[b] School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC 3122, Australia
[c] School of Information Technology and Electrical Engineering, University of Queensland, Brisbane, QLD 4072, Australia

ARTICLE INFO

Article history:
Received 5 March 2018
Revised 11 August 2018
Accepted 12 August 2018
Available online 16 August 2018

Keywords:
Social media
Event detection
Retweeting behavior

ABSTRACT

Event detection over microblogs has attracted great research interest due to its wide application in crisis management and decision making etc. In natural disasters, complex events are reported in real time on social media sites, but these reports are invisible to crisis coordinators. Detecting these crisis events helps watchers to make right decisions rapidly, reducing injuries, deaths and economic loss. In sporting activities, detecting events helps audiences make better and more timely game viewing plans. However, existing event detection techniques are not effective at handling complex social events that evolve over time. In this paper, we propose an event detection method that takes advantage of retweeting behavior for handling the events evolution. Specifically, we first propose a topic model called RL-LDA to capture the social media information over hashtag, location, textual and retweeting behavior. Using RL-LDA, a complex event can be well handled by exploring the correlation between retweeting behavior and the event. Then to maintain the RL-LDA in a dynamic environment, we propose a dynamic update algorithm, which incrementally updates events over real time streams. Experiments over real-world datasets show that RL-LDA detects the temporal evolution of complex events effectively and efficiently.

1. Introduction

Microblogging services provide platforms for users to share their daily lives and report the events occurring around them in real time. Detecting social events is important in real applications. For example, monitoring crisis events like bushfires over social streams could help security officers predict the impact of disasters and supply the best service for the public during natural disasters. Social events can be complex and evolve over time. For instance, when the World Cup was hosted in Brazil in 2014, a large number of messages were posted to microblogs. The discussions over the microblogs evolved along with the game schedule. Detecting these evolving events helped users make the right decisions and adjust their plans in time. In practice, due to the high complexity of evolving events and the huge volume of social media, a satisfying quality and speed of detection has not been achieved yet. Consequently, how to effectively and efficiently detect such complex evolving events has become an important research problem.

We study the problem of complex event monitoring over social streams. A complex event is defined as a set of real-world social media messages happening over a time and location range but evolving over consecutive periods. Given a social media stream $\mathcal{M}$, a topic number K, we aim to continuously identify a set of complex social events $<E_i>$, each of which consists of messages on the same topic. In practice, it is vital to note that social media may involve highly complex and uncertain textual content and contextual information. Apart from the general characteristics of social media data, social streaming has the special requirement of one passing and real time response. In this paper, we focus on the problem of effective and efficient complex event detection over high speed social media streams.

Techniques have been proposed for event detection over microblogs (Avvenuti, Cresci, Marchetti, Meletti, & Tesconi, 2014; Bian, Yang, Zhang, & Chua, 2015; Cai, Yang, Li, & Huang, 2015; Ritter, Etzioni, & Clark, 2012; Yan, Guo, Lan, Xu, & Cheng, 2015; Zhao, Mitra, & Chen, 2007; Zhou & Chen, 2014). Existing detection methods focus on first story discovery (Petrović, Osborne, & Lavrenko, 2010), crisis management (Pohl, Bouchachia, & Hellwagner, 2012; Sakaki, Okazaki, & Matsuo, 2010; Zhou & Chen, 2014), and bursty events detection (Xie, Zhu, Ma, Xie, & Lin, 2014; Yao, Cui, Huang, & Zhou, 2010; Yin, Cui, Lu, Huang, & Yao, 2013). However, these methods only focus on event extraction and ignore event evolution over time. Though Abdelhaq, Sengstock, and Gertz (2013) considered the temporal evolution of events, the evolution was limited to the current time period, and the relationship between the time windows could not be constructed. As a

* Corresponding author.
E-mail addresses: xi.chen4@rmit.edu.au (X. Chen), xiangmin.zhou@rmit.edu.au (X. Zhou), tsellis@swin.edu.au (T. Sellis), xueli@itee.uq.edu.au (X. Li).

https://doi.org/10.1016/j.eswa.2018.08.022

result, this approach produced lower quality results for detecting evolving events. Hashtag-based event discovery (Xing, Wang, Liu, Huang, & Ma, 2016) identified the relationships among hashtags. However, it ignored event development over time. Although event discovery has been studied in various domains (Al Sumait, Barbará, & Domeniconi, 2008; Bian et al., 2015; Cai et al., 2015), there are still challenges due to the particular characteristics of social media. First, raw social data contain a large amount of noise, while useful media information is extremely sparse. Much emotional and personal information unrelated to any event fills the 140 character messages, which frequently contain very limited factual descriptions. Second, social media contain rich contexts, such as location, time and retweet behavior etc., which are valuable for enhancing the quality of event detection but hard to capture effectively. In addition, a large volume of social media flows over microblogs at high speed, which requires real time processing. Considering the media characteristics, to effectively and efficiently detect complex social events, we need to address two challenges. First, we need to construct a robust model that will capture the content and contexts. This is vital because content and contexts describe different aspects of a complex event. Improperly describing them will downgrade the quality of event detection. Then we need to design a robust model maintenance technique over streams. As such, the data model would be able to reflect the social updates of media data in recent time periods.

In this paper, we propose a retweeting behavior-based approach for finding temporally evolving social events. The proposed approach can well capture the intelligent behaviors of users and provide support to them in their decision making, which is significant in the expert and intelligent system field. Just as in general intelligent systems, we study intelligent user behaviors and their impacts on human society. By exploiting the topic modeling techniques in the artificial intelligence field, our approach can perform more accurate and effective operations for solving the related problem of automatic complex social event detection without human expertise. Meanwhile, it makes applications that can sense the environment, perceive relevant information on complex events and learn how to act in critical situations. Specifically, a retweeting behavior-based topic model (RL-LDA) is first constructed over the hashtag, content, location and retweeting behavior of social media. Then we propose a dynamic parameter update strategy to maintain the RL-LDA model under the social updates over streams. Finally, we conduct extensive experiments over real tweet streams to evaluate the performance of our proposed complex event detection approach. Our contributions are listed as follows:

- We propose a retweeting behavior-based topic model (RL-LDA) over hashtag, location, textual content and retweeting behavior, where each location is described as a novel retweeting behavior-based graph. Using RL-LDA, the evolution of an event can be well captured.
- An incremental computation-based update algorithm is proposed to dynamically maintain the RL-LDA model over streams, which well reflects the social updates in the recent time window.
- We have conducted extensive experiments over two real datasets. The test results prove the high effectiveness and efficiency of our proposed approach.

The rest of this paper is organized as follows. First, we briefly survey existing works on social event detection. Then we present our retweeting behavior-based event detection approach, followed by the experimental evaluation of our method. Finally, we conclude the whole paper.

2. Related work

Approaches have been proposed for detecting social events. Existing methods can be categorized into three types: feature-based, topic model-based and social-behavior-based.

Feature-based approaches detect the abnormal feature trends to identify the occurrence of social event in real applications. Commonly used features include statisticss features such as the term frequency (Chen, Amiri, Li, & Chua, 2013) and word co-occurrence (Yin et al., 2013), and context features like location and hashtag etc (Budak, Georgiou, Agrawal, & El Abbadi, 2013; Sakaki et al., 2010; Zhang et al., 2016). Statistics features have been extensively utilized to monitor the potential outbreak of social events. Chen et al. (2013) utilized the term frequency to detect pre-emergency events before the outbreak of an emergency. Yin et al. (2013) considered word co-occurrence in a time period as a Gaussion distribution and calculated its bursty degree by comparing the distribution in the current time slot and in recent historical periods. Context features are more likely used to dig for deep information about social events. Sakaki et al. (2010) considered users as social sensors of events when an earthquake occurred. By gathering the geotagged tweets on the earthquake, the location where earthquake happened could be obtained. GEOBURST (Zhang et al., 2016) detected local events over geotagged tweet streams by ranking the centroid of clusters formed by maximum weighted tweets. Although feature-based approaches perform well in the prenotice of outbreak events and in digging for information, they are not suitable for the detection of complex events that develop gradually and can not be generalized by any single type of information.

Topic-model-based methods detect events by adding layers. They have the extreme capacity of topic discovery due to their robustness over data ambiguity. Topic models are extended to unstructured data and multiple types of features. MGe-LDA (Xing et al., 2016) utilized a hashtag pair occurrence-based graph for detecting social event clusters. The clustering process is accelerated by loosing the sampling of topic assignment. Each word in a tweet is considered as a bridge connecting the hashtag and topic assignment. STM-TwitterLDA (Cai et al., 2015) collected the tweets posted in certain locations to detect local events using two types of dictionaries, specific and general, over words and images respectively. GeoFolk (Sizov, 2010) added two layers, longitude topic distribution and latitude topic distribution, into LDA to generate the location of the topic. TOT (Wang & Mc Callum, 2006) added time into LDA to make it suitable for continuous event detection. It connects event occurrences over time. LTT (Zhou & Chen, 2014) jointly modeled text content, time, longitude and latitude based on LDA to locate the sphere of disasters over streams. However, existing topic-model-based approaches lack the capacity to detect complex events with temporal evolution.

Social behavior-based methods detect the events by digging into the relationship between user behavior and events. User behavior plays a crucial role in event broadcasting. Existing methods discover user behavior over topics, and explore user interests or relationships etc. Wan, Milios, Kalyaniwalla, and Janssen (2009) detected social events based on the email links between users and their neighbors. Cluster deviations were detected to discover event occurrences. Qiu, Zhu, and Jiang (2013) considered four types of user behaviors (post, retweet, reply and mention) over tweets to discover the behavior distribution over topics. The results showed that users have different interests within topics. Achananuparp, Lim, Jiang, and Hoang (2012) weighted each tweet based on multiple features including retweet times and detected the bursty events based on the abnormally weighted tweets. Though these works find the relationships between users and topic interests, the relationship between retweeting behavior and events

Table 1
Comparison of existing approaches.

Method	Text	Time	Location	User behavior	Hashtag
Chen et al. (2013)	✓	✓		✓	
Doulamis, Doulamis, Kokkinos, and Varvarigos (2016)	✓	✓		✓	
Sizov (2010)	✓	✓	✓		
Unankard, Li, and Sharaf (2015)	✓	✓	✓		✓
Wang and Mc Callum (2006)	✓	✓			
Wan et al. (2009)	✓	✓			
Xing et al. (2016)	✓				✓
Yin et al. (2013)	✓	✓			
Zhang et al. (2016)	✓	✓	✓		
Zhou and Chen (2014)	✓	✓	✓		

Table 2
Notations and descriptions.

Note	Descriptions	Note	Descriptions
T	The number of time slots	k	A topic
V	The vocabulary size	h	A hashtag
K	Total number of topics	y_l, y_h	A switch
L	Total number of locations	ϕ	Topic-word distribution
H	Total number of hashtags	θ	Location-topic distribution
D	Total number of messages	θ'	Hashtag-topic distribution
N	Number of words in a message	ψ_l, ψ_h	Bernoulli distribution
G_H	Hashtag graph	γ_h	Proportion of hashtag h
G_L	Location graph	γ_l	Proportion of location l
l	A location	α, β, γ	Dirichlet priors
w	A word	τ_h, τ_l	Dirichlet priors
d	A message	ϵ_h, ϵ_l	Dirichlet priors

has not been considered. We summarize the existing approaches in Table 1 in terms of the information they captured. Note that none of these approaches can capture evolving events.

3. Retweeting behavior-based complex social event detection

In this section, we first present our retweeting behavior-based topic model (RL-LDA) for complex event detection. Then we propose an incremental-based update algorithm for dynamically maintaining our topic model over the streaming environment. Notations and definitions in RL-LDA are shown in Table 2.

3.1. Retweeting behavior-based topic model (RL-LDA)

Recall that social information is extremely noisy and sparse, which requires a model robust to these media characteristics. Given a corpus of social data, various topic models can be used for handling data uncertainty and topic discovery (Cai et al., 2015; Sizov, 2010; Wang & Mc Callum, 2006; Xing et al., 2016; Zhou & Chen, 2014). Among them, Latent Dirichlet Allocation(LDA) (Blei, Ng, & Jordan, 2003) variants have shown superiority in discovering unknown document patterns. However, in our application, an event cares about not only the topic, time and location of a specific real-world occurrence, but also its evolution over a time period. Thus, the social data model should be able to capture this evolution, which can not be obtained using existing LDA variants. Fortunately, the retweeting behavior of users provides useful clues on the social information flow over tweets, which reflects the evolution of events. Thus, we construct our topic model over textual content, time, location, hashtag and retweeting behavior. To avoid the effect of data sparsity, we ignore very short messages with personal and emotional related information, while only constructing our model over the social messages with location, hashtag and retweeting behavior.

We propose a retweeting behavior-based topic model to identify the complex social events with temporal evolution. Unlike the MGe-LDA (Xing et al., 2016) which considers the static hashtag context, our RL-LDA embeds user retweeting behavior that reflects the evolutionary change trend of real-world occurrence. Meanwhile, our RL-LDA model adopts an incremental computation-based maintenance strategy to handle the social updates over streams. Given a social corpus, we describe a tweet d as a combination of a word set $w_d = \left\{w_{d_1}, \cdots, w_{d_N}\right\}$, a location set $l_d = \left\{< la_{d_1}, lo_{d_1} >, \cdots, < la_{d_W}, lo_{d_W} >\right\}$ and a hashtag set $h_d = \left\{h_{d_1}, \cdots, h_{d_M}\right\}$. We extract from each tweet five types of features: content, time, hashtag, location and retweeting behavior. The content feature is described as a set of textual tokens extracted from a tweet and preprocessed by stemming, removing the stop words and emotional symbols. A time feature is described as the time of posting a tweet. A hashtag feature is described as a token starting with # in a tweet. A location is taken from the user profile of the original posting or that of retweeting and described as its latitude/longitude pair $< la_{d_i}, lo_{d_i} >$. The retweeting behavior feature is taken from retweets, and described as a location pair $< tw_p$, $tw_r >$, where tw_p is the location of the original posting and tw_r that of the retweeting. These features are considered as variables in document generation. The generative process of RL-LDA for a document is given below.

1. For each topic k: draw a word distribution $\phi \sim Dir(\beta)$;
2. For each location l: draw a topic distribution $\theta \sim Dir(\eta)$;
3. For each hashtag h: draw a topic distribution $\theta' \sim Dir(\alpha)$;
4. For each tweet $d = 1, \cdots D$, for each word w_{d_n}, $n = 1, \cdots N$
   (a) Draw a hashtag $s_{d_n} \sim P(h|z_h)$
      (i) Draw a switch y_h from ψ; if $y_h = 1$, sample s_{d_n} from g_h; if $y_h = 0$, $s_{d_n} = s_{d_n}$;
   (b) Draw a location $o_{d_n} \sim P(l|z_l)$
      (i) Draw a switch y_l from ψ; if $y_l = 1$, sample o_{d_n} from g_l; if $y_l = 0$, $o_{d_n} = o_{d_n}$;
   (c) Draw a topic $z_{d_n} \sim \theta'_{s_{d_n}}, \theta_{o_{d_n}}$

Fig. 1 shows the graphical model of RL-LDA. RL-LDA contains three levels, corpus level, document level and word level. Unlike MGe-LDA, we add a location layer based on the retweeting behavior

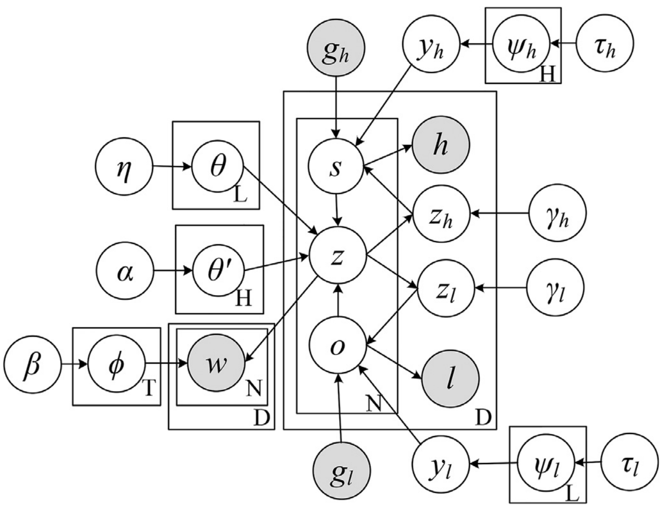

Fig. 1. An RL-LDA model.

to capture its impact on event evolution. Each hashtag or location is represented by a multinomial distribution over topic and each topic is described as a multinomial distribution of words. Thus, the generation of a tweet includes three parts, word, hashtag and location generations respectively. Given a tweet d, let $z_d = \{z_{d_1}, \cdots, z_{d_N}\}$, $s_d = \{s_{d_1}, \cdots, s_{d_N}\}$ and $o_d = \{o_{d_1}, \cdots, o_{d_N}\}$ be its topic, hashtag and location assignments, respectively. For each word w_{d_n} in d, we first choose a hashtag s_{d_n} based on the probability of selecting a hashtag h from the corpus under the condition z_h, $P(h|z_h)$, where $P(h|z_h) \propto P(h) \cdot P(z_h|h)$. Here, z_h is the topic assignments for each hashtag in the corpus, which is connected to the hashtag by the words in the same tweet and indirectly reflects the topic distributions over hashtags. $P(h)$ is the probability of hashtag h appearing in the corpus. The topic assignments of each word in a tweet can be used for the topic assignments of hashtags in the tweet. Then, we choose a location o_{d_n} for each word w_{d_n} in tweet d based on the probability of selecting a location l from the corpus under the condition z_l, $P(l|z_l)$, where $P(l|z_l) \propto P(l) \cdot P(z_l|l)$. Here, z_l denotes the topic assignments for location l, which reflects the effect of topic distribution over the location via words. After that, we choose topic z_{d_n} according to $\theta'_{s_{d_n}}$ and $\theta_{o_{d_n}}$, which are the topic distribution of the hashtag s_{d_n} and that of the location o_{d_n} respectively.

Since there are a very limited number of hashtags and locations in each tweet, the options for the selection of hashtags and locations in it can be very few. Thus, we expand the selection of hashtags and locations to the whole corpus from a single tweet to loose the selection ranges of hashtag s_{d_n} and location o_{d_n}. To achieve this, we construct a hashtag co-occurrence-based graph following the idea in MGe-LDA (Xing et al., 2016), and propose a novel retweeting behavior-based graph for the locations in the whole corpus. Given a set of hashtags, we consider each hashtag as a node, and two hashtags are connected with an edge if they appear in the same tweet. Here, we care only if two nodes are connected, and we ignore the weight of each edge. The subgraph to a hashtag is the nodes that are directly connected to it, based on which the hashtag assignment for a word is conducted. Given a set of locations, we consider each location as a node, and the locations with retweeting behaviors are connected. The subgraph to a location consists of all the nodes directly connecting to it. As such, the location assignment of a word can be done over its location subset. To decide if a hashtag or location is selected from its subgraph or the original set to its tweet, we set two switches, y_h and y_l, which are determined by the values of their Bernoulli distributions ψ_h and ψ_l respectively. If the switch of a hashtag is positive, we conduct the hashtag assignment to a word over its subgraph. Otherwise, the assignment operation is done over the hashtag set of its original tweet. Similarly, the location assignment is conducted over its subgraph if its switch is positive otherwise, over its original location set in a tweet. Using the subgraph-based hashtag assignment and location assignment, the sparsity problem of social media data over these attributes can be overcome.

After the structure of our RL-LDA model is decided, we need a parameter estimation for this model. We use Gibbs sampling to sample hidden variable assignment of θ, θ' and ϕ with η, α, β as prior parameters respectively. According to the processing of RL-LDA, we firstly sampled s_{d_n} from h_{d_n} as follows:

$$P\left(s_{d_n=h}|z_h\right) \propto \frac{n_h^k + \alpha}{\sum_{i=1}^{K} n_h^k + K\alpha} \cdot \frac{n_h}{N_H} \tag{1}$$

where n_h^k is the times that topic k assigned to hashtag h, n_h the times that hashtag h appears in hashtag corpus, and N_H the times of all hashtags appearing in the corpus. The location o_{d_n} is sampled based on the equation as follows:

$$P\left(o_{d_i=l}|z_l\right) \propto \frac{n_l^k + \eta}{\sum_{i=1}^{K} n_l^k + K\eta} \cdot \frac{n_l}{N_L} \tag{2}$$

where n_l^k is the times that topic k assigned to location l, n_l the times that location l appears in whole corpus, and N_L the times of all locations appearing in the corpus. With respect to topic assignment z_{d_n}, it is sampled based on s_{d_n} and z_{d_n}. The calculation is as follows:

$$P\left(z_{d_n=k}|h, l\right) \propto \frac{n_k^w + \beta}{\sum_{i=1}^{V} n_k^w + V\beta} \cdot \frac{n_h^k + \alpha}{\sum_{i=1}^{K} n_h^k + K\alpha} \cdot \frac{n_l^k + \eta}{\sum_{i=1}^{K} n_l^k + K\eta} \tag{3}$$

where n_k^w is the times that word w assigned to topic k. Thus we could get the final result of θ', θ and ϕ as follows:

$$\theta' \propto \frac{n_h^k + \alpha}{\sum_{i=1}^{K} n_h^k + K\alpha}, \theta \propto \frac{n_l^k + \eta}{\sum_{i=1}^{K} n_l^k + K\eta}, \phi \propto \frac{n_k^w + \beta}{\sum_{i=1}^{V} n_k^w + V\beta} \tag{4}$$

3.2. Incremental update

Social media data flow over streams in huge volume at high speed due to user activities. Accordingly, the RL-LDA model constructed over the previous time window can not reflect the data information in the current time period, thus becoming ineffective for topic discovery. To solve this problem, we propose an incremental model maintenance that estimates the parameters of the RL-LDA by using their values for the previous time windows.

To maintain the RL-LDA, we need to estimate the parameters of hashtags, locations and words in the current time slot according to their appearance in the previous slots. Given a set of tweets in time slot t, we estimate α_h^t, η_l^t and β_w^t, the parameter of hashtag h, location l and word w in t, based on their update matrices H_h^{t-1}, L_l^{t-1} and V_w^{t-1} obtained from δ previous time slots respectively. We assume that the parameter estimation in the current time slot can only be affected by their normalization of counts in δ previous time slots. Then the columns of H_h^{t-1}, L_l^{t-1} and V_w^{t-1} are formed by h_j, l_j and w_j, where h_j, l_j and w_j are the normalization of counts of h, l and w in time j respectively, $j \in \{t-\delta-1, \cdots, t-1\}$. So far, three parameter update matrices H_h^{t-1}, L_l^{t-1} and V_w^{t-1} are built over hashtag, location and word in time slot t. We use a weighted δ-dimensional vector $<\omega_1, \omega_2, \cdots, \omega_\delta>$, where the sum of these weights is equal to 1, to reflect the impact of previous time slot sequence on the current time window. Then, the parameter of hashtag h, location l and word w in time slot t can be estimated as follows:

$$\alpha_h^t = H_h^{t-1} \omega^\delta \tag{5}$$

$$\eta_l^t = L_l^{t-1}\omega^\delta \tag{6}$$

$$\beta_w^t = V_w^{t-1}\omega^\delta \tag{7}$$

H_h^t, L_l^t and V_w^t are updated by adding their normalized counts of hashtag h, location l and word w in time window t and removing their values in time slot $t-\delta-1$. It is common that new elements appear in the corpus in the current time slot but do not exist in the previous time window. Thus, the normalized counts of an element in the previous time slots are initialized as 0 if it is a new incoming one. For the first time slot, the parameters of hashtags, locations and words are set as their default constants α, η and β respectively. With the incremental update method over RL-LDA, the evolution of complex events can be well captured under a dynamic environment.

To accelerate the speed of model maintenance, we adaptively decide whether the incremental update process will be conducted based on the difference of hashtag distributions between two neighboring time slots. Given a time slot t, we describe its hashtag distribution D_t by counting the frequency of each hashtag in its hashtag set. Given two neighboring time slots t and $t+1$, we measure the dissimilarity between their hashtag distributions using a Kullback–Leibler divergence-based distance as follows:

$$\mathcal{D}_{ht}(D_t, D_{t+1}) = \frac{1}{2}(\mathcal{D}_{KL}(D_t||D_{t+1}) + \mathcal{D}_{KL}(D_{t+1}, D_t)) \tag{8}$$

where

$$\mathcal{D}_{KL}(D_{t+1}, D_t) = \sum_i D_{t+1}(h_i) log\frac{D_{t+1}(h_i)}{D_t(h_i)} \tag{9}$$

Here h_i is the probability of hashtag i that appears in a time slot. If the dissimilarity between the hashtag distribution is smaller than a given threshold ε, the topic discussed is not changed much; thus, we believe the model for time slot $t+1$ is the same as that for time slot t. Otherwise, we trigger the incremental update maintenance process of RL-LDA. The optimal ε will be evaluated in Section 4.

3.3. Cost analysis

We estimate the CPU costs of training models using different approaches, including RL-LDA, incremental RL-LDA and MGe-LDA (Xing et al., 2016). In RL-LDA, each word in a tweet is attached to a hashtag, a location and a topic. For each word, a hashtag is selected from two hashtag sets based on its switch point. Likewise, a location is selected from its location set based on its location switch point. Here, we estimate the cost of training RL-LDA under the worst situation where the hashtag and location to each word are selected from their corresponding sets over the whole corpus. Let t_s be the cost of Gibbs sampling for one element, N be the number of words in a tweet, H, L, K be the number of hashtags, that of locations and that of topics in the corpus respectively, the CPU cost of training RL-LDA for each tweet is $N * t_s * (H + L + K)$.

Incremental RL-LDA calculates the distance between the hashtag distributions of continuous time intervals to notice the change of event in a consecutive time period. If an event doesn't change within two consecutive time intervals, the RL-LDA model doesn't need to be retrained. Let $t_{(RL-LDA)}$ be the training cost of RL-LDA in a time interval, T be the number of time intervals in the whole dataset, T_{E_m} be the number of time intervals that events do not change. The cost of incremental RL-LDA over the entire dataset is $t_{(RL-LDA)} * (T - T_{E_m})$.

MGe-LDA detects events by utilizing a hashtag-based mutually generative topic model. The CPU cost of training MGe-LDA for each tweet is $N * t_s * (H + K)$. Let $t_{MGe-LDA}$ be the training cost of MGe-LDA in each time interval. The cost of MGe-LDA in dealing with the entire dataset is $T * t_{MGe-LDA}$, where T is the number of time intervals in the whole dataset. Compared with MGe-LDA, RL-LDA needs extra cost to process the location of each tweet for training the model. Under the worst situation, the extra training cost of RL-LDA for dealing with a tweet is $N^*L^*t_s$ compared with MGe-LDA. Thus, for the training cost in a time interval, we have $t_{(RL-LDA)} > t_{MGe-LDA}$. However, for the entire dataset, incremental RL-LDA spends less time on retraining because RL-LDA is retrained only when the event changes over consecutive time intervals.

For a further cost comparison between the proposed approach and MGe-LDA, we conduct statistical analysis over a real-world dataset that contains two events, the World Cup 2014 and the Much Music Video Awards. The dataset contains 1,028,264 tweets, 152,073 hashtags and 22,411 locations. Suppose that the topic number is set to 25, then RL-LDA needs 12.8% more time than MGe-LDA to deal with the location for training the model. Let ε be set to 0.2 as in Section 4.3.2. 28% time intervals do not involve event changes for the given dataset, which does not need the retraining of RL-LDA. Thus, we conclude that the time cost of incremental RL-LDA and that of MGe-LDA are comparable over the whole dataset. Meanwhile, the cost of original RL-LDA incurs the highest time cost for training the models over different time periods, while gaining better effectiveness performance as proved in Section 4.

4. Experiment evaluation

This section demonstrates the effectiveness and efficiency of our proposed approach to detecting events with temporal evolutions.

4.1. Experimental setup

In order to conduct experimental evaluation, we exploit the English tweets posted during 8–21 June 2014, a total of 22 million tweets over 70 GB data, which are divided into two datasets DS_1 and DS_2. DS_1 includes all the tweets in 8–14 June, in which the broadcast event iHeartRadio Much Music Video Awards (MMVAs) was discussed. DS_2 contains all those posted in 15–21 June, in which another broadcast event, the 2014 Brazil World Cup (WC2014), was extensively discussed. To meet the requirements of the RL-LDA model, tweets need contain hashtag, retweeting behavior, location and text. Intuitively, some frequently appearing hashtags, such as #retweet, do not contain any relevant information with any topics, thus are considered as stop hashtags and removed. We consider the locations, each of which appeared at least once in tweets with retweeting behavior. Texts are stemmed and stop words are removed. The final filtered dataset contains 1,028,264 tweets with 152,073 hashtags and 22,411 locations. We manually built the ground truth of these two events. Finally, 87,225 tweets, 712 hashtags and 5415 locations are labeled as WC2014 and 54,060 tweets, 387 hashtags and 2443 locations are labeled as MMVAs.

4.2. Evaluation methodology

We evaluate the effectiveness of our RL-LDA based complex event detection over three metrics, F1 score, probability of missed detection and probability of false alarm over DS_1 and DS_2. F1 score is a commonly used method to evaluate the quality of clusters over recall and precision simultaneously, which is computed by:

$$F1 = \frac{2 * precision * recall}{precision + recall} \tag{10}$$

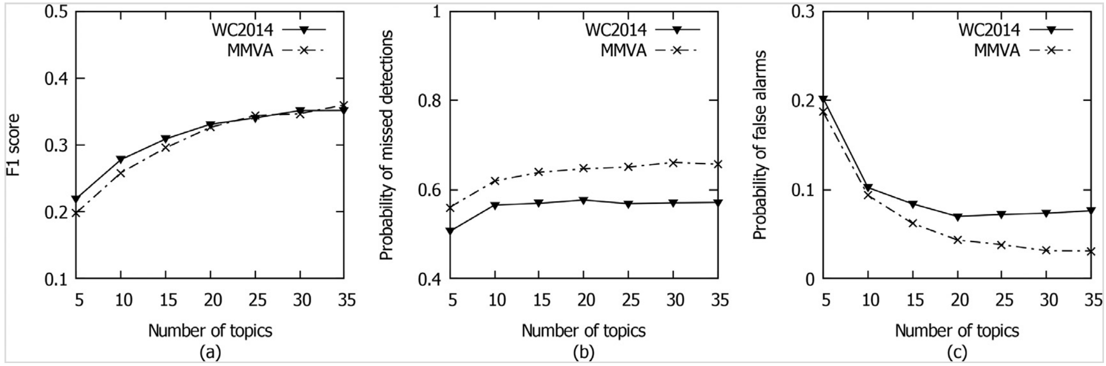

Fig. 2. Effect of K.

Probability of missed tweet detection (P_{Miss}) and probability of false tweet alarm (P_{False}) are metrics used to evaluate the effectiveness of event detection (Cai et al., 2015; Zhou & Chen, 2014). These metrics are defined as:

$$P_{Miss} = \frac{number\ of\ missed\ detections}{number\ of\ targets} \tag{11}$$

and

$$P_{False} = \frac{number\ of\ false\ alarms}{number\ of\ nontargets} \tag{12}$$

A target is defined as a ground truth tweet that should be assigned to an event, while a non-target is the opposite. P_{Miss} and P_{False} evaluate the ratio of the missed true targets and that of falsely assigned non-targets to all targets in ground truth respectively. A high quality event detection method should have a large F1, small P_{Miss} and small P_{False}.

Our effectiveness evaluation includes three parts: (a) the parameter turning of RL-LDA; (b) the effect of threshold for the incremental updated RL-LDA; and (c) the comparison with the state-of-the-art topic-model-based detection methods. We evaluate the efficiency of our proposed approach in terms of the overall time cost of event detection over tweet streams. The whole dataset of 70 GB data streams of 22 million tweets is used for the efficiency test. Tests are conducted on Intel Core i7-2600 @ 3.40GHz, RAM 8.00GB with 64-bit system.

4.3. Effectiveness evaluation

First, we evaluate the effect of topic number K over RL-LDA and update threshold ε to find the optimal default values. Then, we compare RL-LDA and incremental updated RL-LDA with the state-of-the-art model MGe-LDA and LDA.

4.3.1. Effect of topic number

We test the effectiveness of RL-LDA by varying the topic number K from 5 to 35 to find its optimal value. Figs. 2 (a)–(c) show the effectiveness of RL-LDA in terms of three metrics. Clearly, with the increase of K, the F1 and P_{Miss} values of RL-LDA model increase gradually, while its P_{False} value drops quickly from 5 to 25. The reason is that tweets related to different topics are more likely to be assigned as the same topic when K is small. With the increase of K, the topic assignments of tweets become more precise. Meanwhile, we can observe that the effectiveness of RL-LDA keeps steady in terms of F1, P_{Miss} and P_{False} after K=25. This is because the discrimination power of topics reaches a satisfactory level, and there is less improvement space after $K = 25$. Considering the balance between the effectiveness and efficiency of our event detection, we set the default value of K as 25.

Table 3
Efficiency comparison.

Methods	RL-LDA	RL-LDA(updated)	MGe-LDA	LDA
Time costs(s)	2986	1813	2015	604

4.3.2. Effect of ε

We test the effectiveness of updated RL-LDA with the hashtag distribution threshold ε change from 0.1 to 0.35. Figs. 3 (a)–(c) show the effectiveness of updated RL-LDA at each ε in terms of three metrics. As we can see, with the increase of ε from 0.1 to 0.2, the effectiveness of updated RL-LDA degrade slightly. With the further increase of ε, the performance of our model drops significantly. Considering the balance between effectiveness and efficiency, we select 0.2 as the default value of ε.

4.3.3. Effectiveness comparison

We conduct experiments to evaluate the effectiveness of three topic model-based event detection approaches, RL-LDA, updated RL-LDA, MGe-LDA and LDA. Here, K and ε are set to their default values. The comparison results are shown in Fig. 4. Clearly, RL-LDA outperforms MGe-LDA and LDA in terms of F1 and P_{Miss}, whereas they are not effective enough on P_{False}. The reason is that compared with MGe-LDA and LDA, RL-LDA considers the hashtag co-occurrence and the retweeting behavior correlation as well, which effectively helps group messages and reduces missed detections. Meanwhile, due to the large scale of related locations collected based on retweeting behaviors, some irrelevant messages are grouped into clusters as well. Compared with MMVAs, WC2014 has a wide sphere over locations. Thus, RL-LDA performs better on MMVAs in terms of P_{False}. Overall, RL-LDA outperforms MGe-LDA and LDA considering a better balance between P_{Miss} and P_{False}, which is indicated as its better F1 values over all investigated events.

Compared with the original RL-LDA, incremental updated RL-LDA has an effectiveness drop in terms of F1 and P_{Miss}, whereas it has a better performance on P_{False} over WC2014 and MMVAs. The reason is that the grouping of updated RL-LDA not only contains location, hashtag, retweeting behavior in current time slot, but also contains the impact in the previous time slots as well. Overall, the updated RL-LDA can well capture the evolution of a complex event.

4.4. Efficiency comparison

We evaluate the efficiency of RL-LDA, incremental updated RL-LDA, MGe-LDA and LDA by setting the parameters to their default values. The overall time costs of detection using different models are reported in Table 3. RL-LDA costs more time compared with MGe-LDA due to its extra processing on location related

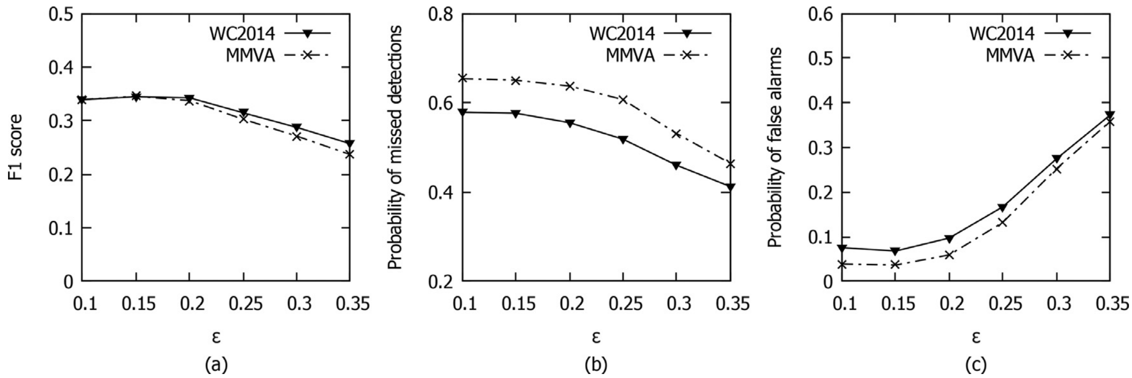

Fig. 3. Effect of ε.

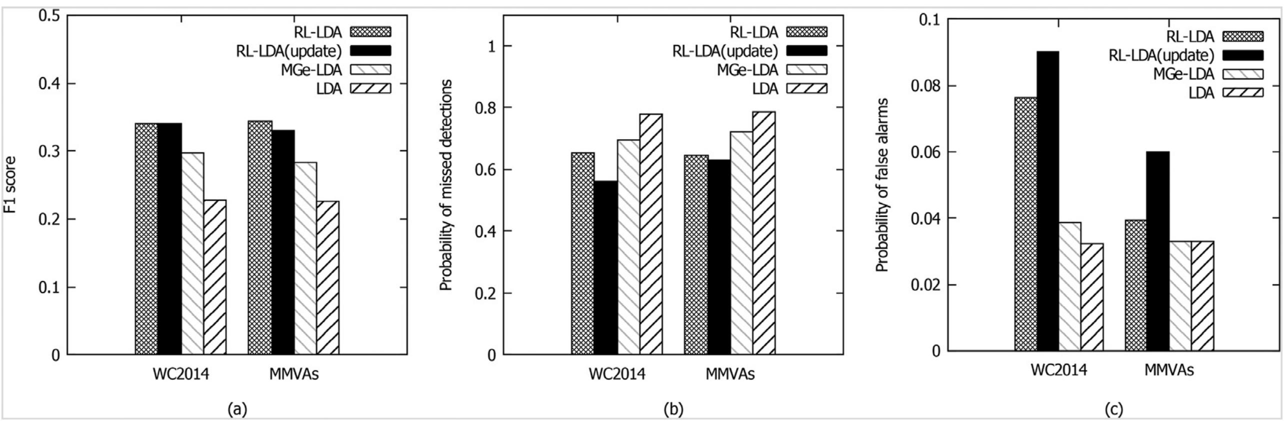

Fig. 4. Effectiveness comparison.

calculation. Incremental updated RL-LDA outperforms MGe-LDA and RL-LDA in terms of efficiency because it adopts incremental update maintenance and adaptively decides the time point for conducting maintenance, which removes the redundant model training operations. Though LDA costs the least time for detection, it has extremely low effectiveness. Considering both effectiveness and efficiency, our proposed models have much better efficacy for complex event detection.

The experimental results show that the effectiveness of the RL-LDA model keeps steady after the number of topics is increased to an optimal value. Thus, we only need to detect a limited number of events to trade off the effectiveness and source consumption of the system. Meanwhile, the event changes between consecutive time slots kept within a certain range. For the experimental results on efficiency evaluation, the results prove that our approach improves the response time of complex event detection significantly. Our proposed approach has provided insights into the characteristics of event occurrence and event evolution. The experimental results indicate that event evolution can be tracked by detecting event changes over time.

5. Conclusion

In this paper, we study the problem of detecting complex evolving events over social media. We first propose a retweeting behavior based topic model, RL-LDA, over text, hashtag, location, and retweeting behavior. Both hashtag co-occurrence and retweeting behavior are exploited to form two types of graphs that overcome the issue of tweet sparsity. Then we propose an incremental based RL-LDA update method over hashtags, locations and words to capture the evolution of events by considering the impacts of previous time slots over the current one. Finally, we conduct extensive tests to evaluate the effectiveness and efficiency of our approach. The experimental results have proved the high performance of our approach to detecting complex events with temporal evolutions.

The proposed RL-LDA model extends the MGe-LDA model by embedding the retweeting behavior of social users to accommodate the temporal evolution of complex events. By connecting the locations with the retweeting behaviors of social users, RL-LDA achieves better performance for complex event detection compared with existing approaches. This indicates that hidden relationships between different attributes in social media contain critical information for event detection. Moreover, the temporal event evolution is captured by measuring the event changes between consecutive time intervals. It inspires us to think that event evolution can be captured by monitoring the highly correlated event attributes. We mathematically show that the efficiency of RL-LDA depends on the characteristics of datasets, and event characteristics decide the speed of capturing event evolution. For practical utility, the proposed approach is significant for game view planning and disaster management.

The RL-LDA model for complex event detection has two limitations. First, we have not considered the evolution of events over social dimensions. The user connection structures may change over time, which reflects the evolution of complex events. Thus, our future work is to further investigate the effect of user connection evolutions. Second, our model is constructed over a single processor, which may not be efficient enough for handling detection over big social streams. To address this issue, for the next step, we will design efficient RL-LDA based complex event detection over a distributed environment. In addition, we will investigate new solutions for predicting complex social events over future time periods, and summarize the complex events for easy interpretation of them to interested social users.

References

Abdelhaq, H., Sengstock, C., & Gertz, M. (2013). Eventweet: Online localized event detection from twitter. *Proceedings of the VLDB Endowment, 6*(12), 1326–1329.

Achananuparp, P., Lim, E. P., Jiang, J., & Hoang, T. A. (2012). Who is retweeting the tweeters? Modeling, originating, and promoting behaviors in the twitter network. *ACM Transactions on Management Information Systems (TMIS), 3*(3), 13.

Al Sumait, L., Barbará, D., & Domeniconi, C. (2008). On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Data mining, 2008. ICDM'08. eighth IEEE international conference on* (pp. 3–12). IEEE.

Avvenuti, M., Cresci, S., Marchetti, A., Meletti, C., & Tesconi, M. (2014). Ears (earthquake alert and report system): A real time decision support system for earthquake crisis management. In *Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1749–1758). ACM.

Bian, J., Yang, Y., Zhang, H., & Chua, T. S. (2015). Multimedia summarization for social events in microblog stream. *IEEE Transactions on Multimedia, 17*(2), 216–228.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research, 3*(January), 993–1022.

Budak, C., Georgiou, T., Agrawal, D., & El Abbadi, A. (2013). *Geoscope: Online detection of geo-correlated information trends in social networks* (7, pp. 229–240).

Cai, H., Yang, Y., Li, X., & Huang, Z. (2015). What are popular: Exploring twitter features for event detection, tracking and visualization. In *Proceedings of the 23rd ACM international conference on multimedia* (pp. 89–98). ACM.

Chen, Y., Amiri, H., Li, Z., & Chua, T. S. (2013). Emerging topic detection for organizations from microblogs. In *Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval* (pp. 43–52). ACM.

Doulamis, N. D., Doulamis, A. D., Kokkinos, P., & Varvarigos, E. M. (2016). Event detection in twitter microblogging. *IEEE Transactions on Cybernetics, 46*(12), 2810–2824.

Petrović, S., Osborne, M., & Lavrenko, V. (2010). Streaming first story detection with application to twitter. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics* (pp. 181–189). Association for Computational Linguistics.

Pohl, D., Bouchachia, A., & Hellwagner, H. (2012). Automatic sub-event detection in emergency management using social media. In *Proceedings of the 21st international conference on world wide web* (pp. 683–686). ACM.

Qiu, M., Zhu, F., & Jiang, J. (2013). It is not just what we say, but how we say them: Lda-based behavior-topic model. In *Proceedings of the 2013 SIAM international conference on data mining* (pp. 794–802). SIAM.

Ritter, A., Etzioni, O., Clark, S., et al. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1104–1112). ACM.

Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). Earthquake shakes twitter users: Real–time event detection by social sensors.. In *Proceedings of the 19th international conference on world wide web* (pp. 851–860). ACM.

Sizov, S. (2010). Geofolk: Latent spatial semantics in web 2.0 social media. In *Proceedings of the third ACM international conference on web search and data mining* (pp. 281–290). ACM.

Unankard, S., Li, X., & Sharaf, M. A. (2015). Emerging event detection in social networks with location sensitivity. *World Wide Web, 18*(5), 1393–1417.

Wan, X., Milios, E., Kalyaniwalla, N., & Janssen, J. (2009). Link-based event detection in email communication networks. In *Proceedings of the 2009 ACM symposium on applied computing* (pp. 1506–1510). ACM.

Wang, X., & Mc Callum, A. (2006). 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* (pp. 424–433). ACM.

Xie, R., Zhu, F., Ma, H., Xie, W., & Lin, C. (2014). Clear: A real-time online observatory for bursty and viral events. *Proceedings of the VLDB Endowment, 7*(13), 1637–1640.

Xing, C., Wang, Y., Liu, J., Huang, Y., & Ma, W. Y. (2016). Hashtag-based sub-event discovery using mutually generative lda in twitter. *Thirtieth AAAI conference on artificial intelligence*.

Yan, X., Guo, J., Lan, Y., Xu, J., & Cheng, X. (2015). A probabilistic model for bursty topic discovery in microblogs. In *AAAI* (pp. 353–359).

Yao, J., Cui, B., Huang, Y., & Zhou, Y. (2010). Detecting bursty events in collaborative tagging systems. In *Data engineering (ICDE), 2010 IEEE 26th international conference on* (pp. 780–783). IEEE

Yin, H., Cui, B., Lu, H., Huang, Y., & Yao, J. (2013). A unified model for stable and temporal topic detection from social media data. In *Data engineering (ICDE), 2013 IEEE 29th international conference on* (pp. 661–672). IEEE.

Zhang, C., Zhou, G., Yuan, Q., Zhuang, H., Zheng, Y., Kaplan, L., et al. (2016). Geoburst: Real-time local event detection in geo-tagged tweet streams. In *Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval* (pp. 513–522). ACM.

Zhao, Q., Mitra, P., & Chen, B. (2007). Temporal and information flow based event detection from social text streams. In *AAAI: vol. 7* (pp. 1501–1506).

Zhou, X., & Chen, L. (2014). Event detection over twitter social media streams. *The VLDB Journal, 23*(3), 381–400.