When to Make a Topic Popular Again? A Temporal Model for Topic Rehotting Prediction in Online Social Networks

Chaokun Wang, *Member, IEEE*, Xin Xin, and Jingwen Shang

***Abstract*—It is really popular to detect hot topics, which can benefit many tasks including topic recommendations, the guidance of public opinions, and so on. However, in some cases, people may want to know when to rehot a topic, i.e., make the topic popular again. In this paper, we address this issue by introducing a temporal user topic participation (UTP) model, which models users' behaviors of posting messages. The UTP model takes into account users' interests, friend-circles, and unexpected events in online social networks. Also, it considers the continuous temporal modeling of topics, since topics are changing continuously over time. Furthermore, a weighting scheme is proposed to smooth the fluctuations in topic rehotting prediction. Finally, experimental results conducted on real-world data sets demonstrate the effectiveness of our proposed models and topic rehotting prediction methods.**

***Index Terms*—EMG algorithm, probabilistic graphical model, social networks, topic re-hotting prediction.**

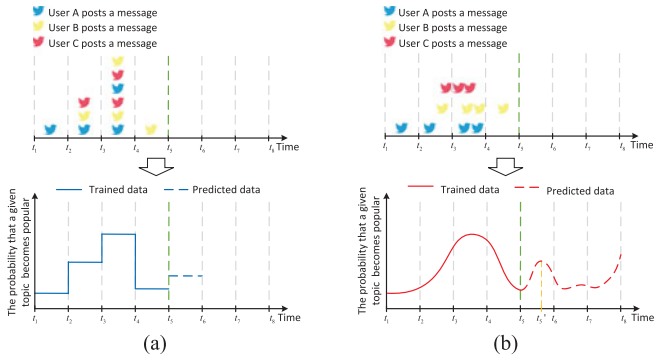


Fig. 1. An illustration of the two strategies for the topic re-hotting prediction problem (a) Discrete modeling strategy. (b) Continuous modeling strategy.

I. Motivation

With the rapid development of data storage, information processing, and networking transmission technologies, online social networks (OSNs) have been becoming indispensable in people's daily life. Everyone could freely post messages, share news, and participate in topic discussions in OSNs, e.g., Twitter (twitter.com) and Weibo (weibo.com). Along with that, many researchers have done lots of work for the convenience to analyze and use OSNs, such as topic detection [1], topic prediction [2], and topic transition [3].

However, the phenomena of topic decay and even disappearance are inevitable. It is reported that 23% of topics have two or more hot (a.k.a. active or popular) periods [4]. Clearly, in many situations, after observing that a hot topic is dwindling, it is very interesting but challenging to intelligently extrapolate when this topic may be *re-hot*, i.e., make the topic hot again at suitable time points. It is called the problem of *topic re-hotting prediction* in this study, and has a lot of practical applications.

Example 1: A top advertiser posts a commodity advertisement which becomes a hot topic in an OSN in a short time. As time goes by, the attention of consumers on the commodity is reduced and the hotness of the advertisement begins to wane. At this time, the advertiser may want to know the best time point to *re-hot* the topic and to keep the advertisement popular again in the OSN.

We argue topic re-hotting prediction is more difficult than topic detection. The methods of topic detection only justify whether or not a new topic is emerging, however the topic re-hotting prediction approaches should tell exact time points when a given topic will re-emerge.

Unfortunately, to the best of our knowledge, few studies considered when to re-hot topics so far. There are several big challenges to deal with this issue. Firstly, it is non-trivial to formalize the problem of topic re-hotting prediction and reasonably model the mechanism of topic participation. Secondly, it is very difficult to precisely obtain opportune time points for re-hotting a given topic. Last but not least, it is not easy to propose an effective topic re-hotting prediction approach.

This paper addresses the problem of topic re-hotting prediction. As shown in Fig. 1, we could consider the following two strategies to deal with the topic re-hotting prediction problem. (1) The discrete modeling strategy divides the whole time domain into contiguous non-overlapping time windows, and then uses the trained data (depicted as blue broken lines) to predict

Manuscript received May 12, 2016; revised October 31, 2016 and January 10, 2017; accepted January 24, 2017. Date of publication February 16, 2017; date of current version February 19, 2018. This work was supported in part by the National Natural Science Foundation of China under Grants 61373023 and 61170064 and in part by the China National Arts Fund (20164129). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yan Lindsay Sun. *(Corresponding author: Chaokun Wang.)*

The authors are with the School of Software, Tsinghua University, Beijing 100084, China (e-mail: chaokun@tsinghua.edu.cn; xin-x13@mails.thu.edu.cn; shangjw15@mails.thu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSIPN.2017.2670498

whether the topic will re-hot in the next time window (i.e., during the period from t_5 to t_6). Although this strategy is easily understandable, it cannot predict accurate time points for re-hotting a given topic. Furthermore, it is hard to describe the changing trends of topics in a fine-grained manner. (2) The continuous modeling strategy argues that topics are continuously changing in the time domain. Based on the trained data (depicted as red solid lines), it predicts accurate time points when the topic will re-hot, e.g., at the time point t'_5. Please note that this strategy could predict the re-hotting time points over a long period of time (depicted as red dotted lines) instead of just the next time window.

In this work, we focus on the second strategy. The main contributions of this paper can be summarized as follows.

1) We present and formalize the problem of topic re-hotting prediction (TRP) in OSNs at the first time. It facilitates a better understanding of the topic characteristics when the focusing topics are dwindling, as well as benefits many related issues, such as topic detection and topic tracing.
2) We propose a novel temporal model, i.e., User Topic Participation (UTP) model, for the TRP problem. UTP can effectively explain users' behaviors of participating in the topic discussions in OSNs. Also, we bring forward an improved EM algorithm called EMG to effectively infer the UTP model.
3) We design a method based on the UTP model to appropriately predict the re-hotting time points for given once-hot topics, i.e., the topics which had been hot before.
4) We evaluate the performance of our methods on three different real-world data sets collected from OSNs. Experimental results demonstrate the effectiveness of both the proposed UTP model and TRP method.

The remainder of this paper is organized as follows. Section II revisits the existing work related to this study. In Section III, we formally propose the problem of topic re-hotting prediction. In Section IV, the UTP model is presented. Afterwards, the inference algorithm and the TRP approach are given in Section V. We evaluate the performance of the re-hot prediction in Section VI and conclude this paper in Section VII.

II. Related Work

In this section, we survey the related research work, which covers different aspects: topic model, hot topic detection, event prediction, temporal behavior prediction, and EM algorithm.

Topic Model: The topic model is a kind of statistical models which are usually used to find abstract topics in a set of documents. Hofmann proposes the PLSA (Probabilistic Latent Semantic Analysis) model [5], which has a huge impact in the field of natural language and text processing. Moreover, in contrast to LSA (Latent Semantic Analysis) [6], the probabilistic variant of PLSA has a solid statistical foundation and defines a proper generative data model. Another topic model is LDA (Latent Dirichlet Allocation) [7] which is one of the most typical models. PAM (Pachinko Allocation Model) [8] is introduced as a model which uses a directed acyclic graph to describe the structure between documents and topics.

Hot Topic Detection: The hot topic detection is pretty popular in artificial intelligence and data mining area. In [9], a filter-refinement framework is proposed to discover hot topics corresponding to geographical dense regions. The authors analyze the cultures, scenes, and human behaviors from videos based on their spatio-temporal distributions. In [10], Wang *et al.* propose an algorithm to predict topic trends, which addresses the problem of short life circles of topics. Furthermore, in [11], a methodology is presented to detect the topic of epidemics based on Twitter. However, all these research work just concentrates on detecting popular topics, and they cannot be directly used to deal with the problem of TRP.

Event Detection: Event detection in social media has recently been studied by many researchers. Zhang *et al.* propose a new method to detect events and to predict their popularity simultaneously [12]. Specifically, they detect events from online microblogging stream by utilizing multiple types of information, i.e., term frequency and users' social relation. Meanwhile, the popularity of detected event is predicted through a proposed diffusion model which takes both the content and user information of the event into account. Stilo and Velardi present an algorithm named SAX* for event discovery [13]–[15]. In [13] and [15], they transform word temporal series into a string of symbols using Symbolic Aggregate approXimation (SAX). In [14], the authors propose a method for hashtag sense clustering based on temporal co-occurrence and similarity of the related time series.

Temporal Behavior Prediction: Many successful temporal prediction methods are based on latent factor models, e.g., PLSA or LDA. In [16], a temporal model called TCAM is proposed to predict users' behaviors, which considers users' interests and the temporal context. In [17], Song *et al.* develop a model to predict the human emergency behavior when natural disasters happen. In [18], Zhang *et al.* address the problem of inferring continuous dynamic users' behavior by utilizing both the social influence and the personal preference.

EM Algorithm: The Expectation-Maximization (EM) algorithm is a broadly used method to compute the maximum likelihood estimates, which benefits a variety of incomplete data problems [19]. The EM algorithm is originally proposed by Dempster, Laird, and Rubin [20]. For models with potential variables, it is difficult to find the maximum likelihood directly. The EM algorithm provides a solution to such problems. As an iterative algorithm, there are two steps in each iteration of the EM algorithm — the Expectation (E) step and the Maximization (M) step. In E-Step, the maximum likelihood can be computed by the estimated value of latent parameters. In M-Step, the parameters are then re-estimated by the maximum likelihood which is got in the E-Step. The algorithm iteratively proceeds E-Step and M-Step until convergence. The EM algorithm is first applied in statistical areas and then broadly used in almost all fields where statistical techniques have been applied [21], [22]. In addition, with the development of the computer science, EM has already become a widely applied method in the research of machine learning [23], behavior analysis [18], computer vision [24] and data clustering [25].

The most related work to ours is [26] whose journal version is [27]. However, there are major differences between ours and

[26]. (1) Our methods can predict the exact re-hotting time points of a given once-hot topic, which is more difficult and precise. But the methods in [26] could only predict the time window of hot topics and cannot predict the exact time points. (2) Besides, the UTP model combines the users' interests [18], friend-circles [28] and unexpected events (e.g., the Zika virus spreading explosively across the Americas at early 2016) in OSNs, which is more comprehensive than the CPB model in [26] which does not combine the factors together to predict the results. (3) What's more, as an unsupervised approach, our re-hotting method is more applicable to evaluate whether a once-hot topic can be hot again and will be re-hot at which time points, since it is difficult to achieve a widely accepted ground truth. However, the work of [26] only focuses on a time window where the certain topic might be hot, which is a kind of supervised methods and has been studied a lot by many researchers.

III. Problem Description

In this section, we present a couple of preliminary definitions and the formal description of the topic re-hotting prediction problem.

A *topic* could be considered as a set of indexed terms [29]. In this work, topic v is composed of several keywords that can be searched in an OSN.

Let $\mathbb{U}$, $\mathbb{T}$, and $\mathbb{V}$ be the set of users, the set of time points, and the set of topics, respectively. $\boldsymbol{C}$ is an $N \times T \times V$ *Cuboid*, where $N = |\mathbb{U}|$, $T = |\mathbb{T}|$, and $V = |\mathbb{V}|$. A data point $C[u, t, v]$ indicates the number of messages, which are associated with a specific topic v, posted by user u at time point t.

When a user posts a message which associates with a specific topic, we call this a "User Topic Participation (UTP)". Furthermore, in this paper we argue that the posting behavior is influenced by friend-circles, types of topics and unexpected events, and use a triple $(u, t, v) \in (\mathbb{U}, \mathbb{T}, \mathbb{V})$ to describe UTP, in which user u posts a message that is about topic v at time point t.

There are two related tasks to predict the re-hotting time for the specific topic. The first task is to predict the topic emergence which aims to detect the hot topics. The second task is to predict the exact re-hotting time for the once-hot topics. Since the first task for topic detection has been widely studied [9]–[11], we mainly focus on the second task to predict the exact re-hotting time for the once-hot topics in this work. In the following, we give the topic re-hotting prediction problem.

The TRP Problem: Given the observed data $\boldsymbol{D} = (\mathbb{U}, \mathbb{T}, \mathbb{V})$, for a once-hot topic v, the problem of topic re-hotting prediction (TRP) aims to predict the *accurate time points* when topic v should be re-hot. The topic v in this problem must have a re-hot time, which means that v is hot at least twice. Notice that the accurate time points come from the next period of time instead of just the next time point. The TRP problem estimates the participation probability of $(1/N)\sum_u P(v|u,t)$ that users would participate in v at t. However, at different time points the participation probability may change differently. For this problem, we aim to capture the change of topics, and use $\widetilde{\boldsymbol{g}}_v[\mathrm{t}] = (1/N)\sum_u P(v|u,t)$ to store the different participation probability values about topic v at different time points. The prediction result is a ranked list T' consisting of candidate time points which are generated as the re-hotting time, ordered by the participation probability.

TABLE I
Notations Used in This Paper

Symbol	Description
u, t, v	user u, time point t, topic v
$\mathbb{U}, \mathbb{T}, \mathbb{V}$	$\mathbb{U}, \mathbb{T}, \mathbb{V}$ are the set of users, time points, and topics, respectively
f, z, e	friend-circle f, type of topics z, unexpected event e
N, V, T, F, Z, K	number of users, topics, time points, friend-circles, types of topics, unexpected events
s	the parameter to determine the topic is generated according to either unexpected events or users interests.
$\phi_f^{u,t}$	probability that user u chooses a friend-circle f at time point t, $\sum_{i=1}^{F} \phi_{f_i}^{u,t} = 1$
$\boldsymbol{\phi}^{u,t}$	friend-circles of user u at time point t
$\alpha_z^{f,t}$	probability that friend-circle f chooses a type of topics z at time point t, $\sum_{i=1}^{Z} \alpha_{z_i}^{f,t} = 1$
$\boldsymbol{\alpha}^{f,t}$	the types of topics of friend-circle f at time point t
$\beta_v^{z,t}$	probability that the type of topics z chooses a topic v at time point t, $\sum_{i=1}^{V} \beta_{v_i}^{z,t} = 1$
$\boldsymbol{\beta}^{z,t}$	the topics in the type of topics z at time point t
θ_e^t	probability that unexpected event e happens at time point t, $\sum_{i=1}^{K} \theta_{e_i}^t = 1$
$\boldsymbol{\theta}^t$	unexpected events that happen at time point t
$\lambda_v^{u,e}$	probability that user u chooses topic v when unexpected event e happens, $\sum_{i=1}^{V} \lambda_{v_i}^{u,e} = 1$
$\boldsymbol{\lambda}^{u,e}$	topics which are chosen by user u when unexpected event e happens
$\tau^{u,t}$	the mixing weight for user u at time point t, which is adopted to compute the value of s to determine the topic v is generated according to either unexpected events or user's interest for user u at time point t
$\boldsymbol{\Psi}$	all the parameter configurations $\boldsymbol{\Psi} = \{\boldsymbol{\phi}^{u,t}, \boldsymbol{\alpha}^{f,t}, \boldsymbol{\beta}^{z,t}, \boldsymbol{\theta}^t, \boldsymbol{\lambda}^{u,e}, \tau^{u,t}\}$
$\boldsymbol{C}$	an $N \times T \times V$ cuboid. A data point $C[u, t, v]$ indicates the number of messages which is posted by user u at time point t, and these messages are associated with a specific topic v.
I	the number of Gaussian distributions

The notations used in this paper are summarized in Table I.

IV. Temporal UTP Behavior Modeling

In this section, we present the temporal user topic participation (UTP) model. This model is capable of explaining a user's behavior of posting a message which is associated with a specific topic in social networks.

A. Event-Driven UTP Model

The Event-driven UTP (E-UTP) model pays more attention to the influence of unexpected events on topics. An *unexpected event* is an external event, such as a terrorist attack, a disease outbreak, or a traffic accident [30].

Definition 1 (unexpected event): An unexpected event $e = (sth, tp)$ is an external event, where sth represents the description for the external event and tp means the time point when the external event happens.

In this work, an event is called an external event for a system if the state of the system changes when the event occurs (i.e., an environmental component acts upon a system component). We

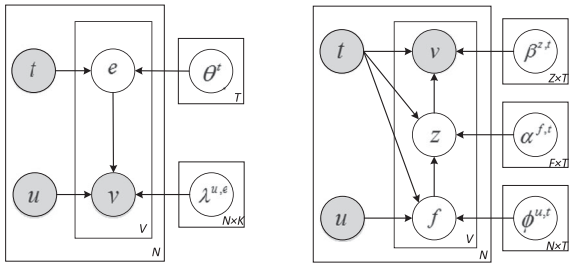


Fig. 2. The graphical representation of E-UTP and I-UTP. (a) E-UTP. (b) I-UTP.

refer readers to [31] for the systematical description and deeply analysis of both events and external events.

Usually, an unexpected event happens at the specific time. A topic emerges and becomes "hot" when the keywords that the topic consists of appear in many news articles within a short time period. Sometimes, a hot topic may be triggered by an unexpected event. For instance, "Artificial Intelligence vs. Human Intelligence" becomes a hot topic since the Alpha Go defeated a famous human Go grandmaster on March 15, 2016. While users' behavior of posting messages may be influenced by their interests, unexpected events will also influence users' behaviors. In our proposed model, unexpected events are denoted by a latent variable [32], which is a random variable whose actual values are not directly observed. At time point t, unexpected events can be denoted by $\boldsymbol{\theta}^t$, and $\theta^t_{e_k}$ is the probability that unexpected event e_k happens at t, $\boldsymbol{\theta}^t \triangleq \{\theta^t_{e_1}, ..., \theta^t_{e_K}\}$, where $1 \leq k \leq K$, and K is the number of unexpected events.

As shown in Fig. 2(a), a hidden variable e denotes unexpected events in the E-UTP model. Then, the model can be explained as follows: An unexpected event e happens when a user u browses information in OSNs at time point t. The event e is likely to affect u's behavior of posting messages which are associated with specific topics. It means when choosing topics at t, user u may tend to make decisions based on the unexpected event e, such as a terrorist attack.

Let $\boldsymbol{\lambda}^{u,e} \triangleq \{\lambda^{u,e}_{v_1}, \ldots, \lambda^{u,e}_{v_V}\}$ be the topics which are chosen by user u when an unexpected event e happens, and $\lambda^{u,e}_{v_i}$ be the probability that u chooses topic v_i when e happens, where $1 \leq i \leq V$. The generative process of the E-UTP model is:

1) Sample $e \sim Multinomial(\boldsymbol{\theta}^t)$,
2) Sample $v \sim Multinomial(\boldsymbol{\lambda}^{u,e})$.

Then, given user u, time point t and event e, the probability of choosing topic v to participate in is:

$$P(v|u,t;\boldsymbol{\theta}^t,\boldsymbol{\lambda}^{u,e}) = \sum_e P(v|u,e;\boldsymbol{\lambda}^{u,e})P(e|t;\boldsymbol{\theta}^t). \quad (1)$$

B. Interest-Driven UTP Model

Different from the E-UTP model, the Interest-driven UTP (I-UTP) model takes into account users' interests. Users' interests are influenced by many factors, in which users' friend-circles [33] and the types of topics are the main influencers.

A *friend-circle* of a user u is a subset of u's followers who have similar interests with u. All the followers of u are divided into many different friend-circles. Thus, a friend-circle usually does not contain all the followers.

Definition 2 (friend-circle): Let F_u be the set of all the followers of u. A friend-circle f of u is a two-tuple $(user\_set, ty)$ where $user\_set \subseteq F_u$ represents a subset of F_u and ty means the certain type of topics, in which all users in $user\_set$ are interested.

The users in the same friend-circle tend to have similar interests while the users in different friend circles might have different interests. Users' behavior of participating in a topic may come from the interest of their friend circles. For example, a user may be interested in a friend-circle in a certain period of time, and this friend-circle is always involved in some topics, like "Grammy Award", "Country music", and "Billboard Music Awards". It means this user likes the American songs and her/his friend-circle is the circle of "American songs" in this period of time. In our model, the friend-circle is denoted by another latent variable which represents hidden subsets of the user's followers. Given a user u, at time point t, u's friend-circles set is denoted by $\boldsymbol{\phi}^{u,t}$, and $\boldsymbol{\phi}^{u,t} \triangleq \{\phi^{u,t}_{f_1}, ..., \phi^{u,t}_{f_F}\}$, where $\phi^{u,t}_{f_i}$ is the probability that u chooses the friend-circle f_i at t, $1 \leq i \leq F$, and F is the number of friend-circles.

Topics can be classified into different types, such as news, quotes, jokes, and other four types [34]. Furthermore, users' friend-circles may focus on talking about different *types of topics* at different time points. Given a friend-circle f, at time point t, f's interested types of topics, i.e., f's topic-type membership, can be denoted by $\boldsymbol{\alpha}^{f,t} \triangleq \{\alpha^{f,t}_{z_1}, ..., \alpha^{f,t}_{z_Z}\}$, where $\alpha^{f,t}_{z_j}$ is the probability that f chooses a topic type z_j at t, $1 \leq j \leq Z$, and Z is the number of topic types.

As shown in Fig. 2(b), the symbols f and z are used to present the hidden friend-circles and types of topics in the I-UTP model. At time point t, user u posts a message which is associated with a specific topic v, and this behavior may be the result of the co-effect of f and z. At different time points, u may make friends with different people and these people may be interested in different types of topics. For example, a person who is interested in the fashion-circle will always post messages which are all about fashion shows or fashion trends. However, as time goes by, (s)he may love literature in the future.

Let $\boldsymbol{\beta}^{z,t} \triangleq \{\beta^{z,t}_{v_1}, \ldots, \beta^{z,t}_{v_V}\}$ be the topics included in topic type z at time point t, and $\beta^{z,t}_{v_i}$ be the probability that z chooses a topic v_i at t, where $1 \leq i \leq V$.

The generative process of the I-UTP model is:

1) Sample $f \sim Multinomial(\boldsymbol{\phi}^{u,t})$,
2) Sample $z \sim Multinomial(\boldsymbol{\alpha}^{f,t})$,
3) Sample $v \sim Multinomial(\boldsymbol{\beta}^{z,t})$.

Then, given user u, time point t, friend-circle f and topic type z, the probability of choosing topic v to participate in is:

$$P(v|u,t;\boldsymbol{\phi}^{u,t},\boldsymbol{\alpha}^{f,t},\boldsymbol{\beta}^{z,t}) = \sum_f \sum_z P(f|u,t;\boldsymbol{\phi}^{u,t})P(z|f,t;\boldsymbol{\alpha}^{f,t})P(v|z,t;\boldsymbol{\beta}^{z,t}). \quad (2)$$

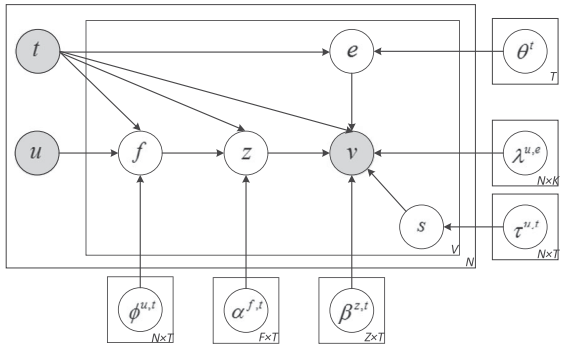


Fig. 3. The Graphical Representation of UTP.

C. Overall UTP Model

The Overall UTP (abbr. UTP model later if there is no ambiguity) model conflates the E-UTP model and the I-UTP model, in which a hidden variable s is adopted as a switch to determine the topic is generated according to either unexpected events or users' interests. As shown in Fig. 3, when $s = 0$, the unexpected events are the main influencer; when $s = 1$, users' behavior is heavily influenced by their own interests. In particular, s is user-time-specific, which means users may make different choices at different time points.

Let $\tau^{u,t}$ be the mixing weight for user u at time point t, and $\mathbf{\Psi}$ be the configuration of all parameters, i.e., $\mathbf{\Psi} = \{\boldsymbol{\phi}^{u,t}, \boldsymbol{\alpha}^{f,t}, \boldsymbol{\beta}^{z,t}, \boldsymbol{\theta}^{t}, \boldsymbol{\lambda}^{u,e}, \tau^{u,t}\}$.

The overall generative probability of the behavior that u participates in the discussion of topic v at t in the UTP model is:

1) Sample $s \sim Bernoulli(\tau^{u,t})$,
2) If $s = 0$
   a) Sample $e \sim Multinomial(\boldsymbol{\theta}^{t})$,
   b) Sample $v \sim Multinomial(\boldsymbol{\lambda}^{u,e})$,
3) If $s = 1$
   a) Sample $f \sim Multinomial(\boldsymbol{\phi}^{u,t})$,
   b) Sample $z \sim Multinomial(\boldsymbol{\alpha}^{f,t})$,
   c) Sample $v \sim Multinomial(\boldsymbol{\beta}^{z,t})$.

Given $\boldsymbol{C}$ and $\mathbf{\Psi}$, the log-likelihood can be written as:

$$L(\mathbf{\Psi}; \boldsymbol{C}) = \sum_{u}\sum_{t}\sum_{v} C[u,t,v] \log P(v|u,t;\mathbf{\Psi}) \tag{3}$$

where $P(v|u,t;\mathbf{\Psi})$ is defined as

$$P(v|u,t;\mathbf{\Psi}) = \tau^{u,t}\left(\sum_{f}\sum_{z} P(f|u,t;\mathbf{\Psi})P(z|f,t;\mathbf{\Psi})P(v|z,t;\mathbf{\Psi})\right) + (1-\tau^{u,t})\left(\log\sum_{e} P(v|u,e;\mathbf{\Psi})P(e|t;\mathbf{\Psi})\right). \tag{4}$$

V. Inference Algorithm and Rehot Prediction

In this section, we first infer the parameters in the UTP model, and then propose the UTP model based method to predict the accurate time points when a topic should be re-hot.

A. Parameters Inference of the UTP Model

In order to infer the hidden variables in the UTP model, the Expectation Maximization Gaussianization (EMG) method is utilized since the parameters cannot be obtained directly. Also, the continuous dynamic change of topics can be captured by the EMG approach.

1) E-Step: For a given data point $C[u,t,v]$, at first we compute the posterior probability of the hidden variables.

The posterior distribution of hidden variable e:

$$P(e|u,t,v;\widehat{\mathbf{\Psi}}) = P(e|s=0,u,t,v;\widehat{\mathbf{\Psi}}) P(s=0|u,t,v;\widehat{\mathbf{\Psi}}), \tag{5}$$

where

$$P(e|s=0,u,t,v;\widehat{\mathbf{\Psi}}) = \frac{P(v|u,e;\widehat{\mathbf{\Psi}})P(e|t;\widehat{\mathbf{\Psi}})}{\sum_{e'} P(v|u,e';\widehat{\mathbf{\Psi}})P(e'|t;\widehat{\mathbf{\Psi}})}. \tag{6}$$

The posterior distribution of hidden variables f and z:

$$P(f,z|u,t,v;\widehat{\mathbf{\Psi}}) = P(f,z|s=1,u,t,v;\widehat{\mathbf{\Psi}}) P(s=1|u,t,v;\widehat{\mathbf{\Psi}}), \tag{7}$$

where

$$P(f,z|s=1,u,t,v;\widehat{\mathbf{\Psi}}) = \frac{P(f|u,t;\widehat{\mathbf{\Psi}})P(z|f,t;\widehat{\mathbf{\Psi}})P(v|z,t;\widehat{\mathbf{\Psi}})}{\sum_{f'}\sum_{z'} P(f'|u,t;\widehat{\mathbf{\Psi}})P(z'|f',t;\widehat{\mathbf{\Psi}})P(v|z',t;\widehat{\mathbf{\Psi}})}. \tag{8}$$

The posterior distribution of hidden variable s:

$$P(s|u,t,v;\widehat{\mathbf{\Psi}}) = \frac{s\tau^{u,t}P_1 + (1-s)(1-\tau^{u,t})P_2}{\tau^{u,t}P_1 + (1-\tau^{u,t})P_2}, \tag{9}$$

where

$$P_1 = \sum_{f}\sum_{z} P(f|u,t;\widehat{\mathbf{\Psi}})P(z|f,t;\widehat{\mathbf{\Psi}})P(v|z,t;\widehat{\mathbf{\Psi}}),$$

$$P_2 = \sum_{e} P(v|u,e;\widehat{\mathbf{\Psi}})P(e|t;\widehat{\mathbf{\Psi}}).$$

In Equation (9), s has two states. When $s = 1$, it means at time point t, user u participates in the discussion of topic v from her/his own interests. However, when $s = 0$, it indicates that at t, the unexpected events influence the decision of u to participate in the discussion of v.

Afterwards, we could compute the expectation of the log-likelihood of all the observed and hidden data:

$$\begin{aligned} Q(\mathbf{\Psi};\widehat{\mathbf{\Psi}}) = &\sum_{u}\sum_{t}\sum_{v} C[u,t,v]\left(\sum_{f}\sum_{z} P(f,z|u,t,v;\widehat{\mathbf{\Psi}})\right. \\ &\times \log\left(\tau^{u,t}P(f|u,t;\widehat{\mathbf{\Psi}})P(z|f,t;\widehat{\mathbf{\Psi}})P(v|z,t;\widehat{\mathbf{\Psi}})\right) \\ &+ \sum_{e} P(e|u,t,v;\widehat{\mathbf{\Psi}})\log\Big((1-\tau^{u,t})P(v|u,e;\widehat{\mathbf{\Psi}}) \\ &\left.\times P(e|t;\widehat{\mathbf{\Psi}})\Big)\right). \end{aligned} \tag{10}$$

2) M-Step: For M-Step, we find the estimation $\mathbf{\Psi}$ by maximizing the Q-function, which needs to be incorporated with other constraints: $\sum_f P(f|u,t;\boldsymbol{\phi}^{u,t})=1$, $\sum_z P(z|f,t;\boldsymbol{\alpha}^{f,t})=1$, $\sum_v P(v|z,t;\boldsymbol{\beta}^{z,t})=1$, $\sum_e P(e|t;\boldsymbol{\theta}^t)=1$, and $\sum_v P(v|u,e;\boldsymbol{\lambda}^{u,e})=1$.

Therefore, we can easily obtain the optimal values:

$$P(f|u,t;\boldsymbol{\phi}^{u,t}) = \frac{\sum_z \sum_v C[u,t,v]P(f,z|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_{f'}\sum_v\sum_z C[u,t,v]P(f',z|u,t,v;\widehat{\mathbf{\Psi}})}, \tag{11}$$

$$P(z|f,t;\boldsymbol{\alpha}^{f,t}) = \frac{\sum_u \sum_v C[u,t,v]P(f,z|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_u\sum_{z'}\sum_v C[u,t,v]P(f,z'|u,t,v;\widehat{\mathbf{\Psi}})}, \tag{12}$$

$$P(v|z,t;\boldsymbol{\beta}^{z,t}) = \frac{\sum_u \sum_f C[u,t,v]P(f,z|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_u\sum_f\sum_{v'} C[u,t,v]P(f,z|u,t,v';\widehat{\mathbf{\Psi}})}, \tag{13}$$

$$P(e|t;\boldsymbol{\theta}^t) = \frac{\sum_u \sum_v C[u,t,v]P(e|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_u\sum_v\sum_{e'} C[u,t,v]P(e'|u,t,v;\widehat{\mathbf{\Psi}})}, \tag{14}$$

$$P(v|u,e;\boldsymbol{\lambda}^{u,e}) = \frac{\sum_t C[u,t,v]P(e|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_t\sum_{v'} C[u,t,v]P(e|u,t,v';\widehat{\mathbf{\Psi}})}, \tag{15}$$

$$\tau^{u,t} = \frac{\sum_v C[u,t,v]P(s=1|u,t,v;\widehat{\mathbf{\Psi}})}{\sum_v\sum_s C[u,t,v]P(s|u,t,v;\widehat{\mathbf{\Psi}})}. \tag{16}$$

3) G-Step: In G-Step, we sample t from the Gaussian mixture distribution. Some other distributions can also be used, however the Gaussian mixture distribution fits the topic trends quite well, as we can see in experiments later.

Following the idea of [35], [36], we assume that the number of users' behaviors for participating in a specific topic v obeys the Gaussian mixture distribution on time t, i.e., $g_v(t;\mathbf{\Psi}) = \sum_u P(v|u,t;\mathbf{\Psi})$, where g_v represents the Gaussian mixture distribution of topic v. Then, the Gaussian mixture distribution on t can be defined as follows:

$$g_v(t;\mathbf{\Psi}^*) = \sum_{i=1}^{I} a_i \cdot \exp\left\{-\frac{(t-\mu_i)^2}{2\sigma_i^2}\right\} \tag{17}$$

where I is the number of Gaussian distributions and controls the complexity of the Gaussian mixture distribution. a_i denotes the weight of a Gaussian distribution, and $\sum_{i=1}^{I} a_i = 1$.

It is difficult to determine the number of Gaussian mixture distributions for a particular topic. Empirically, we could estimate the appropriate number of Gaussian mixture distributions as three or four based on the experimental evaluation in this study.

The aim of G-Step is to get the parameters a, μ, and σ, which can be obtained by another EM algorithm. Thus, the log-likelihood in G-Step can be defined as follows:

$$L(t;\mathbf{\Psi}^*) = \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} a_i \cdot \exp\left\{-\frac{(t-\mu_i)^2}{2\sigma_i^2}\right\}\right) \tag{18}$$

where M is the total number of the sample data, i.e., the total number of possible user participation.

As there is a summation inside the log function, the hidden parameters cannot be obtained by derivation directly. Thus, firstly, for each time point t, we calculate the probability generated by the i-th Gaussian distribution. Secondly, the value of each estimated parameter can be obtained.

Step 1: The posterior distribution of $\mathcal{I}=i$ is:

$$P(\mathcal{I}=i \mid t;\mathbf{\Psi}^*) = \frac{a_i \cdot \exp\left\{-\frac{(t-\mu_i)^2}{2\sigma_i^2}\right\}}{\sum_i a_i \cdot \exp\left\{-\frac{(t-\mu_i)^2}{2\sigma_i^2}\right\}}; \tag{19}$$

Step 2: The optimal values can be obtained as follows, where $\widehat{\mathbf{\Psi}^*}$ is the $\mathbf{\Psi}^*$ in the previous iteration.

$$\mu_i = \frac{1}{M_i}\sum_m P(\mathcal{I}=i|t;\widehat{\mathbf{\Psi}^*})t, \tag{20}$$

$$\sigma_i^2 = \frac{1}{M_i}\sum_m P(\mathcal{I}=i|t;\widehat{\mathbf{\Psi}^*})(t-\mu_i)^2, \tag{21}$$

$$a_i = \frac{M_i}{M}, \tag{22}$$

$$M_i = \sum_m P(\mathcal{I}=i|t;\widehat{\mathbf{\Psi}^*}). \tag{23}$$

At the end of G-Step, we iteratively execute Steps 1 and 2 until convergence. Then, the expression of the Gaussian mixture distribution can be obtained. Notice that different sizes of I lead to different performance. The larger size of I exploits more Gaussian distributions, and then leads to better performance. However, it also has the risk of over-fitting. Therefore, the appropriate size of I varies from topic to topic.

The overall EMG algorithm starts with the random parameters $\mathbf{\Psi}^*$, and then repeats E-Step, M-Step, and G-Step iteratively to improve the estimates of parameters until convergence.

Theorem 1: The proposed EMG method is convergent.

Proof: We can prove the convergence of the EMG method by comprehensively considering the boundedness and the monotonicity of the log-likelihood. The details of the proof are presented in Appendix A. ■

B. Predicting Topic Rehotting Time Points

In this subsection, we first introduce a weighting scheme which can smooth the fluctuations of topics, so that the rehotting time points can be predicted accurately. Afterwards, the temporal re-hotting prediction method is proposed.

1) Enhancement of UTP: In this part, we propose a weighting scheme to improve the UTP model. In this scheme, our goal is to reduce the influence of topic fluctuation, so that we can predict the re-hotting time points more accurately.

As we know, the change of topics is fluctuant. It is not always the most appropriate re-hotting time point when a topic just rises slightly, since this small increment may decline immediately after a short period of time. We believe that the most appropriate re-hotting time point should be the time point when a topic has enough increments.

To address the problem of the fluctuation of topics, we propose a weighting scheme to reduce the influence of the topics' slight ascending trend. When a topic declines, we compute the weight for its each time point. The specific weight calculation method is as follows:

$$w(v,t) = \begin{cases} C[u,t,v]; & t \text{ is no later than } t_{\text{Max}(v)} \\ \frac{\sum_{i=t_{\text{Max}(v)}}^{t} V_i(v)}{\text{Max}(v) - V_t(v)}; & t \text{ is later than } t_{\text{Max}(v)} \end{cases} \tag{24}$$

where $\sum_{i=t_{\text{Max}(v)}}^{t} V_i(v)$ indicates the cumulative impact of topic v after the topic declines, and $V_t(v) = \sum_u C[u,t,v]$ is the total number of messages which are associated with topic v at time point t. $Max(v)$ is the largest number of messages which are associated with topic v among every time point in v's whole lifecycle, and $Max(v) = \max\{V_t(v)\}$, $t \in [t_0, t_T]$, where t_0 and t_T mean the start time point and the end time point of topic v, respectively. For Equation (24), in order to reduce the influence of topic fluctuation, we give v a smaller weight at t.

The meaning of Equation (24) is to compute the weight for a topic v at time point t when v declines. In the numerator, we compute $\sum_{i=t_{\text{Max}(v)}}^{t} V_i(v)$ to accumulate the number of messages that users participate in the topic v at time point t. In the denominator, we compute the difference of $Max(v)$ and $V_t(v)$. This weight calculated by Equation (24) can be used to reduce the fluctuation because it can help us avoid predicting the non-re-hot time points to be the re-hot time points. If the number of participations of the users is low, the denominator value in Equation (24) is high. Although the accumulation of users is high, the $w(v,t)$ calculated by Equation (24) is low because of the high value of the denominator. Thus, according to the low value of $w(v,t)$, we can predict that this time point is not a re-hot time point. In this way, Equation (24) helps us reduce the fluctuation of topics. The motivation of this method is to reduce the influence of the topics' slight ascending trend and improve the accuracy of prediction for re-hot time points.

Integrating the weights of topics defined in Equation (24), we obtain the weighted cuboid $\widehat{\boldsymbol{C}}$ from the original $\boldsymbol{C}$ as follows:

$$\widehat{C}[u,t,v] = C[u,t,v]w(v,t), \tag{25}$$

which can be used in the UTP model.

2) Rehot Prediction: Based on the above generated temporal model on users' posting behaviors, we can predict the appropriate time point when a topic should be re-hot.

In the prediction problem, we concern that at some specific time points whether we should re-hot a topic or not. When given a set of users during time interval $[t_s, t_e]$, the probability of re-hotting topic v in time interval $[t_s, t_e]$ can be calculated as:

$$P(v|[t_s,t_e];\boldsymbol{\Psi}) = \int_{t_s}^{t_e} g_v(t;\boldsymbol{\Psi})\,\mathrm{d}t \tag{26}$$

where t_s indicates the start of the interval, and t_e indicates the end of that.

The overall re-hot topics prediction algorithm is outlined in Algorithm 1. First, the input $\boldsymbol{C} = [\mathbb{U}, \mathbb{T}, \mathbb{V}]$ is enhanced by Equation (25) to reduce the influence of topics fluctuation (Line 1), and the set of parameters $\boldsymbol{\Psi}^*$ is initialized with random values (Line 2). Then, E-Step (Line 5), M-Step (Line 6), and G-Step (Line 7) are repeated iteratively to improve the estimation of parameters until the convergence (Lines 3–8). Finally, the re-hot topics set $\mathcal{T}$ is generated by Equation (26) (Line 9).

Algorithm 1: Re-Hot Topic Prediction.

Input: $\boldsymbol{C} = [\mathbb{U}, \mathbb{T}, \mathbb{V}]$, time interval $[t_s, t_e]$
Output: The set of re-hot topics $\mathcal{T}$

1: Enhance $\boldsymbol{C}$ by Equation (25) to reduce the influence of topics fluctuation;
2: Initialize the set of parameters $\boldsymbol{\Psi}^*$ with random values;
3: Repeat the following steps until convergence to obtain the optimal parameters $\boldsymbol{\Psi}^*$
4: {
5: Compute the posterior probability of the hidden variables by Equations (5)–(9);
6: Find the estimation $\boldsymbol{\Psi}$ by maximizing the Q function, which needs to be incorporated with other constraints $\sum_f P(f|u,t;\boldsymbol{\phi}^{u,t}) = 1$, $\sum_z P(z|f,t;\boldsymbol{\alpha}^{f,t}) = 1$, $\sum_v P(v|z,t;\boldsymbol{\beta}^{z,t}) = 1$, $\sum_e P(e|t;\boldsymbol{\theta}^t) = 1$, and $\sum_v P(v|\,u,e;\boldsymbol{\lambda}^{u,e}) = 1$.
7: Obtain the parameters a, μ, and σ by maximizing the Equation (18);
8: }
9: Predict the set of re-hot topics $\mathcal{T}$ by computing the probability of each topic v in time interval $[t_s, t_e]$ by Equation (26);
10: Return the re-hot topics set $\mathcal{T}$ in time interval $[t_s, t_e]$.

The time complexity of E-Step is $O(T \cdot V \cdot (m+n) \cdot (F \cdot Z + K))$, that of M-Step is $O(T \cdot V(m+n) \cdot (F \cdot Z + K + 1) + T \cdot ((m+n) \cdot F + F \cdot Z + V \cdot Z) + K \cdot (T+V))$, and that of G-Step is $O(T \cdot I)$. In addition, the Enhancement computation is constant time, which can be ignored. Thus, the time complexity of our method is $O(T \cdot V \cdot (m+n) \cdot (F \cdot Z + K + 1) + T((m+n) \cdot F + F \cdot Z + V \cdot Z + I) + K \cdot (T+V))$ where $m+n$ is the size of training data and testing data.

VI. Experiments

In this section, we evaluate the performance of our proposed models by various metrics. At first, the experimental setup is described. Then, experimental results on the topic re-hotting prediction are presented. Next, the analysis of several parameters which influence the performance of the UTP model is introduced. Finally, a case study of the TRP problem is brought forward.

A. Experimental Setup

In this subsection, we introduce the data sets, metrics and the competing models which are used in the following experiments.

1) Data sets: In this work, our experiments are conducted on three real-world data sets crawled from Twitter and Weibo, separately. The statistics of these data sets are shown in Table II.

TABLE II
STATISTICS OF THE THREE REAL-WORLD DATA SETS

	DST	DSW	DST2
Number of tweets/microblogs	31,346	86,160	219,604
Number of topics	15	24	100
Number of users	6471	27,662	34,848

DST. Twitter is one of the most popular OSNs all over the world. Our crawled Twitter data set DST contains 6471 users, 34,838 tweets, and 15 different topics; the time span is from January 1, 2010 to February 1, 2013. The 15 topics involve "NBA", "Golden Globe Awards", TV show "Big Bang Theory", and so on.

DST2. Another data set DST2 which is also extracted from Twitter is a denser data set. DST2 contains 34,848 users, 219,604 tweets, and 100 different topics; the time span is from January 1, 2009 to December 31, 2009.

DSW. Weibo is one of the most popular OSNs in China. It allows users to share information with others using at most 140 words per message. We crawled Weibo data from February 11, 2014 to March 3, 2014. The generated data set DSW includes 27,662 users, 86,160 microblogs, and 24 different topics. These topics involve "the Situation of Egypt", "Conflict between North and South Korea", "US Midterm Election", and so forth.

Please note that the three data sets are adopted and utilized after some kind of preprocessing. The data preprocessing filters out the tweets that do not conform to our requirements as follows. Firstly, the tweets which are irrelevant to any topics, such as tweets to express personal emotions, are removed. Besides, the tweets which are not related to certain topics that can be hot at least once are filtered out. Finally, we choose the tweets corresponding to the topics that can be re-hot at least once to obtain the resultant data sets. After the above-mentioned data preprocessing, the tweets that meet our requirements are sparser than the initial data sets. However, the refined data sets become more suitable for the experiments.

The time point in our model can be refined to month, day and hour, even to minute and second. However, in order to verify the effectiveness of our model more clearly and directly, we have to choose an appropriate time point according to the changes of the topics. The time points in Weibo data sets are measured in days, and those in Twitter data sets are measured in months. In our model, the unexpected event and the friend circle are denoted by two latent variables which are not observed in the data sets. In the experiments, we use 80% of each data set for training, and the rest 20% for testing.

2) Ground Truth: For a given hot topic, the aim of this study is how to predict some appropriate time points when the original topic could become popular again easily. Obviously, it is very difficult to get the ground truth for verification of our proposed method since (1) there is indeed complicated to make a topic hot, and (2) it is not easy to judge whether or not a topic has become popular at an exact time point. Thus, we deeply analyze the process of ascending a topic trend, and consider some time points (e.g., the number of participants at this time point is $n\%$ higher than that at the previous) as the most suitable ones to re-hot the topic, in the process of this ascension.

It is important to note that the tendency of topics is not stable, and often presents the situation of fluctuation. In the process of the topic trend change, there may exist a situation where the trend of a topic slightly ascends first and then declines quickly. Obviously, there is no need to predict the time points in that situation. In another word, too small values of n are not stable. In sum, in order to avoid mistakenly excluding suitable time points, the value of n could not be too large; in order to avoid the interference of fluctuation, it could not be too small. In this work, we set $n = 10$ by default, which means if the number of participants of the given topic at a time point is 10% higher than that at the previous one, then we think at this time point the topic could be easily re-hot. Therefore, this time point is considered as one of the ground truth. It should be noted that here we just give a method to get the ground truth, and the specific value of n should be chosen reasonably according to the practical application.

3) Metrics: In this study, three metrics, i.e., $Precision$, $Recall$, and F_1-$measure$, are used to assess the quality of re-hotting prediction. The definitions are given as follows.

$$Precision = \#hit/k \tag{27}$$

where k is the number of predicted time points, and $\#hit$ is the number of correct predicted ones.

$$Recall = \#hit/g \tag{28}$$

where g is the total number of topics' correct re-hotting time points. $Recall$ refers to how many entries are retrieved accurately over the all accurate re-hotting time points.

F_1-$measure$ is a comprehensive evaluation metric defined as

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{29}$$

which is based on $Precision$ and $Recall$.

4) Competing Models: To the best of our knowledge, there is no existing method to address the problem of TRP. Therefore, we use the E-UTP and I-UTP models as competing models to our proposed UTP model. Also, the EMG inference method (**+G**) and the Enhancement method (**+En**) are evaluated.

1) *E-UTP:* This event-driven UTP model predicts each time point when the probability of user's participation of a topic is higher than its former one as the suitable re-hot time point. Notice that E-UTP is inferred only by the EM algorithm which is without G-Step and Enhancement.
2) *I-UTP:* For the interest-driven UTP model, the inference and prediction methods are similar to that of E-UTP.
3) *UTP:* For the overall UTP model, the inference and prediction methods are also similar to that of E-UTP and I-UTP.
4) *UTP + G:* The inference method of the UTP + G model is EMG algorithm. Compared with UTP, UTP + G only predicts the time points when the participation probabilities of a topic are some of the highest values. These participation probabilities can be calculated by Equation (26).

TABLE III
PREDICTION PERFORMANCE OF DIFFERENT METHODS@DSW

	$Precision$	$Recall$	F_1	$Time\,cost$ (min)
E-UTP	0.157	0.467	0.235	0.086
I-UTP	0.133	0.332	0.189	0.136
UTP	0.166	0.505	0.250	3.591
UTP + G	0.332	0.253	0.285	3.249
UTP + En	0.177	**0.507**	0.313	3.655
UTP + G + En	**0.368**	0.294	**0.324**	**3.795**

TABLE IV
PREDICTION PERFORMANCE OF DIFFERENT METHODS@DST

	$Precision$	$Recall$	F_1	$Time\,cost$ (min)
E-UTP	0.304	0.322	0.313	0.025
I-UTP	0.287	0.313	0.299	0.021
UTP	0.340	0.323	0.331	0.402
UTP + G	0.354	0.314	0.333	0.416
UTP + En	0.351	**0.406**	0.377	0.420
UTP + G + En	**0.406**	0.353	**0.378**	**0.433**

5) *UTP + En:* The UTP + En model combines UTP with the above-mentioned Enhancement method. Different from the UTP model, UTP + En can better reduce the fluctuation influence on topic re-hotting. In addition, the inference method of UTP + En is the same with that of E-UTP.
6) *UTP + G + En:* The UTP + G + En model is the comprehensive one, which combines UTP + G and UTP + En.

B. Evaluation

In this subsection, we first report the results of the topic re-hotting prediction evaluation. Then, the influence of parameter variation is analyzed. At last, the reasonability of Gaussian mixture distribution used in our model is discussed.

1) Performance of Prediction: We have conducted experiments on both DSW and DST to evaluate the effectiveness of our proposed models.

As shown in Tables III and IV, the UTP + G + En has the highest $Precision$ and F_1 scores than the other methods. However, since UTP + En predicts the time points when topics have a bit rising trends, it may predict more candidate time points than UTP. As a result, the $Precision$ of UTP + En is lower than that of UTP + G + En, and the $Recall$ of UTP + En is higher than that of UTP + G + En. However, the F_1 score of UTP + En is lower than that of UTP + G + En. Furthermore, $Time\,cost$ of UTP + G + En gets the highest, since it is a comprehensive method which is based on EMG and Enhancement.

We can obtain more information from both Table III and Table IV. The performance of UTP is better than E-UTP and I-UTP, since more influence factors are considered. Moreover, G-Step has an improvement on $Precision$, and Enhancement has another improvement on $Recall$. Thus, the best performance is obtained by UTP + G + En, which is based on EMG and Enhancement.

We use the most general method to compute the value of F_1 which is the most persuasive metric to evaluate the prediction result. In practical applications, if the requirement is higher precision, we can choose the UTP + G + En model. On the other hand, if the requirement is higher recall, we can choose the UTP + En model.

2) Parameter Analysis: Here, we analyze the parameters F, K, s and I which may influence the performance of our proposed model.

As we know, the value of F and K may affect the performance of our proposed model, and thus the goal of this part is to find the relation between parameter values and the metric performance. As shown in Figs. 4 and 5, increasing the value of F and K at first improves the performance of our model, and then achieves the peak at $F = 2$ and $K = 2$. However, the performance decreases after $F = 2$ and $K = 2$ due to the over-fitting. It is worth noting that increasing the value of F and K may also lead to a sharp growth of the time cost since it needs to compute more probability values.

As shown in Figs. 4 and 5, we present the performance of our proposed method (UTP + G + En) in DST and DSW when $F \neq K$, ($K = 2, Z = 7, F = 1, 2, 4, 6, 8, 10$; $F = 2, Z = 7, K = 1, 2, 4, 6, 8, 10$). In order to find the appropriate value of I, we use the "Control variable method". As shown in Figs. 4 and 5, our proposed method gets the best performance when $F = K = 2$.

Next, we evaluate the influence of the hidden variable s when users post messages. At different time points, people may be affected by varying degrees of users' interests or unexpected events, and thus the value of $\tau^{u,t}$ is changing over time. As illustrated in Fig. 6, we depict the percentage distributions of influence probabilities of both the interests and the unexpected events across all users at different time points on DSW. Fig. 6(a) shows the cumulative percentage distribution of the interest influence probabilities at different time points, and the cumulative percentage distribution of the unexpected events influence probabilities is also shown in Fig. 6(b). Specifically, for a given value of τ on the x-axis, the z-axis shows the percentage of users with interests or events influence probabilities less than τ, and the y-axis shows the time points. Furthermore, if $\tau \in [0, 0.5)$, the unexpected events play an important role when users post messages. However, if $\tau \in (0.5, 1]$, the behaviors of users posting messages are much influenced by the users' interests. As we can see from Fig. 6(a), at time point 0, very few unexpected events happen, and 2% of users' τ value is no more than 0.6. In another word, at time point 0, 98% of users' interests influence probability is more than 0.6, which indicates that people are more likely to post messages based on their interests. Furthermore, in Fig. 6(a), at time point 16, many unexpected events spark widespread discussion, and the τ value of 64% users is less than 0.4, which indicates that most of the people are influenced by the unexpected events when the events happen. From the discussion above, we can conclude that people always post messages which come from their interests when the unexpected events do not happen. However, when the events happened, people can also be influenced by them.

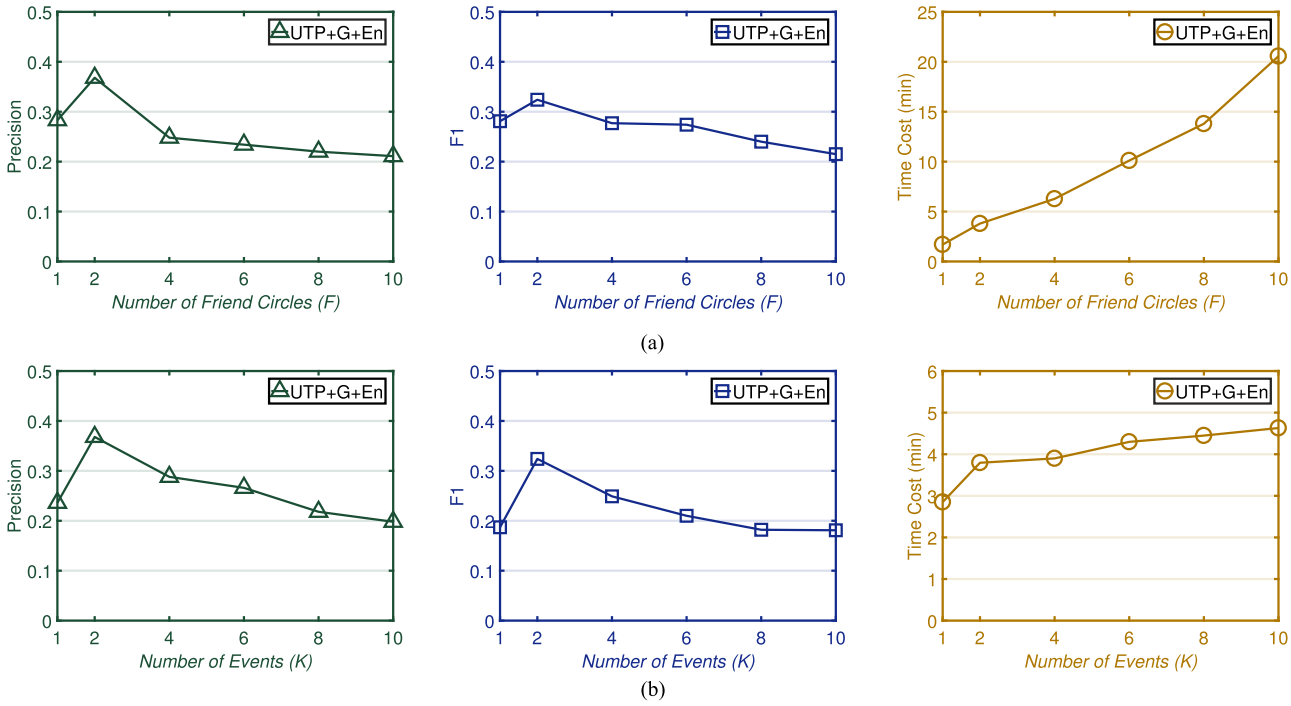


Fig. 4. Performance of varying number of F and K @DSW. (a) Performance of varying number of F ($K = 2, Z = 7$). (b) Performance of varying value of K ($F = 2, Z = 7$).

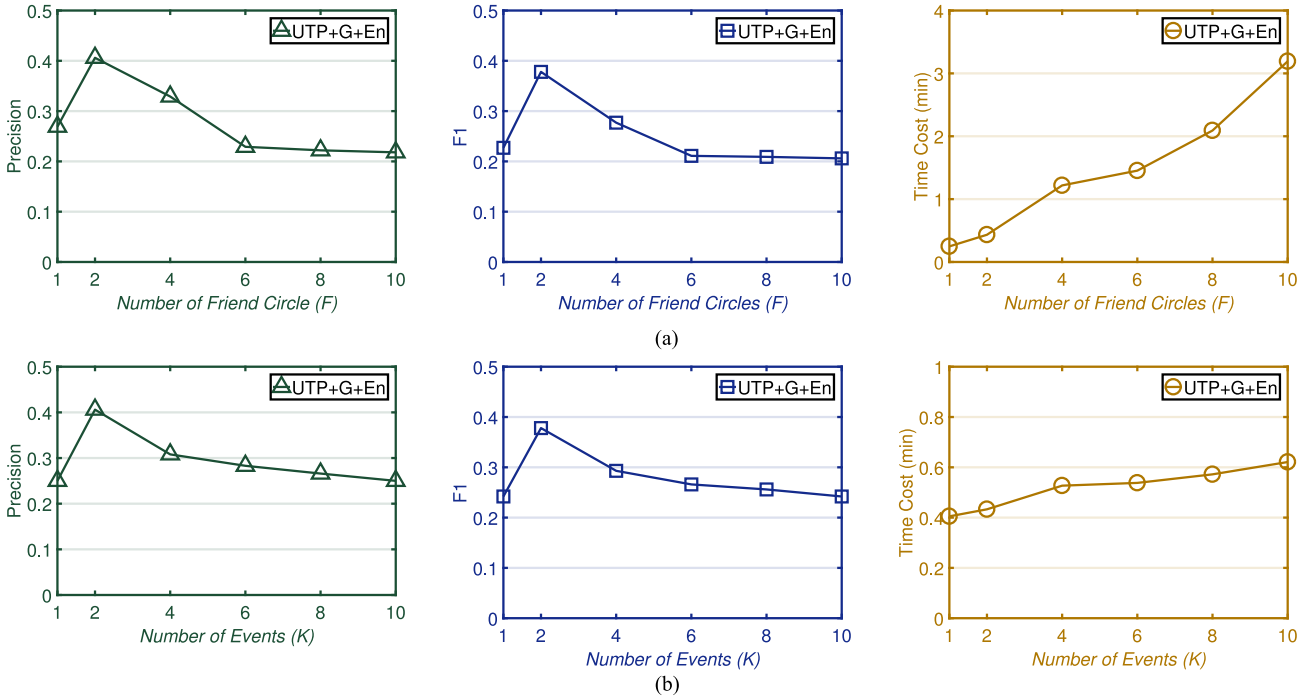


Fig. 5. Performance of varying value of F and K @DST. (a) Performance of varying number of F ($K = 2, Z = 7$). (b) Performance of varying number of K ($F = 2, Z = 7$)

At last, we estimate the performance of UTP + G + En and UTP + G when adjusting the complexity parameter I. As shown in Figs. 7 and 8, increasing I firstly improves the performance of both UTP + G + En and UTP + G, and then achieves the peak at $I = 4$. However, after that, the continuous increase of I leads to a worse performance due to the over-fitting.

We select the proper values of the parameters by conducting the above experiments. From the experimental results on DST

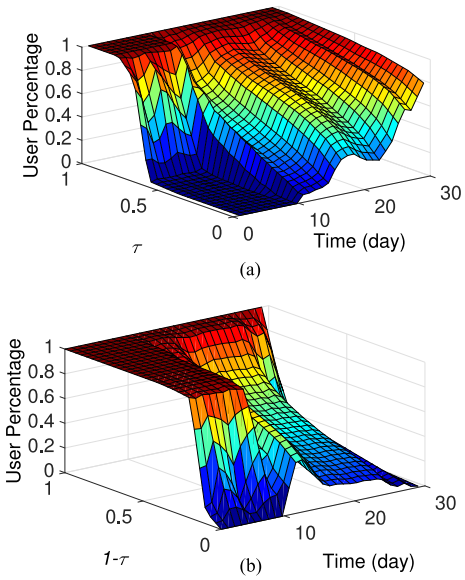


Fig. 6. Results of the influence of users' interests and unexpected events @DSW. (a) Users interests influence. (b) Unexpected events influence.

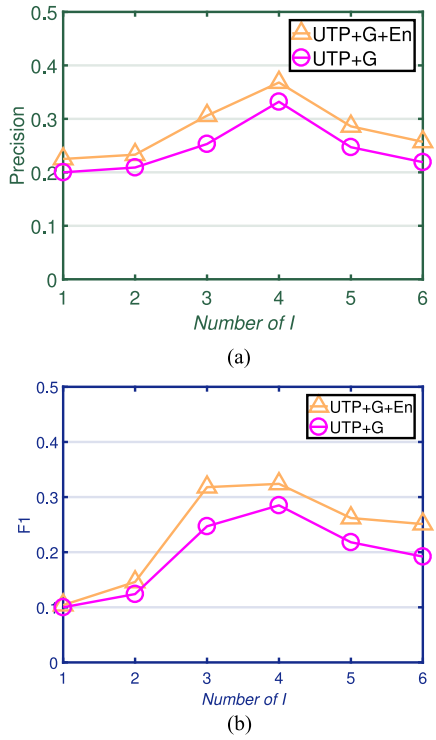


Fig. 7. Performance of varying value of I @DSW ($K = 2$, $F = 2$, $Z = 7$). (a) *Precision*. (b) F_1.

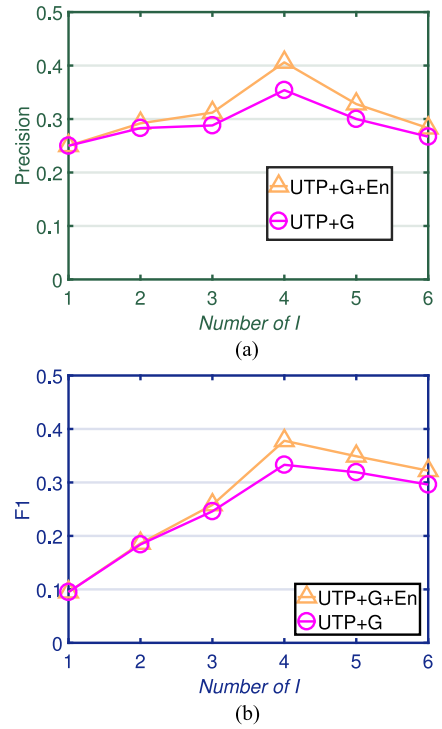


Fig. 8. Performance of varying value of I @DST ($K = 2$, $F = 2$, $Z = 7$). (a) *Precision*. (b) F_1.

TABLE V
THE PERFORMANCE OF THE ALGORITHM WITH DIFFERENT VALUES OF n ON DST

	10%			20%			30%		
	Precision	*Recall*	F_1	*Precision*	*Recall*	F_1	*Precision*	*Recall*	F_1
E-UTP	0.304	0.322	0.313	0.300	0.180	0.225	0.303	0.067	0.112
I-UTP	0.287	0.313	0.299	0.278	0.167	0.209	0.267	0.053	0.088
UTP	0.340	0.323	0.331	0.333	0.200	0.250	0.310	0.100	0.151
UTP + G	0.354	0.314	0.333	0.354	0.314	0.333	0.354	0.314	0.333
UTP + En	0.351	**0.406**	0.377	0.344	0.233	0.278	0.340	0.107	0.163
UTP + G + En	**0.406**	0.353	**0.378**	**0.406**	**0.353**	**0.378**	**0.406**	**0.353**	**0.378**

and DSW in Figs. 4, 5, 7, and 8, we can see that when $F = K = 2$, $Z = 7$, $I = 4$, our proposed method can get the best performance.

In the following, we analyze the performance of the prediction models with different values of threshold n on the DST data set. As shown in Table V, with the increasing value of n, the number of predicted time points satisfying that the number of participants gets $n\%$ higher than the previous one decreases. For prediction models which are not based on the probability density distribution, i.e., E-UTP, I-UTP, UTP, and UTP + En, the numbers of predicted time points are decreasing and those of the $\#hit$ time points are also decreasing. However, the number of re-hot time points which are in the ground truth are not changing. Therefore, with the increasing value of n, the Precision value decreases slightly due to the reducing of both the number of predicted time points and that of the $\#hit$ time points, the Recall value decreases sharply due to the reducing of $\#hit$ time pints only, and the F_1 score decreases sharply due to the sharply reducing of Recall. For the prediction models which are based on the probability density distribution, i.e., UTP + G and UTP + G + En, the changing size of n will not affect these models. Thus,

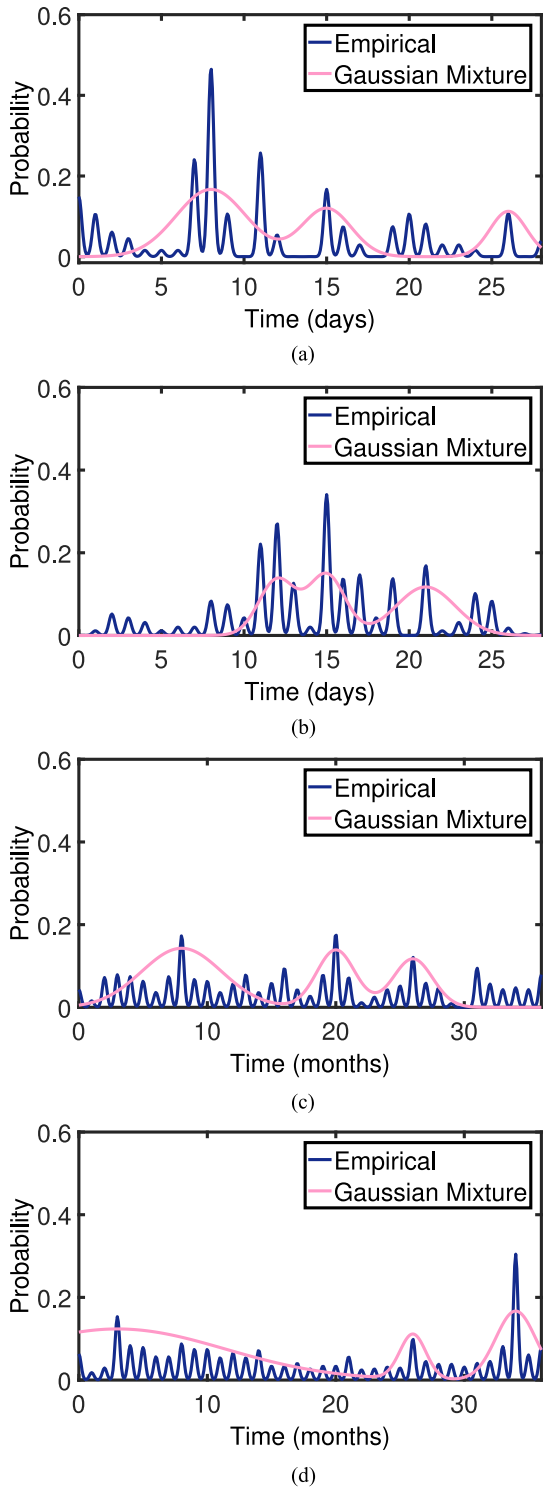


Fig. 9. Probability density functions for the four different topics. (a) The Situation of Egypt. (b) US Midterm Election. (c) The Big Bang Theory. (d) iPad.

the Prediction, Recall and F_1 scores do not change. Besides, as shown in Table V, it reveals the robustness of the G-Step for topic re-hotting prediction.

3) Change of Topics: In this part, we evaluate the reasonability of Gaussian mixture distribution used in Section V-A. It is worth noting that the changing trends of topics in DST and DSW are similar with each other, and then we just plot four of the topics as the representative.

The empirical curve is determined by the following steps. Firstly, given a topic, we compute the discrete probability value for each time point based on the statistical method, according to the ground-truth in the data set. Second, these discrete probabilities of the topic can be used to acquire their probability density function by computing the sum of Dirac functions $\delta(x)$ [37]. We use the Gaussian function to define the approximation to the function $\delta(x)$. Third, we plot the generated probability density function in Fig. 9. Because we convert the discrete probability value to the corresponding continuous probability density function, the result of empirical curve in Fig. 9 is a smoothed plot.

As shown in Fig. 9, "The Situation of Egypt" and "US Midterm Election" come from the Weibo data set (DSW), while "The Big Bang Theory" and "iPad" come from the Twitter data set (DST). There are three observations can be drawn from these plots. First, the changes of the topic trends are fluctuant. Second, Gaussian mixture distribution can better fit the changes of topic trends. Thirdly, as shown in Fig. 9(d), there is no need to predict the re-hotting time point which is between 5 and 30. Although the topic ascends several times when the time points are between 5 and 30, these time points only rise on a small scale and almost have no help to re-hot a topic. Above all, the Gaussian mixture distribution used in Section V-A can better fit the changes of the topic trends.

C. Case Study

In this subsection, we show the effectiveness of our topic re-hotting approach through a case study on the data sets DST and DST2. For convenience, the UTP + G + En method is used, and the topics "iPad" extracted from the data set DST as well as "G20" from DST2 are considered as the examples. The iPad, first released in 2010, is a kind of tablet computer produced by Apple Corp. The emergence of iPad was an innovation and achieved great success. The iPad had been to the fourth generation in 2012. Moreover, each time when there was a new product announcement or iPad was on sale, it attracted a lot of attention in OSNs.

Fig. 10 visualizes the result of our prediction on "iPad", i.e., the probability density distribution of "iPad" over time. From Fig. 10, we know that the complexity parameter I is set to 3 (Fig. 10(a)) and 4 (Fig. 10(b)) respectively. As shown in Fig. 10(b), the Gaussian Mixture is composed of G_1, G_2, G_3, and G_4. Furthermore, G_3 and G_4 get their maximum values at $t = 26$ (Mar. 2012) and $t = 34$ (Nov. 2012), respectively. It means that our model predicted the "iPad" would ascend at $t = 26$ and $t = 34$, i.e., the suitable time points for re-hotting. The fact is that, when $t = 26$, the Apple Corp held a new product announcement for the third generation of iPad in the USA Yerba Buena Center, and when $t = 34$, the fourth generation of iPad was announced. Therefore, this case study verifies the correctness and the effectiveness of our proposed model. Besides that, as shown in Fig. 10(a), we also set a smaller value of I

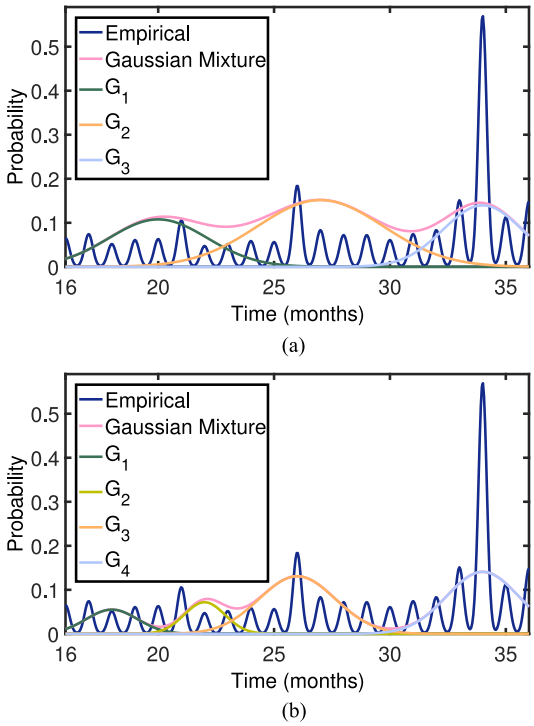


Fig. 10. The probability density distribution of "iPad" over time. (a) $I = 3$. (b) $I = 4$.

to 3. Since $I = 4$ reveals more details of topic trend changes, the performance of $I = 4$ is better than that of $I = 3$. However, increasing the value of I does not always lead to a good performance due to the over-fitting.

As we can see, Fig. 11 shows the result of the prediction on topic "G20" (Group of 20), i.e., the probability density distribution of "G20" over time. In this experiment, we use the data before April 11 as the training data, and use the rest as the testing data. From Fig. 11, we can know that the parameter I is set to 2 (Fig. 11(a)), 3 (Fig. 11(b)), and 4 (Fig. 11(c)) in this part, respectively. As shown in Fig. 11(b), the Gaussian Mixture is composed of G_1, G_2 and G_3. Furthermore, G_2 and G_3 get their maximum value at Aug. 10–28 (we set $t = 23$) and Sep. 18–27 (we set $t = 26$) respectively, which indicates that our model predicted the hotness of "G20" would ascend at $t = 23$ and $t = 26$, i.e., the appropriate time points for topic re-hotting. Actually, the fact is that at $t = 23$ time point, the United States issued a series of economic programs to promote the economic growth, which is in preparation for the G20 summit, and at time point 26, the 12th G20 summit was held in Pittsburgh, USA. As different values of I may influence the performance of our model, we set three different values of I to find the most appropriate one. As shown in Fig. 11, when the value of I is very small, i.e., $I = 2$, the number of the mixture of Gaussian is also very small, which cannot reflect much details of the changes of the topic and then leads to an inaccurate result. However, when the value of I is large ($I = 4$), it will also lead to a worse performance due to the over-fitting. For the larger data set DST2, when I is 3, we can obtain the best prediction result.

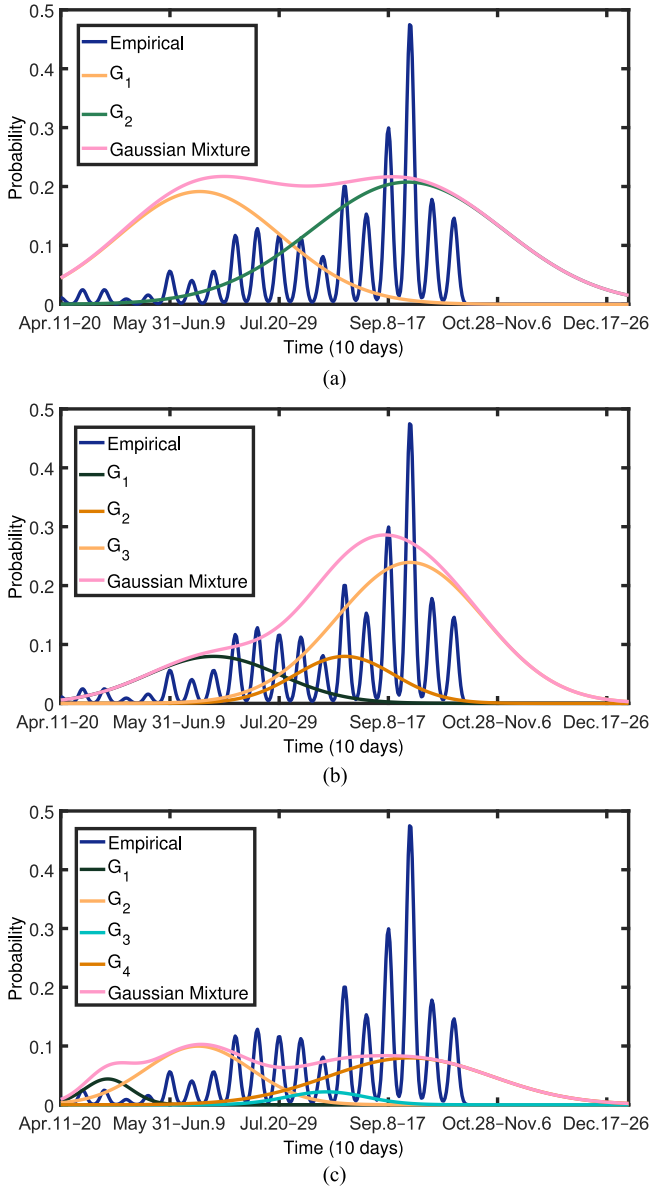


Fig. 11. The probability density distribution of "G20" over time. (a) $I = 2$. (b) $I = 3$. (c) $I = 4$.

VII. Conclusion

This paper proposes a temporal UTP model to solve the challenging problem of topic re-hotting prediction in OSNs. By taking into account three factors, i.e., users' friend-circles, types of topics, and unexpected events, UTP combines users' interests (I-UTP) and unexpected events (E-UTP). Furthermore, we propose the EMG algorithm for model inference and a prediction method to predict the re-hotting time points accurately. Moreover, in order to reduce the influence of slight fluctuations of topics, a weighting scheme is proposed. Finally, we demonstrate the performance of the proposed methods on three real-world data sets, and analyze the interesting phenomena which appear in our experiments. In the future, for predicting the re-hotting time points more accurately, some data preprocessing methods can be used to reduce the noise in OSN data.

APPENDIX A
A. PROOF OF THEOREM 1

In this appendix, we prove the convergence of the proposed EMG algorithm. EMG is composed of E-Step, M-Step, and G-Step. Since EM algorithm has been proved to be convergent by many researchers [38]–[41], we just prove the convergence of G-Step here.

As we know, the data which is generated by the EM algorithm follows the Gaussian mixture distribution. We define $f_v(t;\mathbf{\Psi}^*) = a_i \cdot \exp\{-\frac{(t-\mu_i)^2}{2\sigma_i^2}\}$. Then, the log-likelihood in G-Step is as follows:

$$L(t;\mathbf{\Psi}^*) = \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} f_v(t;\mathbf{\Psi}^*)\right). \tag{30}$$

Thus, our goal is to prove that $L(t;\mathbf{\Psi}^{*(k)})$ can be convergent with the increase of the number of iterations.

The boundedness of the log-likelihood:

Because $L(t;\mathbf{\Psi}^{*(k)})$ is a continuously differentiable function on $\mathbf{\Psi}^{*(k)}$ and $\{\mathbf{\Psi}^{*(k)}\}$ is bounded, the limit value of $L(t;\mathbf{\Psi}^{*(k)})$ is presented, i.e., $\lim_{k\to\infty} L(t;\mathbf{\Psi}^{*(k)}) = L^*$. Therefore, we know the following relationship:

$$L(t;\mathbf{\Psi}^{*(0)}) \le L(t;\mathbf{\Psi}^{*(k)}) \le L^* \tag{31}$$

where $L(t;\mathbf{\Psi}^{*(0)})$ is the initial value of $L(t;\mathbf{\Psi}^{*(k)})$. From the above, the $L(t;\mathbf{\Psi}^{*(k)})$ is bounded.

The monotonicity of the log-likelihood:

As we know

$$\begin{aligned} L(t;\mathbf{\Psi}^{*(k)}) &= \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} f_v(t;\mathbf{\Psi}^{*(k)})\right) \\ &= \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}\right) \end{aligned} \tag{32}$$

where $\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}$ is the expectation of $\frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}$. Furthermore, based on the Jensen inequality on concave functions, we get the following relationship:

$$F\left(\left(E_{\mathbf{\Psi}^{*(k)}}\left[\frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}\right]\right)\right) \ge E_{\mathbf{\Psi}^{*(k)}}\left[F\left(\frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}\right)\right]. \tag{33}$$

Thus, the log-likelihood has the inequation as follows:

$$\begin{aligned} L(t;\mathbf{\Psi}^{*(k)}) &= \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} f_v(t;\mathbf{\Psi}^{*(k)})\right) \\ &= \sum_{m=1}^{M} \log\left(\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}\right) \\ &\ge \sum_{m=1}^{M}\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \log\frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})}. \end{aligned} \tag{34}$$

From Equation (34), we know that we only need to maximize the lower bound of the log-likelihood, and then the following relationship is presented:

$$\begin{aligned} L(t;\mathbf{\Psi}^{*(k+1)}) &\ge \sum_{m=1}^{M}\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \log\frac{f_v(t;\mathbf{\Psi}^{*(k+1)})}{q(t;\mathbf{\Psi}^{*(k)})} \\ &\ge \sum_{m=1}^{M}\sum_{i=1}^{I} q(t;\mathbf{\Psi}^{*(k)}) \log\frac{f_v(t;\mathbf{\Psi}^{*(k)})}{q(t;\mathbf{\Psi}^{*(k)})} \\ &= L(t;\mathbf{\Psi}^{*(k)}). \end{aligned} \tag{35}$$

From Equation (35), we can conclude that $L(t;\mathbf{\Psi}^*)$ is monotonically increasing, i.e., $L(t;\mathbf{\Psi}^{*(k+1)}) \ge L(t;\mathbf{\Psi}^{*(k)})$.

Comprehensively considering the boundedness and the monotonicity of the log-likelihood, we proved the convergence of the G-Step. Therefore, the EMG algorithm is proved to be convergent.

REFERENCES

[1] Y. Chen, H. Amiri, Z. Li, and T.-S. Chua, "Emerging topic detection for organizations from Microblogs," in *Proc. 36th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2013, pp. 43–52.

[2] J. Chen, J. Yu, and Y. Shen, "Towards topic trend prediction on a topic evolution model with social connection," in *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. Agent Technol.*, 2012, pp. 153–157.

[3] Y. Wang, E. Agichtein, and M. Benzi, "TM-LDA: Efficient online modeling of latent topic transitions in social media," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Min.*, 2012, pp. 123–131.

[4] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proc. 19th Int. Conf. World Wide Web*, 2010, pp. 591–600.

[5] T. Hofmann, "Probabilistic latent semantic indexing," in *Proc. 22nd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 1999, pp. 50–57.

[6] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *J. Amer. Soc. Inf. Sci.*, vol. 41, no. 6, pp. 391–407, 1990.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, 2003.

[8] W. Li and A. McCallum, "Pachinko Allocation: DAG-structured mixture models of topic correlations," in *Proc. Int. Conf. Mach. Learn.*, 2006, pp. 577–584.

[9] K. Liu, J. Xu, L. Zhang, Z. Ding, and M. Li, "Discovering hot topics from Geo-tagged video," *Neurocomputing*, vol. 105, pp. 90–99, 2013.

[10] X. Wang, L. Qi, C. Chen, J. Tang, and M. Jiang, "Grey system theory based prediction for topic trend on internet," *Eng. Appl. Artif. Intell.*, vol. 29, pp. 191–200, 2014.

[11] P. Velardi, G. Stilo, A. E. Tozzi, and F. Gesualdo, "Twitter mining for fine-grained syndromic surveillance," *Artif. Intell. Med.*, vol. 61, pp. 153–163, 2014.

[12] X. Zhang, X. Chen, Y. Chen, S. Wang, Z. Li, and J. Xia, "Event detection and popularity prediction in microblogging," *Neurocomputing*, vol. 149, pp. 1469–1480, 2015.

[13] G. Stilo and P. Velardi, "Efficient temporal mining of micro-blog texts and its application to event discovery," *Data Min. Knowl. Discovery*, vol. 30, no. 2, pp. 372–402, 2016.

[14] G. Stilo and P. Velardi, "Temporal semantics: Time-varying Hashtag sense clustering," in *Proc. Int. Conf. Knowl. Eng. Knowl. Manage.*, 2014, pp. 563–578.

[15] G. Stilo and P. Velardi, "Time makes sense: Event discovery in Twitter using temporal similarity," in *Proc. IEEE/WIC/ACM Int. Joint Conf. Web Intell. Agent Technol.*, 2014, vol. 2, pp. 186–193.

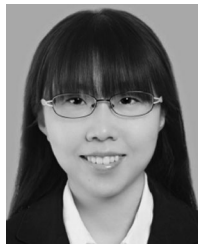
[16] H. Yin, B. Cui, L. Chen, Z. Hu, and Z. Huang, "A temporal context-aware model for user behavior modeling in social media systems," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2014, pp. 50–57.

[17] X. Song, Q. Zhang, Y. Sekimoto, and R. Shibasaki, "Prediction of human emergency behavior and their mobility following large-scale disaster," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Min.*, 2014, pp. 5–14.

[18] J. Zhang, C. Wang, J. Wang, and J. X. Yu, "Inferring continuous dynamic social influence and personal preference for temporal behavior prediction," *Proc. VLDB Endowment*, vol. 8, no. 3, pp. 269–280, 2014.

[19] P. Awasthi, M. Charikar, R. Krishnaswamy, and A. K. Sinop, "Spectral embedding of k-cliques, graph partitioning and k-means," in *Proc. ACM Conf. Innov. Theo. Comput. Sci.*, 2016, pp. 301–310.
[20] A. P. Dempster, N. M. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Statist. Soc.*, vol. 39, no. 1, pp. 1–38, 1977.
[21] Y. Yang, H. K. T. Ng, and N. Balakrishnan, "A stochastic expectation-maximization algorithm for the analysis of system lifetime data with known signature," *Comput. Statist.*, vol. 31, no. 2, pp. 609–641, 2016.
[22] C. Santiago, J. C. Nascimento, and J. S. Marques, "A robust active ahape model using an expectation-maximization framework," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 6076–6080.
[23] S. Oyama, Y. Baba, Y. Sakurai, and H. Kashima, "EM-based inference of true labels using confidence judgments," in *Proc. 1st AAAI Conf. Human Comput. Crowdsourcing*, 2013, pp. 58–59.
[24] J. Stückler and B. Sven, "Efficient dense 3D rigid-body motion segmentation in RGB-D video," in *Proc. 24th Brit. Mach. Vis. Conf.*, 2013, pp. 51.1–51.11. [Online]. Available: http://www.bmva.org/bmvc/2013/Papers/paper0051/index.html
[25] B. Quost and T. Denœux, "Clustering and classification of fuzzy data using the fuzzy EM algorithm," *J. Roy. Statist. Soc.*, vol. 286, no. 1, pp. 134–156, 2016.
[26] S. Wang, Z. Yan, X. Hu, P. S. Yu, and Z. Li, "Burst time prediction in cascades," in *Proc. 29th AAAI Conf. Artif. Intell.*, 2015, pp. 325–331.
[27] S. Wang, Z. Yan, X. Hu, P. S. Yu, Z. Li, and B. Wang, "CPB: A classification-based approach for burst time prediction in cascades," *Knowl. Inf. Syst.*, vol. 49, no. 1, pp. 243–271, 2016.
[28] X. Xin, C. Wang, X. Ying, and B. Wang, "Deep community detection in topologically incomplete networks," *Physica A, Statist. Mech. Appl.*, vol. 469, pp. 342–352, 2017.
[29] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval: The Concepts and Technology Behind Search*, 2nd ed. New York, NY, USA: ACM press, 1999.
[30] Y. Mylonas, M. Lestas, A. Pitsillides, P. Ioannou, and V. Papadopoulou, "Speed adaptive probabilistic flooding for vehicular ad hoc networks," *IEEE Trans. Vehicular Technol.*, vol. 64, no. 5, pp. 1973–1990, May 2015.
[31] Y. Wand and R. Weber, "An ontological model of an information system," *IEEE Trans. Softw. Eng.*, vol. 16, no. 11, pp. 1282–1292, Nov. 1990.
[32] A. Skrondal and S. Rabe-Hesketh , "Latent variable modelling: A survey," *Scand. J. Stat.*, vol. 34, no. 4, pp. 712–745, 2007.
[33] M. Wang, C. Wang, J. X. Yu, and J. Zhang, "Community detection in social networks: An in-depth benchmarking study with a procedure-oriented framework," *Proc. VLDB Endowment*, vol. 8, no. 10, pp. 998–1009, Jun. 2015.
[34] J. Ma, M. Sun, S. Yin, and C. Han, "The elements and formation mechanism of micro-blog information ecological chain," *Library Inf. Service*, vol. 56, pp. 73–77, 2012.
[35] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang, "Geographical topic discovery and comparison," in *Proc. 20th Int. Conf. World Wide Web*, 2011, pp. 247–256.
[36] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 113–120.
[37] P. A. M. Dirac, *The Principles of Quantum Mechanics*, 4th ed. London, U.K.: Oxford Univ. Press, 1981.
[38] C. F. J. Wu, "On the convergence properties of the EM algorithm," *Ann. Statist.*, vol. 11, pp. 95–103, 1983.
[39] D. Nettleton, "Convergence properties of the EM algorithm in constrained parameter spaces," *Can. J. Statist.*, vol. 27, no. 3, pp. 639–648, 1999.
[40] K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *J. Comput. Assisted Tomography*, vol. 8, no. 2, pp. 306–316, 1984.
[41] A. N. Iusem, "A short convergence proof of the EM algorithm for a specific poisson model," *Brazilian J. Probab. Statist.*, vol. 6, no. 1, pp. 57–67, 1992.

Chaokun Wang received the B.Eng. degree in computer science and technology, the M.Sc. degree in computational mathematics, and the Ph.D. degree in computer software and theory, in 1997, 2000, and 2005, respectively, from Harbin Institute of Technology, Harbin, China. He joined the faculty of School of Software at Tsinghua University in February 2006, where he is currently an Associate Professor. He has published more than 70 refereed papers, got two Best Paper Awards at international conferences, and holds 12 patents. His current research interest includes social network analysis, graph data management, and music computing.



Xin Xin received the B.Eng. degree in software engineering from Northeastern University, Shenyang, China, and the M.Eng. degree in software engineering from Tsinghua University, Beijing, China, in 2013 and 2016, respectively. She is currently with the China Gold Coin Corporation. Her research interests include social network analysis, data mining, community detection, and deep learning.

Jingwen Shang received the B.Eng. degree in information security from Northeastern University, Shenyang, China, in 2015. She is currently working toward the M.Eng. degree in the School of Software, Tsinghua University, Beijing, China. Her research interests include social network analysis and data mining.