A Unified Model for Stable and Temporal Topic Detection from Social Media Data

Hongzhi Yin† Bin Cui† Hua Lu‡ Yuxin Huang† Junjie Yao†
†*Department of Computer Science and Technology*
Key Laboratory of High Confidence Software Technologies, Peking University
‡*Department of Computer Science, Aalborg University*
†{bestzhi, bin.cui, huangyuxin, junjie.yao}@pku.edu.cn, ‡luhua@cs.aau.dk

***Abstract*— Web 2.0 users generate and spread huge amounts of messages in online social media. Such user-generated contents are mixture of temporal topics (e.g., breaking events) and stable topics (e.g., user interests). Due to their different natures, it is important and useful to distinguish temporal topics from stable topics in social media. However, such a discrimination is very challenging because the user-generated texts in social media are very short in length and thus lack useful linguistic features for precise analysis using traditional approaches.**

In this paper, we propose a novel solution to detect both stable and temporal topics simultaneously from social media data. Specifically, a unified user-temporal mixture model is proposed to distinguish temporal topics from stable topics. To improve this model's performance, we design a regularization framework that exploits prior spatial information in a social network, as well as a burst-weighted smoothing scheme that exploits temporal prior information in the time dimension. We conduct extensive experiments to evaluate our proposal on two real data sets obtained from Del.icio.us and Twitter. The experimental results verify that our mixture model is able to distinguish temporal topics from stable topics in a single detection process. Our mixture model enhanced with the spatial regularization and the burst-weighted smoothing scheme significantly outperforms competitor approaches, in terms of topic detection accuracy and discrimination in stable and temporal topics.

I. Introduction

User-generated contents (UGC) in Web 2.0 are valuable resources capturing people's interests, thoughts and actions. Such contents cover a wide variety of topics that present online and offline lives. For example, the microblog services gather many short but quickly-updated texts that contain both temporal and stable topics. Such topics form a huge and rich repository of various kinds of interesting information.

Stable topics are often on users' regular interests and their daily routine discussions, which usually evolve at a rather slow speed. The extraction of such stable topics enables us to personalize the results and to improve the result relevance in many applications such as computational advertising, content targeting, personal recommendation and web search.

In contrast, temporal topics are on popular real-life events or hot spots. In many circumstances, temporal topics, e.g., breaking events in the real world, bring about popular discussion and wide diffusion on the Internet, where social networks further boost the discussion and diffusion. Take Twitter, the most popular microblog service, as an example. Many social events can be discovered in Twitter's posts (tweets), such as emergencies (e.g., earthquakes), politics (e.g., elections), public events (e.g., Olympics), and business (e.g., the release of a new smartphone).

An example of stable and temporal topics from Twitter is illustrated in Figure 1. We can tell the difference between them from the temporal distributions and the description keywords. A temporal topic has its text related to a certain event like "Independence Day celebration" in a certain period of time, and its popularity goes through a sharp increase at the occurring time of the event. A stable topic has its description on user's regular interest like "Pet Adoption" and its temporal distribution exhibits no sharp, spike-like fluctuation.

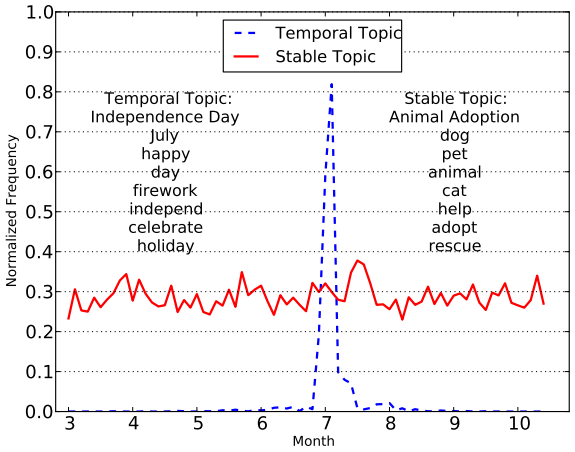


Fig. 1. Stable and Temporal Topics in Twitter

It is important and useful to distinguish the temporal topics from the stable topics since they convey different kinds of information. However, temporal topics are discussed with less urgent themes in the background, and therefore temporal topics are deeply mixed with stable topics in social media. As a result, it is a challenging problem to detect and differentiate temporal and stable topics from large amounts of user-generated social media data.

Research on traditional topic detection and tracking employs on-line incremental clustering [1] or retrospective off-line clustering [25] for documents and extracts representative features for clusters as a summary of the events. These methods are suitable for conventional web pages where most documents are long, rich in keywords, and related to certain popular events.

978-1-4673-4910-9/13/$31.00 © 2013 IEEE

However, social media data differ substantially from web pages. E.g., in collaborative tagging systems, description of resource is constrained to a few words. In microblog systems like Twitter, the length of each post cannot exceed 140 characters. Statistics show that each post covers only 12.79 words on average in the real data sets used in our experiments. User-generated contents with short textual descriptions result in very sparse keyword-document distributions, rendering traditional topic detection methods unsuitable for social media data.

Other methods like topic models and burst feature clustering have also been applied to detect topics with temporal features. However, these methods lack explicit combination of stable and temporal features. Hence, temporal topics detected by them would be overwhelmed by stable keywords with abstract semantics, especially on data sets where stable topics occupy advantaged proportions. This shortcoming will be demonstrated in our experiments in Section VI.

In this paper, we aim to detect temporal topics and extract stable topics simultaneously in a unified mixture model. Specifically, we propose a user-temporal mixture model based on the intuition that the content generated by a user is based on either his/her personal stable interest or a popular topic that attracts his/her attention at that time.

Stable topics represent commonly discussed themes within steady user groups. Therefore, we assume that stable topics are generated by users or user communities. Accordingly, we design a spatial regularizer that makes use of the social network structure to enhance the mixture model.

On the other hand, temporal topics have their time-sensitive popularity, i.e., their trends go through a drastic increase in a certain period of time, resulting in a synchronous increase of correlated words occurrence. This can be seen from the temporal topic shown in Figure 1. Aware of this important observation, we design a temporal regularizer and a burst-weighted smoothing scheme to further enhance the mixture model for simultaneous temporal and stable topic detection.

We make the following contributions in this paper.

- We propose a user-temporal mixture topic model which combines stable and temporal features in topic detection. To the best of our knowledge, this is the first mixture model that allows simultaneous detection of stable and temporal topics in a direct way.
- We propose a regularization framework to enhance our user-temporal mixture model by exploiting both spatial and temporal information.
- We propose a novel burst-weighted smoothing scheme to improve the process of temporal topic detection and presentation.
- We conduct extensive experiments to evaluate our proposals. The experimental results demonstrate the superiority of our model in terms of topic detection accuracy and discrimination in stable and temporal topics.

The rest of the paper is organized as follows. Section II briefly reviews related works. Section III presents preliminaries and formulates the problem to tackle. Section IV details the basic user-temporal mixture model for simultaneous detection of stable and temporal topics. Section V elaborates on the enhancements for the basic mixture model. Section VI reports the experimental results. Section VII concludes the paper.

II. Related Works

Topic models provide a principled and nice way to discover topic structures from large document collections. Standard topic models such as LDA [4] and PLSA [10], [11] do not consider the temporal information. A number of temporal topic models have been proposed to consider topic evolution and topic trends over time. Mei and Zhai [19], [24] studied mining evolutionary topics from texts by comparing topics in consecutive time intervals. Wang and McCallum [23] designed Topic Over Time (TOT) model that treats time stamp as an observed continuous variable generated by topic. This model was designed for capturing temporal features with beta distribution, and confined topics neatly focused on time by keeping the temporal components. Some other models [2], [3], [8] are also proposed to study topic changes over time, but the focus is not on bursty patterns.

Hong and Davison [12] applied author-topic model on Twitter system, and made use of detected topics for information retrieval. Lin et al. [17] designed a model that combines document stream and network stream into popular event tracking in social media. This model also employs a Markov chain to model topic evolution along the time. Hong et al. [13] proposed to merge volume , as keywords' appearance in periods, into Dynamic Topic Model. We point out that these proposals take temporal features into consideration by concerning evolution of topics rather than catching burst spikes.

Detecting and presenting temporal features are critical for burst detection. Kleinberg [15] invented a 2-state HMM based burst detection algorithm for streams. Zhao et al. [28] applied graph cut algorithm on information flow graph to detect temporal events. Chieu and Lee [5] built a system for extracting events based on query. Leskovec et al. [16] proposed to cluster text phrase graphs to detect events in news articles. Pui et al. [20] proposed to extract burst features from text stream and group them into burst events with a Baysian probabilistic model. Yao et al. [26], [27] designed sliding window and graph clustering based approach for burst event detection in tags. Prasad et al. [14] applied document clustering to only the novel documents identified by the dictionary learning technique within each time slice, and subsequently their detected emerging topics consist of novel documents. Unlike these works focused on detecting individual bursty keywords or novel documents, this paper is intended to detecting interesting topics which are directly associated with a word distribution.

Considering that posts are short and noisy in microblogs, Zhao et al. [29] propose a Twitter-LDA model, which only considers the users' personal interests but not the global topic trends. The TimeUserLDA model proposed by Diao et al [7] is most relevant to our proposed basic model in this paper, which is designed for finding bursty topics from microblogs. To detect bursty topics, a state machine-based method must

be applied to the topics discovery by TimeUserLDA. But the model can not be directly used to distinguish temporal topics from stable topics. Besides, the social aspect is ignored in the model although the social information is so important to improve the process of topic discovery for the social media data [18]. Moreover, the model does not consider the burst information of words which is most helpful to detect bursty/temporal topics, as is evidenced in our experiment.

Our work in this paper distinguishes from previous works in several points. First, this paper combines user and temporal features in one mixture model, which can be directly used to detect stable and temporal topics simultaneously. In contrast, previous works do not provide such a mixture model or unified detection of stable and temporal topics in a direct way. Second, this paper takes advantage of user posts' temporal information plus individual bursty keywords to mine topics with temporal features. Unlike the existing burst detection methods, our proposed method automatically extracts correlated texts with temporal features and clusters them into temporal topics. Third, this paper is intended to detecting topics from social media data in which a document contains up to a hundred plus words. Whereas most previous works target at traditional, long text documents.

III. Preliminaries and Problem Formulation

In this section, we formally define the social media data that we are interested in and formulate the problems accordingly.

We first define the concept of document that captures the textual contents of social media data.

Definition 1: **(Document)** Given a vocabulary W, a document d is a bag of words from W, i.e., $d = \{w_1, w_2, \ldots, w_{|d|}\}$ where $w_i \in W$ $(1 \leq i \leq |d|)$.

This definition follows most works in information retrieval and topic modeling [2], [9], [17]–[19], [24]. Given a document $d = \{w_1, w_2, \ldots, w_{|d|}\}$, we ignore the ordering of words. For the sake of simple representation, we use $\mathcal{C}$ to denote the complete *document collection* in a social media data set.

In the context of social media, a document is generated by a user through an action that is usually called "posting".

Definition 2: **(Posting)** A posting is a triple (u, d, t) that means user u generated document d at time t.

Usually there exist huge numbers of such postings in a social media platform. In order to analyze such huge data effectively, two preprocessing steps are conducted on all postings. First, the document d in each posting is checked against a stop word list and each appearance of a stop word is eliminated from d. Without the loss of generality, we still use d to denote the document free of stop words. Second, all postings are then aggregated into a structure called user-time-keyword (hyper-)matrix that is defined below.

Definition 3: **(User-Time-Keyword Matrix)** A user-time-keyword matrix $M(U, T, W)$ is a hyper-matrix whose three dimensions refer to user, time and keyword respectively. A cell $M[u, t, w]$ in such a matrix stores the occurrence of word w in all postings that are generated by user u within time interval t.

In the aforementioned second preprocessing step, the time span of all postings is divided into time intervals each of which corresponds to a series of cells with the same user and the same word. Initially, all cells in matrix M is set to 0. Subsequently, for each posting (u, d, t), $M[u, t, w]$ is increased by 1 for each word appearance $w \in d$.

Definition 4: **(Topic)** A semantically coherent topic in a text collection $\mathcal{C}$ is represented by a topic model θ, which is a probability distribution of words $\{p(w|\theta)\}_{w \in W}$. Clearly, we have $\sum_{w \in W} p(w|\theta) = 1$.

To illustrate the content of a topic, we pick up k words with the highest appearance probabilities to represent it. In our work, we distinguish between **stable topics** and **temporal topics**. A stable topic is a summary of regular themes posted by users, which remains stable popularity over time, i.e., the trends of such topics have been stabilizing over time. A temporal topic has a clear temporal feature that the popularity of the topic presents increasing or declining trends over time and reaches its peak only during a certain period of time.

Definition 5: **(Social Network)** A social network is a graph $G = (V, E)$, where V is a set of vertices and E is a set of edges between vertices in V. Particularly, each vertex in V corresponds to a user, and an edge $e = (u, v) \in E$ stands for the social relationship between two users u and v. The strength of the tie between users u and v is defined as a normalized non-negative value $\pi(u, v)$.

Given a document collection $\mathcal{C}$, a user-time-keyword matrix M, and a social network G, our intension in this paper is to find interesting topics from $\mathcal{C}$ by exploiting the information captured in M and G. More explicitly, we intend to perform the following two tasks in a unified way.

Task 1: Extracting Stable Topics. This task is to model and extract a set of stable topic models, $\Theta_U = \{\theta_i\}$, where $|\Theta_U| = k_1$ and k_1 is a user specified parameter.

Task 2: Detecting Temporal Topics. The task is to discover and detect a set of temporal topic models, $\Theta_T = \{\theta_j\}$, where $|\Theta_T| = k_2$ and k_2 is a user specified parameter.

The result of Task 1 can be utilized to model stable interests of a user, which can bring lots of benefits to computational advertising, content targeting and web search with personalized results and improved relevance. The result of Task 2 can be used to analyze the topic trends and to identify the currently most popular and hot topics.

To resolve these two tasks in a unified way is challenging due to several reasons. First, no existing model supports both tasks at the same time. Second, there exists no framework that can simultaneously embed a social network structure and a temporal structure in a topic model. Indeed, it is an open question whether a social network structure can help extract stable topics. It is also still unclear whether temporal information can improve the process of temporal topic detection. Our research in this paper will provide answers to these questions.

It is also noteworthy that traditional topic detection and tracking (TDT) techniques, as well as topic models, do not suit the context of user-oriented social media. Such traditional techniques work on documents as basic units, assuming that

a document is relatively rich in keywords and all keywords in a document are coherent. However, user-generated contents in today's social media usually consist of very short textual descriptions, which result in very sparse keyword-document distributions. Such a substantial distinction from traditional materials for information retrieval also calls for novel techniques for topic detection from social media data [12].

IV. A Mixture Model for Detecting Stable and Temporal Topics

In this section, we propose a user-temporal mixture topic model that integrates user and temporal features, followed by an EM-based algorithm for inferring model parameters.

A. User-Temporal Model

SYMBOL	DESCRIPTION
u, t, w	user, time stamp, keyword
U, T, W	set of users, time stamps and keywords
$M[u, t, w]$	frequency of w used by u within time stamp t
λ_U, λ_T	parameter controlling the branch selection
θ_i	stable topic indexed by i
θ_j	temporal topic indexed by j
Θ_U, Θ_T	stable and temporal topic set

TABLE I
NOTATIONS

In the user-temporal mixture model, we pre-categorize topics into two types: stable topics and temporal topics. Stable topics summarize theme reflected from regular postings according to the stable interest of a user or a community. While temporal topics capture the popular events or controversial news igniting hot discussion in a certain period. In this model, we aim to detect both temporal and stable topics in one generating process. Table I lists the relevant notations we use.

The mixture model is represented in Equation 1. For a stable topic θ_i we pay particular attention to its user u who generates it. For a temporal topic θ_j we pay more attention to when, indicated by time t, it is generated. Like PLSA [10], [11], our user-temporal model consists of three layers and two branches mixing user and temporal features, each branch deciding a different topic type. Parameters λ_U and λ_T in Equation 1 are the probability coefficients controlling the branch choice, which also denote the proportions of stable and temporal topics in the data set.

$$p(w|u,t) = \lambda_U \sum_{\theta_i \in \Theta_U} p(\theta_i|u)p(w|\theta_i) + \lambda_T \sum_{\theta_j \in \Theta_T} p(\theta_j|t)p(w|\theta_j) \tag{1}$$

For the user branch, a stable topic is chosen according to the interest of a particular user. For the time branch, a temporal topic is generated according to the time stamp of a post, which means the post belongs to the topics that are popular for a short period of time around that time stamp. Temporal topics have their distribution on the time dimension, which indicates its popularity probabilities. The time period during which a temporal topic has its highest probability is its popularity period. In our setting, the user interest is assumed to be stable through time, and we ignore the possible slight evolution of user interest.

Our mixture model contains four probability matrices, namely $p(\theta_i|u)$, $p(w|\theta_i)$, $p(\theta_j|t)$ and $p(w|\theta_j)$. These parameters are estimated from the observation data, which is a user-time-keyword matrix M where each cell $M[u, t, w]$ indicates how many times user u post word w during time period t. This matrix is used in estimating both types of topics.

Whether a keyword is generated by either a stable topic or a temporal topic is decided by relevant statistics which are the contributions by the user u and the time slice t, respectively. For instance, if many other users also use keyword w in a certain period t, w would be treated as a temporal keyword with higher probability. Otherwise, it would belong to stable topics. Thus, keywords with clear temporal features would be clustered into temporal topics whose popularity synchronizes to that of their keywords.

The topics generated in the two branches are not estimated individually. Both types of topics interact with each other during the learning procedure. This two-branch assumption can filter out the stable components from burst topics by stable branch. It also helps refine the quality of stable topics without disturbance from breaking events as time elapses.

B. Estimation of Model Parameters

Given an observation matrix $M(U, T, W)$, the learning procedure of our model is to estimate the maximum probability of generating the observed samples. The log-likelihood of the whole document collection $\mathcal{C}$ by our approach is in Equation 2, where $p(w|u, t)$ is defined according to Equation 1.

$$L(\mathcal{C}) = \sum_U \sum_T \sum_W M[u, t, w] \log p(w|u, t) \tag{2}$$

The goal of parameter estimation is to maximize Equation 2. As this equation cannot be solved directly by applying Maximum Likelihood Estimation (MLE), we apply an EM approach instead. In an expectation (E) step of the EM approach, posterior probabilities are computed for the latent variables based on the current estimates of the parameters. In a maximization (M) step, parameters are updated by maximizing the so-called expected complete data log-likelihood that depends on the posterior probabilities computed in the E-step.

In our model, we have parameter set $\{p(w|\theta_i),\ p(w|\theta_j),\ p(\theta_i|u),\ p(\theta_j|t)\}$ and the latent variables are hidden topics θ_i and θ_j. For simplicity, we use ψ to denote all these parameters. The detailed parameter estimation procedure for the user-temporal model is illustrated in following equations.

E-step:

$$p(\theta_i|u, t, w) = \frac{\lambda_U p(w|\theta_i)p(\theta_i|u)}{\lambda_U B(w|u) + \lambda_T B(w|t)} \tag{3}$$

$$p(\theta_j|u, t, w) = \frac{\lambda_T p(w|\theta_j)p(\theta_j|t)}{\lambda_U B(w|u) + \lambda_T B(w|t)} \tag{4}$$

where $B(w|u)$ and $B(w|t)$ are defined as follows:

$$B(w|u) = \sum_{\theta_i \in \Theta_U} p(w|\theta_i)p(\theta_i|u)$$

$$B(w|t) = \sum_{\theta_j \in \Theta_T} p(w|\theta_j)p(\theta_j|t)$$

M-step: With simple derivations [10], we can obtain the relevant part of the expected complete data log-likelihood for our proposed mixture model:

$$Q(\psi) = \sum_U \sum_W \sum_T M[u,t,w]\{ \sum_{\theta_i \in \Theta_U} p(\theta_i|u,t,w) log[\lambda_U p(w|\theta_i)p(\theta_i|u)] + \sum_{\theta_j \in \Theta_T} p(\theta_j|u,t,w) \log [\lambda_T p(w|\theta_j)p(\theta_j|t)]\} \quad (5)$$

Maximizing $Q(\psi)$ with respect to the parameters ψ and with the constraints $\sum_{w \in W} p(w|\theta_i) = 1$, $\sum_{w \in W} p(w|\theta_j) = 1$, $\sum_{\theta_i \in \Theta_U} p(\theta_i|u) = 1$ and $\sum_{\theta_j \in \Theta_T} p(\theta_j|t) = 1$, we can obtain the M-step re-estimation equations as follows:

$$p(\theta_i|u) = \frac{\sum_W \sum_T M[u,t,w]p(\theta_i|u,t,w)}{\sum_{\Theta_U} \sum_W \sum_T M[u,t,w]p(\theta_{i'}|u,t,w)} \quad (6)$$

$$p(\theta_j|t) = \frac{\sum_W \sum_U M[u,t,w]p(\theta_j|u,t,w)}{\sum_{\Theta_T} \sum_W \sum_U M[u,t,w]p(\theta_{j'}|u,t,w)} \quad (7)$$

$$p(w|\theta_i) = \frac{\sum_T \sum_U M[u,t,w]p(\theta_i|u,t,w)}{\sum_W \sum_T \sum_U M[u,t,w']p(\theta_i|u,t,w')} \quad (8)$$

$$p(w|\theta_j) = \frac{\sum_T \sum_U M[u,t,w]p(\theta_j|u,t,w)}{\sum_W \sum_T \sum_U M[u,t,w']p(\theta_j|u,t,w')} \quad (9)$$

With an initial random guess of $\{p(w|\theta_i),\ \{p(w|\theta_j),\ p(\theta_i|u),\ p(\theta_j|t)\}$, we alternately apply the E-step and M-step until a termination condition is met. Instead of picking a fixed λ_U, we estimate its value in M-step. This floating treatment can automatically adapt the model parameter estimation to various data sets with different proportions of temporal topics. After λ_U is decided in each iteration, we simply set λ_T to $1 - \lambda_U$. Specifically, λ_U is estimated as follows.

$$\lambda_U = \frac{\sum_T \sum_U \sum_W M[u,t,w] \sum_{\Theta_U} p(\theta_i|u,t,w)}{\sum_T \sum_U \sum_W M[u,t,w]C(\theta_i,\theta_j|u,t,w)} \quad (10)$$

where $C(\theta_i,\theta_j|u,t,w)$ is defined as

$$C(\theta_i,\theta_j|u,t,w) = \sum_{\Theta_U} p(\theta_i|u,t,w) + \sum_{\Theta_T} p(\theta_j|u,t,w)$$

V. Enhancement of the Mixture Model

In this section, we develop three methods to enhance our proposed mixture model for stable and temporal topics. Specifically, we attempt to exploit the temporal and spatial information to enhance the prior knowledge about users' topic distribution and temporal topic distribution.

Our enhancements make use of two intuitions. First, if two users $u, v \in U$ are close in the social network G, their user conditional probability distributions $p(\theta_i|u)$ and $p(\theta_i|v)$ are similar to each other. Second, if two time slices $t, t' \in T$ are close in the time axis, the temporal conditional probability distributions $p(\theta_j|t)$ and $p(\theta_j|t')$ approximate to each other. In other words, the user topic distribution $p(\theta_i|u)$ and the temporal topic distribution $p(\theta_j|t)$ change smoothly along the spatial and the temporal dimensions, respectively.

A. Spatial Regularization

The phenomenon of homophily in social networks is attributed to the effects of selection and social influence [22]. Selection means that people tend to form relationships with those sharing similar attitudes and interests. Due to social influence, related people in a social network tend to influence each other, and thus become more similar [22]. For example, two researchers who often coauthor with each other are likely to be working on the same topics. As another example, the interests and posting behaviors of a user on a microblog platform, like retweeting and forwarding on Twitter, are often affected by his neighbors on the graph structure of interactions. The increasing availability of online social media data has provided an valuable source of spatial information which, however, has not been fully utilized.

Motivated as such, we in this section exploit the spatial information in a social network to improve the process of topic detection. To integrate the information of spatial smoothness into our model, we make each user's topic distribution dependent on the topic distributions of her/his direct neighbors in the social network. Specifically, we propose a new spatial regularization framework to model user topics in the presence of a user network. The criterion of the regularizer in the framework is succinct and natural: users who are connected to each other tend to have similar weights of topics $p(\theta_i|u)$.

1) Framework: Formally, we define a regularized data likelihood as follows:

$$\mathcal{O}(\mathcal{C}, G) = L(\mathcal{C}) - \lambda R(\mathcal{C}, G) \quad (11)$$

where $L(\mathcal{C})$ is the document collection $\mathcal{C}$'s log likelihood (defined in Equation 2), and $R(\mathcal{C}, G)$ is a harmonic regularizer defined on the social network graph G. We adopt the network regularizer $R(\mathcal{C}, G)$ [18] as our spatial regularizer, which is defined as follows:

$$R(\mathcal{C}, G) = \frac{1}{2} \sum_{(u,v) \in E} \pi(u,v) \sum_{\Theta_U} (p(\theta_i|u) - p(\theta_i|v))^2 \quad (12)$$

Our proposed spatial regularization framework can leverage the power of both the topic model and the social network. Intuitively, $L(\mathcal{C})$ in Equation 11 measures how likely the data is generated by the topic model. By maximizing $L(\mathcal{C})$, we make $\{p(\theta_i|u)\}$ and $\{p(w|\theta_i)\}$ fit the text data as much as possible. By minimizing $R(\mathcal{C}, G)$, we smooth the topic distribution on the social network, where adjacent vertices have similar topic distributions. The parameter λ controls the balance between the data likelihood and the smoothness of topic distribution over the network. It is easy to see that if $\lambda = 0$, the objective function degenerates to the log likelihood of generating the document collection $\mathcal{C}$ in Equation 2.

By integrating Equations 2 and 12 into Equation 11, we have

$$\mathcal{O}(\mathcal{C}, G) = \sum_U \sum_T \sum_W M[u,t,w] \log p(w|u,t) - \frac{\lambda}{2} \sum_{u,v \in E} \pi(u,v) \sum_{\Theta_U} (p(\theta_i|u) - p(\theta_i|v))^2 \quad (13)$$

2) Model Parameter Estimation: In this part, we discuss parameter estimation of the mixture model enhanced with spatial regularization.

The enhanced model adopts the same generating schemes as that of the basic mixture model. Thus, the enhanced model has exactly the same E-step as that of the basic model. For the M-step, it can be derived that the relevant part of the expected complete data log-likelihood for the enhanced model is:

$$\begin{aligned}\mathcal{Q}(\psi) &= Q(\psi) - \lambda R(\mathcal{C}, G) \\ &= Q(\psi) - \frac{\lambda}{2}\sum_{u,v\in E}\pi(u,v)\sum_{\Theta_U}(p(\theta_i|u) - p(\theta_i|v))^2\end{aligned}$$

Since the spatial regularization part $R(\mathcal{C}, G)$ only involves the parameters $p(\theta_i|u)$, we can get the same M-step re-estimation equation for $p(\theta_j|t)$, $p(w|\theta_i)$ and $p(w|\theta_j)$ as in Equations 7, 8, and 9. However, we do not have a close form re-estimation equation for $p(\theta_i|u)$. In this case, the traditional EM algorithm cannot be applied.

In the following, we discuss how to use the generalized EM algorithm (GEM) [21] to maximize the regularized log-likelihood of our enhanced model in Equation 11. The major difference between EM and GEM is in the M-step. Instead of finding the globally optimal solution ψ which maximizes the expected complete data log-likelihood $\mathcal{Q}(\psi)$ in the M-step of EM algorithm, GEM only needs to find a "better" ψ in each new iteration. Let ψ_n denote the parameter values of the previous iteration and ψ_{n+1} denote the parameter values of the current iteration. The convergence of GEM algorithm only requires $\mathcal{Q}(\psi_{n+1}) \geq \mathcal{Q}(\psi_n)$ [21].

In each M-step, we have parameter values ψ_n and try to find ψ_{n+1} satisfying $\mathcal{Q}(\psi_{n+1}) \geq \mathcal{Q}(\psi_n)$. Obviously, $\mathcal{Q}(\psi_{n+1}) \geq \mathcal{Q}(\psi_n)$ holds if $\psi_{n+1} = \psi_n$. As $\mathcal{Q}(\psi) = Q(\psi) - \lambda R(\mathcal{C}, G)$, we first find $\psi_{n+1}^{(1)}$ which maximizes $Q(\psi)$ instead of the whole $\mathcal{Q}(\psi)$. This can be done by simply applying Equations 6, 7, 8 and 9. Clearly, $\mathcal{Q}(\psi_{n+1}^{(1)}) \geq \mathcal{Q}(\psi_n)$ does not necessarily hold. We then decrease $R(\mathcal{C}, G)$ by starting from $\psi_{n+1}^{(1)}$, which can be done by Newton-Raphson method [21]. Notice that $R(\mathcal{C}, G)$ only involves parameters $p(\theta_i|u)$, we only need to update the $p(\theta_i|u)$ part in ψ_{n+1}.

Given a function $f(x)$ and the initial value x_m, the Newton-Raphson updating formula to decrease (or increase) $f(x)$ is

$$x_{m+1} = x_m - \gamma\frac{f'(x_m)}{f''(x_m)} \tag{14}$$

Since we have the following regularizer:

$$R(\mathcal{C}, G) = \frac{1}{2}\sum_{u,v\in E}\pi(u,v)\sum_{\Theta_U}(p(\theta_i|u) - p(\theta_i|v))^2 \geq 0,$$

the Newton-Raphson method decreases $R(\mathcal{C}, G)$ in each updating step. Integrating $R(\mathcal{C}, G)$ and $\psi_{n+1}^{(1)}$ into Equation (14), we obtain the close form solution for $\psi_{n+1}^{(2)}$, and then $\psi_{n+1}^{(3)}$, ..., $\psi_{n+1}^{(m)}$, ..., where

$$\begin{aligned}p(\theta_i|u)_{n+1}^{(m+1)} &= (1-\gamma)p(\theta_i|u)_{n+1}^{(m)} \\ &\quad + \gamma\frac{\sum_{(u,v)\in E}\pi(u,v)p(\theta_i|v)_{n+1}^{(m)}}{\sum_{(u,v)\in E}\pi(u,v)}\end{aligned} \tag{15}$$

Obviously, $\sum_{\theta_i\in\Theta_U} p(\theta_i|u)_{n+1}^{(m+1)} = 1$ and $p(\theta_i|u)_{n+1}^{(m+1)} \geq 0$ hold in the above equation as long as $\sum_{\theta_i\in\Theta_U} p(\theta_i|u)_{n+1}^{(m)} = 1$ and $p(\theta_i|u)_{n+1}^{(m)} \geq 0$. Notice that the $p(w|\theta_i)_{n+1}$, $p(w|\theta_j)_{n+1}$ and $p(\theta_j|t)_{n+1}$ parts in ψ_{n+1} will keep the same.

Every iteration of Equation 15 makes the topic distribution smoother with respect to the nearest neighbors in the social network. The step parameter γ can be interpreted as controlling factor for smoothing the topic distribution among neighbors. When it is set to 1, the new topic distribution of a user is the average of the old distributions from the neighbors. This parameter affects the convergence speed only but not the convergence result.

We continue the iteration of Equation 15 until $\mathcal{Q}(\psi_{n+1}^{(m+1)}) \leq \mathcal{Q}(\psi_{n+1}^{(m)})$. After that, we test whether $\mathcal{Q}(\psi_{n+1}^{(m)}) \geq \mathcal{Q}(\psi_n)$. If not, we reject the proposal of $\psi_{n+1}^{(m)}$, and return the ψ_n as the result of the M-step. The above spatial regularization process is formalized in Algorithm 1.

Algorithm 1: Spatial Regularization

Input: Parameters ψ_{n+1}, Spatial Regularization parameters λ, Newton step parameter γ;
Output: $p(\theta_i|u)_{n+1}$;

$p(\theta_i|u)_{n+1}^{(1)} \leftarrow p(\theta_i|u)_{n+1}$;
`Compute` $p(\theta_i|u)_{n+1}^{(2)}$ as in Eqn. (15);
while $\mathcal{Q}(\psi_{n+1}^{(2)}) \geq \mathcal{Q}(\psi_{n+1}^{(1)})$ **do**
  $p(\theta_i|u)_{n+1}^{(1)} \leftarrow p(\theta_i|u)_{n+1}^{(2)}$;
  Compute $p(\theta_i|u)_{n+1}^{(2)}$ as in Eqn. (15);
end
if $\mathcal{Q}(\psi_{n+1}^{(1)}) \geq \mathcal{Q}(\psi_n)$ **then**
  $p(\theta_i|u)_{n+1} \leftarrow p(\theta_i|u)_{n+1}^{(1)}$;
end
`Return` $p(\theta_i|u)_{n+1}$;

B. Temporal Regularization

As temporal topics account for the evolution of global trends, a reasonable prior belief is that they change smoothly over time. We further assume that each time feature vector depends only on its immediate predecessor, and two consecutive temporal topic distributions are similar. So we propose a temporal regularizer as follows:

$$R(\mathcal{C}, T) = \sum_{t=1}^{|T|-1}\sum_{\Theta_T}^{k}(p(\theta_j|t) - p(\theta_j|t+1))^2 \tag{16}$$

Formally, we define the data likelihood with temporal regularization as follows:

$$\mathcal{O}(\mathcal{C}, T) = L(\mathcal{C}) - \xi R(\mathcal{C}, T) \tag{17}$$

The parameter ξ plays a similar role as λ in Equation 11, controlling the balance between the data likelihood and the smoothness of topic distribution over the time. Similar to the spatial regularization, the user-temporal mixture model with temporal regularization adopts the same generative schemes as that of the basic mixture model. Thus, the enhanced model has exactly the same E-step as that of the basic model. For the M-step, it can be derived that the expected complete data log-likelihood for the enhanced model is:

$$\mathcal{Q}(\psi) = Q(\psi) - \xi R(\mathcal{C}, T) \tag{18}$$

Due to the same reason with the spatial regularization, we cannot obtain a close form solution for $p(\theta_j|t)$ in the M-step. The generalized EM algorithm is also adopted to estimate all parameters and latent variables in the enhanced model with the temporal regularization since $R(\mathcal{C}, T)$ enjoys a similar form with the spatial regularizer $R(\mathcal{C}, G)$. Clearly, just as the spatial regularizer $R(\mathcal{C}, G)$, the temporal regularizer $R(\mathcal{C}, T)$ is non-negative, so the Newton-Raphson method decreases $R(\mathcal{C}, T)$ in each updating step. By integrating $\psi_{n+1}^{(1)}$ and $R(\mathcal{C}, T)$ into Equation 14, we obtain the close form solution for $\psi_{n+1}^{(2)}$, and then $\psi_{n+1}^{(3)}$, ..., $\psi_{n+1}^{(m)}$, ..., where

$$p(\theta_j|t)_{n+1}^{(m+1)} = (1-\gamma)p(\theta_j|t)_{n+1}^{(m)} + \gamma\frac{p(\theta_j|t-1)_{n+1}^{(m)} + p(\theta_j|t+1)_{n+1}^{(m)}}{2} \tag{19}$$

Notice that the $p(w|\theta_i)_{n+1}$, $p(w|\theta_j)_{n+1}$ and $p(\theta_i|u)_{n+1}$ remain the same in ψ_{n+1}, and the temporal regularization process is formalized in Algorithm 2.

Algorithm 2: Temporal Regularization

Input: Parameters ψ_{n+1}, Temporal Regularization parameters ξ, Newton step parameter γ;
Output: $p(\theta_j|t)_{n+1}$;

$p(\theta_j|t)_{n+1}^{(1)} \leftarrow p(\theta_j|t)_{n+1}$;
`Compute` $p(\theta_j|t)_{n+1}^{(2)}$ as in Eqn. (19);
while $\mathcal{Q}(\psi_{n+1}^{(2)}) \geq \mathcal{Q}(\psi_{n+1}^{(1)})$ **do**
  $p(\theta_j|t)_{n+1}^{(1)} \leftarrow p(\theta_j|t)_{n+1}^{(2)}$;
  Compute $p(\theta_j|t)_{n+1}^{(2)}$ as in Eqn. (19);
end
if $\mathcal{Q}(\psi_{n+1}^{(1)}) \geq \mathcal{Q}(\psi_n)$ **then**
  $p(\theta_j|t)_{n+1} \leftarrow p(\theta_j|t)_{n+1}^{(1)}$;
end
`Return` $p(\theta_j|t)_{n+1}$;

C. Burst-Weighted Smoothing

Topic models without considering individual burst words will lead to the problem that words with high occurrence rate, mostly denoting semantically abstract concepts, are likely to appear at top positions in summaries [6]. This phenomenon widely exists and is the direct result of the principle of probabilistic models, because words with more appearances tend to be estimated with high generation probabilities and ranked at top positions in a topic. In stable topics, these general words can illustrate the domains of topics at a first glimpse. However, in temporal topics, words with notable burst feature are superior in expressing temporal information compared to those with abstract meanings since users are more interested in burst words than in abstract categories when browsing a temporal topic.

An example of two kinds of words is shown in Figure 2. Three burst words "mj", "moonwalk" and "michaeljackson" have their distribution curves with sharp spikes. We can see that although the trends of these words do not always synchronize, they all go through a drastic increase and reach peaks in July 2009. The bursts in their curve are ignited by a real life event, i.e., Michael Jackson's death. An effective topic model should capture these words into one topic.

On the other hand, abstract words like "news" and "world" maintains high occurrences throughout the year in Figure 2 but they convey little information. Although they are relevant to the event in July, they also have relationships to many other topics. For example, word "news" could be used to represent various different news. However, such abstract words shadow the spikes of more meaningful words. The high occurrences of such abstract words during the burst period of the burst words may overwhelm the latter and render them unnoticed.

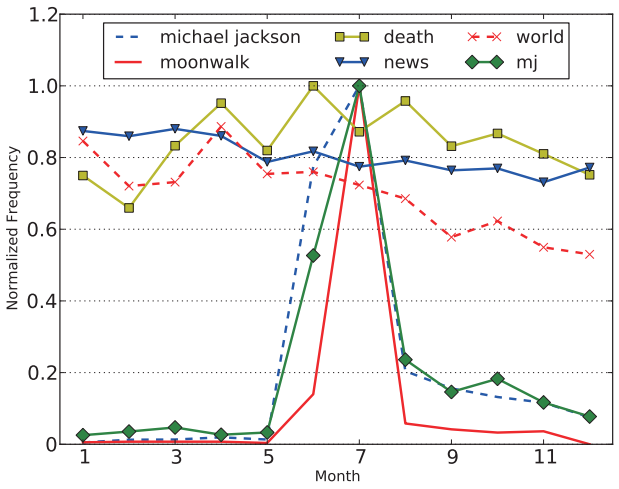


Fig. 2. Normalized Word Frequency Distribution on "Michael Jackson's Death" in 2009

To boost interesting temporal topics, we propose a smoothing technique that merges correlated words into one temporal topic and achieves higher visibility for such a topic.

In particular, we first conduct burst detection [15], [27] to detect individual burst words. Most of the spikes on words' temporal distribution curves are brought by discussion of correlated temporal topics. Burst detection on word granularity cannot clearly express the whole topic because of their semantic limitations. However, they are the core component of temporal topics, like what is illustrated in Figure 2.

Since words' burst information cannot be directly inferred from raw data, we apply an existing approach [27] by modeling word occurrence stream with sliding windows. We assume

that the occurrence of a word remains stable with a little fluctuation, which can be modeled by a Gaussian distribution $freq \sim (\mu, \sigma^2)$. As a result, $freq_t$ is the occurrence of the word at time t. We compare a word's occurrence $freq_t$ with sliding window containing its recent history periods $[t-n, t-1]$, and define word w's burst degree at t as $burst\_degree(w,t) = (freq_t - \mu_t)/\sigma_t$. Parameters μ_t and σ_t are mean and variance estimated by samples in a recent history window of t:

$$\mu_t = \frac{1}{n} \sum_{i=t-n}^{t-1} freq_i \tag{20}$$

$$\sigma_t = \sqrt{\frac{1}{n} \sum_{i=t-n}^{t-1} (freq_i - \mu_t)^2} \tag{21}$$

By applying the burst degree calculation on individual words, we detect burst periods for words. In Gaussian distribution, the probability for occurrence larger than $\mu + 2 * \sigma$ is less than 5%. Therefore, we deem that a word w is in its burst state when its burst degree $burst\_degree(w,t)$ is larger than 2. Statistics show that 14.7% of words in our data set have gone through at least one burst state accordingly.

Subsequently, we implement a burst smoothing step to escalate the probability of these burst words during the procedure of detecting temporal topics. The detailed usage of burst features is described as follows. First, when choosing words for input of our model, we loosen the occurrence threshold for burst words, which enables more of them to participate in training of topics even with slightly less occurrence. Second, we implement a smoothing step after each few E-M iterations for burst words. In this step, a word w will have its boosted probability on a temporal topic only if w's burst period overlaps with that of the topic. Specifically, we pick up time slice set $T_t = \{t_{j1}, t_{j2}, ..., t_{jm}\}$ on which topic θ_j has a high probability. A word w has its burst time slice set $T_w = \{t_{w1}, t_{w2}, ..., t_{wn}\}$. Only in the time slice $t \in T_w \cap T_t$, can the probability $p(w|\theta_j)$ be boosted according to Equation 22, as follows:

$$p(w|\theta_j) = p(w|\theta_j) * \sqrt{\max(burst\_degree(w,t), \alpha)} \tag{22}$$

where α is a positive parameter, denoting the threshold for activating the burst-weighted smoothing procedure. This is intuitive since we have to ensure the positive correlation between words and temporal topics. Only when the burst word is positively correlated with a temporal topic can the burst-weighted smoothing procedure be activated.

VI. Experimental Evaluation

In this section, we experimentally evaluate the effectiveness of our user-temporal mixture model and its enhanced variants.

A. Data Sets

Because no benchmark data is available for performance evaluation on stable and temporal topic detection, we collected two real-life data sets from Twitter and Del.icio.us.

Twitter: Twitter has been one of the most popular microblog services worldwide. The fast-updating nature of Twitter tweets attracts people to discuss many trending social events and their daily life. The Twitter data set contains 9,884,640 tweets posted by 456,024 users in the period of Mar. 2009 to Oct. 2009. Each user uploaded at least 200 posts. We first removed all the stop words and discarded tweets with fewer than three words. We then removed words with a tweet frequency less than 10.

Del.icio.us: Del.icio.us is a collaborative tagging system on which users can upload and tag web pages. We collected 200,000 users and their uploading behaviors from the period of Feb. 2008 to Dec. 2009. The data set contains 7,103,622 tags. Topics on technology and electronics cover more than half of all the posts. Nevertheless, breaking news with strong temporal features can also be traced in this data set. To avoid the time cost and noise brought by the long tail, we removed tags with a frequency less than 20 and discarded users with fewer than 10 tags.

B. Topic Detection Approaches in Comparison

Our proposed user-temporal mixture model has four variants: **BUT** is the basic model (Section IV), **EUTS** is the model enhanced with spatial regularization (Section V-A), **EUTT** is the model enhanced with temporal regularization (Section V-B), and **EUTB** is the model enhanced with the spatial regularization as well as the burst-weighted smoothing scheme (Sections V-C and V-A). We compare these variants with five categories of competitor approaches.

PLSA Model on Time Slices: We first implemented the traditional PLSA model by following a previous work [19]. In particular, we partitioned a given data set into time slices by posts' time stamps, and learned a set of topics for each time slice. We then built up story lines by comparing and linking similar topics in adjacent time slices to compose stable topics.

Detecting Stable and Temporal Topic Models Individually: The second competitor approach detects stable and temporal topics in two separate procedures. For the stable topic extraction, we ran the PLSA model on the whole data set. To detect temporal topics, we implemented the coordinated temporal topic detection method [24] with a single text stream. This method assumes that topics are generated by the time stamp, with a background topic for smoothing. We compare the stable and temporal topics separately detected to their counterparts detected by our user-temporal mixture model. To make the results comparison fair, we set the number of stable (and temporal) topics in this approach equal to that of the stable (and temporal) topics in our mixture model.

Topic Over Time Model (TOT): The third competitor is an LDA-style model [23] that treats both time stamps and words as variables generated by the latent topics. This model estimates the time distribution of each topic based on the beta distribution assumption. Whether a topic is labeled as "temporal" or "stable" is decided by the parameter which controls the shape of beta distribution.

TimeUserLDA: The TimeUserLDA model [7] is most relevant to our basic model in the aspect that they both consider users' topical interests and global topic trends. But there is only one type of topics in TimeUserLDA while our models pre-categorize topics into two types: stable topics and temporal topics. Each topic discovered by TimeUserLDA is in either the bursty state or the stable state for each time slice, which can be detected through a state machine method. To make the results comparable, we categorize their topics into two types according to their temporal states, as follows: a topic is assumed to belong to temporal topics if it has at least one bursty state; otherwise, the topic is assumed to be stable.

Twitter-LDA: The Twitter-LDA model [29] assumes that each post on microblogs is assigned a single topic and some words can be background words. As is presented in literature [7], this model in fact is a degenerate variation of TimeUserLDA and is also called UserLDA since it only considers the users' personal interests but not the temporal topical trends. To categorize the discovered topics into temporal topics and stable topics, a state machine-based method is applied, just like the method used in TimeUserLDA.

Tuning model parameters, such as the number of topics for all models, is critical to the model performance. We only report the optimal performance with tuned parameters and omit the details due to space constraint. By default, the parameter α, denoting burst degree threshold, was set to 2, the Newton step parameter γ was 0.1, the regularization parameters λ and ξ were 1000. For k_1 and k_2, we only report the performance with 60 stable topics and 40 temporal topics. Additionally, we set iteration number to 100, and partitioned each month into 10 time slices each of which covers about 3 days.

C. Evaluation Results

We use three methods to evaluate the results returned by the topic detection approaches in our comparison: *time stamp prediction*, *temporal component analysis* and *user study*. Note that the metric of perplexity, commonly used to measure the detected topics' fitness, does not suit temporal topic detection because it captures no temporal information.

1) Time Stamp Prediction: A prediction method [23] has been proposed to evaluate the quality of temporal topics by measuring the capability of predicting the time stamp given the words in a post. This method provides an opportunity to quantitatively compare our proposed models with the competitors in terms of the accuracy of predicting the time stamp of temporal posts.

In order to evaluate the quality of both stable and temporal topics in our experiments, we generated a testing set composed of both stable and temporal posts as follows. We first collected the representative keywords of popular events as seeds, and then we randomly sampled posts containing these seeds as temporal posts. Stable posts were generated likewise by choosing posts with the stable seeds. The composition of the testing set was controlled by changing the ratio between temporal and stable posts. Each temporal post is associated with a real time stamp. Either a correct guess of the time stamp for a temporal post or a successful recognition of a stable post would be considered as a correct prediction.

The process of prediction assumes that a post is generated from either a stable topic or a temporal topic. If a post has its keywords generated from one of the stable topics with the highest probabilities, the post is predicted as "stable". Otherwise, the testing post's time stamp is estimated from its corresponding temporal topic. We do not provide comparison with Twitter-LDA in this prediction task since Twitter-LDA does not consider temporal information and cannot be used to predict the time stamp of the temporal post. The prediction accuracy curve is illustrated in Figure 3.

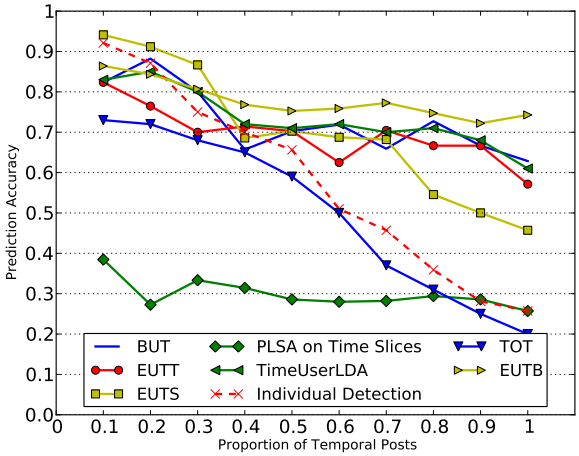


Fig. 3. Accuracy Curve of Time Stamp Prediction

From Figure 3, we observe that our proposed user-temporal mixture model and its enhanced variants outperform the non-mixture competitor approaches, especially when the proportion of temporal posts is high. As is expected, the mixture competitor TimeUserLDA achieves the similar performance with our basic model BUT, since it is also a mixture model that consider both users' topical interests and the global topic trends when discovering topics. It is should be noted that the accuracy of our proposed EUTB doubles that of all non-mixture competitors and is also higher than that of the TimeUserLDA when the testing set is composed of temporal posts only, which indicates that most of the temporal posts are successfully detected and predicted by EUTB. This superiority of EUTB verifies our intuition that by merging temporal topics with bursty features, we can better capture the bursty keywords in temporal posts and map them into the correct time stamps.

When the proportion of stable posts is high, our proposed models also yield comparable results. In particular, EUTS performs the best but its accuracy decreases clearly as the proportion of temporal topics increases. This indicates that spatial regularization is effective in stable topic detection. Although the competitor "Individual Detection" also achieves high prediction accuracy in this situation, its accuracy drops more drastically when the temporal proportion increases. This separate temporal topic detection method does not perform well as it is not able to grasp temporal features accurately.

By comparison among BUT, EUTB and EUTT, we observe that the temporal regularization dose not improve the temporal topic detection as the spatial regularization promotes the stable topic extraction, although they share similar forms and principles. This is because temporal topics enjoy a different nature from stable topics. Specifically, temporal topics are sensitive to time and are often related with breaking events while temporal regularization aims to reduce the difference between different time slices.

In summary, we have three conclusions. First, the accuracy of both temporal and stable topic detections benefit from the unified model that integrates two aspects of information. Second, spatial regularization, e.g., social network regularization, can improve the stable topic detection. Third, our proposed burst-weighted smoothing scheme can further improve the accuracy of temporal topic detection, which is supported by the accuracy difference between EUTB and EUTS.

2) Temporal Feature Component: Typical temporal topics should contain enough burst features to be distinguished from stable topics. Since different models have different latent topic parameters, their topic distributions are incomparable. To represent the topic's temporal distribution from the illustrative view, we define the topic distribution over the time axis as a weighted aggregation of its representative keywords: $D(\theta, t) = \sum_w p(w|\theta) * \frac{c(t,w)}{\sum_{w \in W} c(t,w)}$. Here $c(t, w)$ denotes the occurrence of word w at time t for all users. This metric can reflect the quality of topics from the aspect of valuing the temporal features inside topics. A successfully detected temporal topic should contain representative bursty keywords with spikes over time, while a stable topic should have a flat distribution over time.

We measure the curve's unevenness by its normalized variance of the curve. Intuitively, the larger the difference is between N-variances of temporal and stable topics, the better the model performs on this measurement. The results are listed in Table II. Our proposed user-temporal mixture model and its variants outperform other competitors in this metric. Temporal topics detected by user-temporal mixture model have much sharper curves, and therefore their normalized variances of curves significantly outnumber that of the temporal topics detected by competitors. On the other hand, our models have lower normalized variances on stable topics than the competitors, which means stable topics extracted by our models present smoother curve for stable topics. The competitor TimeUserLDA performs best among all competitor methods since it considers both users' interests and temporal topic trends simultaneously, just like ours. But, this method is beaten by our proposed EUTB due to that TimeUserLDA does not consider the social aspect and the burst features of words when discovering topics.

Among our proposed models, EUTS obtains the lowest normalized variance for stable topic extraction, which is consistent with the aforementioned conclusion that spatial regularization can improve the stable topic extraction. In addition, EUTB has the largest normalized variance value for temporal topic detection. These results support the aforementioned conclusion that burst-weighted smoothing scheme can further improve the accuracy of temporal topic detection.

Topic Detection Approach		N-Variance	Difference
BUT	stable topics	0.36	0.95
	temporal topics	1.31	
EUTS	stable topics	**0.21**	1.05
	temporal topics	1.26	
EUTT	stable topics	0.38	0.92
	temporal topics	1.30	
EUTB	stable topics	0.26	**1.34**
	temporal topics	**1.60**	
TOT	stable topics	0.39	0.11
	temporal topics	0.50	
Individual Detection	stable topics	0.38	0.61
	temporal topics	0.99	
TimeUserLDA	stable topics	0.39	0.93
	temporal topics	1.32	
Twitter-LDA	stable topics	0.38	0.58
	temporal topics	0.96	

TABLE II
N-VARIANCES OF DIFFERENT APPROACHES

3) User Study: In order to evaluate the quality of temporal topics detected by different models, we adopted similar method used in [7], and conducted a user survey on the Twitter data set by hiring 3 volunteers as annotators. For each topic, we extracted keywords with the highest probabilities to represent its content. Each topic was labeled by two different annotators, and if they disagreed a third annotator was introduced. Three exclusive labels were provided to indicate the quality of temporal topic detection.

- **Excellent:** a nicely presented temporal topic
- **Good:** a topic containing bursty features
- **Poor:** a topic without obvious bursty features

	Excellent	Good	Poor
EUTB	42.5%	32.5%	25%
TOT	10%	40%	50%
Individual Detection	20%	37.5%	42.5%
TimeUserLDA	29.5%	38%	32.5%
Twitter-LDA	13.5%	39%	47.5%

TABLE III
COMPARISON ON TEMPORAL TOPIC QUALITY

The labeling results are summarized in Table III. Up to 75% of the temporal topics detected by EUTB were labeled as "Excellent" or "Good", and 42.5% were regarded as "Excellent". Among all competitors, TimeUserLDA performs best. 67.5% of the detected temporal topics were judged as "Excellent" or "Good", and 29.5% were regarded as "Excellent". Other competitors got merely or slightly more than 50% of their detected topics labeled as "Excellent" or "Good". In particular, the competitors got significantly less "Excellent" labels. These results demonstrate that our proposed user-temporal mixture model enhanced with spatial regularization and burst-weighted smoothing outperforms its competitors in a remarked way.

4) Illustration of Topics Detected: In this section, we list part of the stable and temporal topics detected by our user-temporal mixture model on the two real data sets.

PLSA on slices	Individual Detection	TOT model	EUTB	TimeUserLDA
latest	michaeljackson	news	michaeljackson	news
headline	july	world	jackson	jackson
news	breaking	breaking	mj	michael
investigative	news	jackson	moonwalk	michaeljackson
michaeljackson	headline	michaeljackson	death	death
event	investigative	death	news	investigative

TABLE IV

TOPIC "MICHAEL JACKSON" DETECTED BY DIFFERENT APPROACHES

T77	T78	T87	T89	T60	T71
2009.1.12-2009.1.31	2009.6.15-2009.6.27	2009.4.24-2009.5.6	2009.5.27-2009.6.6	2009.1.24-2009.1.27	2009.1.1-2009.1.6
obama 0.144	moon 0.090	flu 0.158	google 0.061	droid 0.125	2008 0.099
inauguration 0.106	space 0.068	swineflu 0.124	googlewave 0.059	go 0.113	webcomics 0.046
bush 0.059	apollo11	pandemic 0.062	wave 0.042	dragonage 0.101	macworld 0.033
president 0.021	apollo 0.023	swine 0.050	bing 0.040	android 0.083	websites 0.028
gaza 0.017	nasa 0.019	health 0.020	apps 0.040	tricks 0.016	predictions 0.026
whitehouse 0.012	competition 0.015	disease 0.010	realtime 0.038	jailbreak 0.013	videogame 0.022

TABLE V

TEMPORAL TOPICS DETECTED ON DEL.ICIO.US DATA SET

T63	T86	T66	T70	T78	T94
2009.7.6-2009.7.15	2009.7.1-2009.7.6	2009.10.7-2009.10.15	2009.6.24-2009.6.30	2009.3.1-2009.4.15	2009.9.12-2009.9.15
july 0.012	july 0.035	free 0.012	michael 0.038	easter 0.010	kanye 0.016
free 0.010	happy 0.020	nobel 0.012	jackson 0.036	happy 0.009	patrick 0.011
summer 0.008	day 0.016	prize 0.011	rip 0.007	spring 0.008	swayze 0.011
live 0.007	firework 0.009	peace 0.008	farrah 0.007	day 0.007	west 0.007
potter 0.006	independ 0.006	win 0.008	dead 0.005	april 0.007	taylor 0.006
harry 0.006	celebrate 0.005	obama 0.008	sad 0.005	march 0.007	vma 0.004

TABLE VI

TEMPORAL TOPICS DETECTED ON TWITTER DATA SET

Our mixture model is able to detect a lot of meaningful temporal topics like "swineflu" or "webcomics" which competitor approaches cannot detect. Our mixture model also presents detected bursty topics in a more meaningful way than the competitors. An example of "michael jackson's death" is illustrated in Table IV. In other approaches, keywords with abstract semantics like "event", "headline", and "world" are estimated to be generated with high probabilities due to their high frequencies, and thus they are ranked in top positions of the detected topic. But, these words are of little help to describe the event. In contrast, our model clearly promotes concrete words like "moonwalk" because they have tight co-burst relationship with "Michael Jackson". Note that such low frequency words are generally ignored by existing approaches.

Further, we list the stable topics and temporal topics detected by our mixture model in Tables V to VIII. From these results, we can clearly tell the differences between stable and temporal topics. Stable topics are about a certain, common aspect with more general words as its composition, like topics "technology", "news", "international news" in Table VII. While temporal topics are closely related to specific events during a certain period of time. Referring to Table V, these six topics shown are about "U.S president inauguration", "Apollo 11 ceremony", "swineflu", "googlewave", "android" and "webcomics", respectively. All of them correspond to real-life events. The time when these topics reached their peaks, shown on the second row in the table, is very close to the time when these events happened in real life.

T10	T27	T33
windows 0.049	news 0.107	food 0.034
tools 0.048	latest 0.102	recipe 0.033
freeware 0.038	current 0.099	cooking 0.030
firefox 0.038	world 0.094	dessert 0.026
google 0.029	events 0.084	shopping 0.021
security 0.028	newspaper 0.084	home 0.016

TABLE VII

STABLE TOPICS DETECTED ON DEL.ICIO.US DATA SET

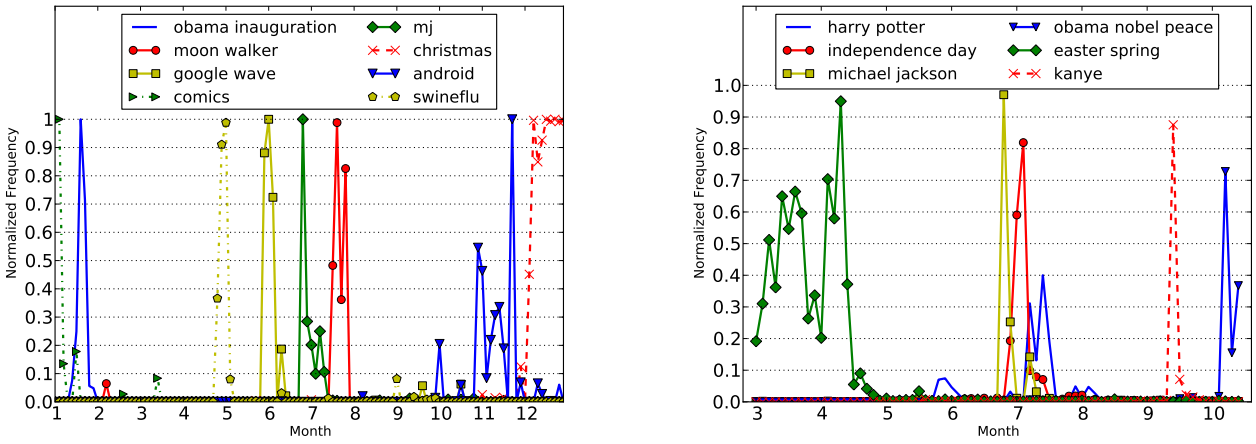
T5	T11	T53
free 0.020	day 0.104	assassin 0.039
market 0.011	travel 0.009	attempt 0.034
money 0.010	hotel 0.008	wound 0.024
people 0.007	check 0.006	level 0.020
check 0.007	site 0.004	reach 0.016
help 0.006	golf 0.004	account 0.013

TABLE VIII

STABLE TOPICS DETECTED ON TWITTER DATA SET

As shown in Table VI and VIII, topics detected on Twitter are more closely related to daily life, compared to the topics on Del.icio.us where users are more inclined to discuss technology. Table VI shows that temporal topics on Twitter are more correlated to season, festivals, breaking news as well as entertainment. These results are consistent with the different natures of Twitter and Del.icio.us.

Finally, Figure 4 shows the popularity curves of temporal topics detected by our model. Most of these temporal topics reach a peak in a short period of time and then disappear very soon, making a sharp spike on the curve. Topics with



(a) On Del.icio.us Data Set

(b) On Twitter Data Set

Fig. 4. Temporal Topic Distribution over Time

such a temporal distribution are the ones we intend to detect with typical bursty feature. These results clearly show that our mixture model is very effective in detecting such temporal topics from social media data.

VII. Conclusions

In this paper, we proposed a user-temporal mixture model for detecting temporal and stable topics simultaneously from the social media data. To enhance this mixture model, we developed a novel regularization framework that supports spatial and temporal regularizers in a unified way, and a burst-weighted smoothing scheme that promotes bursty features for temporal topic detection. Using two real social media data sets, we conducted extensive experiments to evaluate our proposals. The results demonstrate that our mixture model is able to discover and distinguish temporal topics from stable topics. Our mixture model enhanced with the spatial regularization and the burst-weighted smoothing scheme significantly outperforms competitor approaches, in terms of topic detection accuracy and discrimination in stable and temporal topics.

VIII. Acknowledgments

This research was supported by the National Natural Science Foundation of China under Grant No. 60933004, 61073019 and 61272155.

References

[1] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *SIGIR*, pages 37–45, 1999.
[2] L. AlSumait, D. Barbara, and C. Domeniconi. On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *IEEE Conf. on Data Mining*, pages 993–1022, 2008.
[3] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML*, pages 113–120, 2006.
[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. In *Journal of Machine Learning Research*, pages 993–1022, 2003.
[5] H. L. Chieu and Y. K. Lee. Query based event extraction along a timeline. In *SIGIR*, 2004.
[6] B. Cui, J. Yao, G. Cong, and Y. Huang. Evolutionary taxonomy construction from dynamic tag space. In *WISE*, 2010.
[7] Q. Diao, J. Jiang, F. Zhu, and E.-P. Lim. Finding bursty topics from microblogs. In *ACL*, pages 536–544, 2012.
[8] A. Gohr and A. Hinneburg. Topic evlolution in a stream of documents. In *SIAM*, 2009.
[9] D. He and D. Parker. Topic dynamics: An alternative model of 'bursts' in streams of topics. In *KDD*, 2010.
[10] T. Hofmann. Probabilistic latent semantic analysis. In *UAI*, 1999.
[11] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.
[12] L. Hong and B. D.Davison. Empirical study of topic modeling in twitter. In *SOMA*, 2010.
[13] L. Hong, D. Yin, J. Guo, and B. D. Davison. Tracking trends: Incorporating term volume into temporal topic models. In *KDD*, 2011.
[14] S. P. Kasiviswanathan, P. Melville, A. Banerjee, and V. Sindhwani. Emerging topic detection using dictionary learning. In *CIKM*, pages 745–754, 2011.
[15] J. Kleinberg. Bursty and hierachical structure in streams. In *KDD*, pages 91–101, 2002.
[16] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *KDD*, pages 497–506, 2009.
[17] C. X. Lin, B. Zhao, Q. Mei, and J. Han. Pet: A statistical model for popular events tracking in soical communities. In *KDD*, 2010.
[18] Q. Mei, D. Cai, D. Zhang, and C. Zhai. Topic model with network regularization. In *WWW*, pages 101–110, 2008.
[19] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: An exploration of temporal text mining. In *KDD*, pages 198–207, 2005.
[20] G. Pui, C. Fung, J. X. Yu, P. S. Yu, and H. Lu. Parameter free bursty events detection in text streams. In *VLDB*, pages 181–192, 2005.
[21] R.Neal and G.Hinton. *A view of the EM algorithm that justifies incremental, sparse, and other variants*. Kluwer, 1998.
[22] S.Wasserman and K.Faust. Cambridge Univeristy, 1994.
[23] X. Wang and A. McCallum. Topics over time: A non-markov continueous-time model of topical trends. In *KDD*, pages 424–433, 2006.
[24] X. Wang, C. Zhai, X. Hu, and R. Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *KDD*, pages 784–793, 2007.
[25] Y. Yang, T. Pierce, and J. Carbonell. A study on retrospective and on-line event detection. In *SIGIR*, 1998.
[26] J. Yao, B. Cui, Y. Huang, and X. Jin. Temporal and social context based burst detection from folksonomies. In *AAAI*, 2010.
[27] J. Yao, B. Cui, Y. Huang, and Y. Zhou. Detecting bursty events in collaborative tagging systems. In *ICDE*, pages 780–783, 2010.
[28] Q. Zhao, P. Mitra, and B. Chen. Temporal and information flow based event detection from social text streams. In *AAAI*, pages 1501–1506, 2007.
[29] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li. Comparing twitter and traditional media using topic models. In *ECIR*, pages 338–349, 2011.