Information Sciences 414 (2017) 102–116

Contents lists available at ScienceDirect

Information Sciences

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

User clustering in a dynamic social network topic model for short text streams

Zhangcheng Qiu[a], Hong Shen[a,b,*]

[a] School of Data and Computer Science, Sun Yat-sen University, China
[b] School of Computer Science, University of Adelaide, Australia

ARTICLE INFO

Article history:
Received 11 December 2016
Revised 19 March 2017
Accepted 10 May 2017
Available online 17 May 2017

Keywords:
User clustering
Social network
Topic model
Text clustering

ABSTRACT

Recently user clustering has become an increasingly important subject because of the high popularity of social medias like Twitter, Weibo and Facebook. The state-of-the-art algorithms of user clustering are all focused on the long or short text streams without considering factors of social network information, making them either unable to capture the social connectivity from short text streams or unable to conform to the sparsity, high-dimensionality and dynamically changing topics of short text streams. To address these issues, we propose a user clustering method named dynamic social network topic model (DSM) in this paper to cluster users by modeling their topics with dynamic features and social connectivity in short text streams. Experimental results show our topic model outperforms the state-of-the-art methods in the context of short text streams with social network information.

1. Introduction

With the explosive growth of the online social medias like Twitter and Weibo, short texts are prevailing in recent years. According to the Twitter's publication on June 30th 2016, it has 313 millions monthly active users and 500 millions tweets posted per day,[1] where a tweet is a short text of length not exceeding 140 words. Because a huge number of users share their opinions through social medias, if we can extract accurate and dynamic profiles from daily posted tweets of users, it obviously will be very helpful to the design of effective recommender systems, personalized tweets search engines and viral marketing. In this paper we investigate effective methods for user clustering on short text streams with social network information, aiming to infer topic distribution of users over time and dynamically cluster users based on their topic distributions in a way that users in the same cluster share similar interest and users in different clusters differ in their interest.

1.1. Motivation and scope

The state-of-the-art methods all focus on how to mine users behavior patterns based on long text, static short text or cluster users by using dynamic short text without considering social network information [8,15,22,38]. For instance, Mobasher et al. [22] modeled user features based on browsing vectors and then clustered users by the *k*-means method,

* Corresponding author at: School of Data and Computer Science, Sun Yat-sen University, China.
E-mail address: hong@cs.adelaide.edu.au (H. Shen).

[1] https://about.twitter.com/company.

http://dx.doi.org/10.1016/j.ins.2017.05.018
0020-0255/

Zhao et al. [38] proposed a topic model named UCT to generate user's topic distribution over time in short text streams first and next utilized *k*-means to cluster those user features to get user clusters. However, these methods assumed that there are no social circles among short text streams and ignored user's expression trends in the social network, so they are unable to capture user's interactive patterns. In the context of social networks, users often interact with their friends, such as "like", "repost" or comment on tweets from their followees. These interactions reflect the closeness among users and can affect their motivations to post a tweet. Unlike the previous work, we cluster users by extra using information of user social circles and their interactions in the social network to capture the latent topics from relations among users.

1.2. Research problem

It is still challenging to incorporate social network information in user clustering based on short text streams with short text sparsity, concept drift and user interactions. There are two main classes of traditional topic models: probabilistic latent semantic indexing (PLSI) [11], latent Dirichlet allocation (LDA) [4] and author topic model (AT) [26] focusing on static long text topic modeling; dynamic topic model (DTM) [3], topic over time model (ToT) [29] and topic tracking model (TTM) [13] concentrating on dynamic long text topic modeling. Besides, Gibbs sampling algorithm for the Dirichlet multinomial mixture model (GSDMM) [36] was designed for static short document and the dynamic user clustering model (UCT) [38] did not take into consideration of social network information. Hence, the existing methods do not conform to the context of dynamic short text with social circles. This motivates us to do research on developing a new user clustering model with the consideration of both user's social network information and dynamic short streams, which fuses the topics of users and their friends together. To our best knowledge, this is the first attempt to cope with user clustering using topic model by considering social circles.

The contributions of this work are summarized as follows:

1. We propose a dynamic social network topic model (DSM).
2. We propose a collapsed Gibbs sampling algorithm for DSM to infer the dynamic topic distributions of users.
3. We absorb the social circles in topic model to generate topic distribution.
4. We analyze the effectiveness of the proposed clustering models and demonstrate that our method significantly outperforms state-of-the-art methods.

The remainder of the paper is organized as follows: Section 2 discusses related work; Section 3 describes the proposed clustering model; Section 4 describes our experimental setup; Section 5 shows our experimental results and performance comparison with the existing methods. We conclude the paper in Section 6.

2. Related work

Related research to our work includes clustering, topic modeling and social network, which we summarize, respectively, below.

2.1. User clustering and text clustering

The previous studies on user clustering mainly fall into two categories which are respectively web user clustering and content-based user clustering. For the former, papers [1,5,22,28] studied user clusters formed by web logs including queries, user mouse click data and cursor movements to analyze user behavioral patterns. For instance Arapakis et al. [1] proposed to cluster users by using web logs including mouse tracking data. For the latter, papers [2,9,10] focused on extracting features by leveraging content similarity to cluster users. Recently, Zhao et al. [38] proposed a user clustering method based on twitter data set using topic model. But all these studies ignored the social circle factors during user feature extracting and hence failed to capture the true meanings of user's tweets in social networks during user clustering.

As a counterpart of user clustering, text clustering can also be categorized into two types respectively: long text clustering and short text clustering. For long text clustering [16,30,37], Wei et al. proposed a cluster-based document retrieval model, where the clusters are generated by LDA. For short text clustering [17,24,36], Yin and Wang [36] proposed a collapsed Gibbs Sampling algorithm for Dirichlet Multinomial Mixture model to tackle the sparsity and high-dimensionality problem of short text. Then Liang et al. [17] proposed a topic model named DCT to track dynamically changing topic distributions over documents. But again these methods did not consider the affect of social network information on clustering, therefor their models only work with static long text corpus or dynamic short text without social circle information.

2.2. Topic modeling

As for topic modeling, state-of-the-art topic models can be basically categorized into four types: static long text based topic modeling, dynamic long text based topic modeling, static and dynamic short text topic modeling. For the long text based topic modeling, Hofmann [11] proposed Probabilistic Latent Semantic Indexing (PLSI) which is an automated document indexing approach based on a statistical latent class model for factor analysis of count data. Blei et al. [4] proposed latent Dirichlet allocation (LDA), a generative probabilistic model for collections of discrete data such as text corpora, which

Table 1
Main notations used in this paper.

Notation	Gloss
z_d	Topic of document d
u_d	User of document d
e_w	Type of expression trend of word w
w	Word in document
V	A sorted set of words contained in whole corpus
Z	Total number of latent topics
$d_{u,t}$	Corpus of the documents created by user u in time slice $\boldsymbol{t}$
$\mathbf{d_t}$	Set of documents in time slice $\mathbf{t}$
$\lvert\mathbf{d_t}\rvert$	Number of arriving documents in time slice $\mathbf{t}$
f_u	A friend list of user u
N_d	Number of words in document d
$\phi_{t,e,z}$	Multinomial distribution of words specific to expression trend e (introduced in Section 3.3) with topic z in time slice $\boldsymbol{t}$
$\theta_{t,u}$	Multinomial distribution of topics specific to user u
$\beta_{t,e,z}$	Hyper-parameter of Dirichlet distribution, which is the prior distribution of topic-word distribution $\phi_{t,e,z}$ at time slice $\boldsymbol{t}$
$\alpha_{t,u}$	Hyper-parameter of Dirichlet distribution which is the prior distribution of $\theta_{t,u}$ in time slice $\boldsymbol{t}$

is a three-level hierarchical Bayesian model where each item of a collection is modeled as a finite mixture over an underlying set of topics. These topic models cannot capture the dynamically changing topics. Wang and McCallum [29] proposed ToT which is an LDA-style topic model that captures not only the low-dimensional structure of data, but also how the structure changes over time, and extensions of ToT such as DTM [3], TTM [13], DMM [31] were designed, with the focus on long text topic modeling. Furthermore, for short text based topic modeling, Yin and Wang [36] presented a topic model GSDMM that is static short text based topic model. And Zhao et al. [38] proposed UCT, they considered the sparsity of short text and the dynamically changing topics but ignored the social information in short text streams, which is not consistent with real world social networks and causes the clustering results do not accord with the real world social text streams due to lack of user social network information.

2.3. Social networks

The prior studies on text streams in social networks are mainly focused on recommender systems, such as [7,18,19,23,25,32,34,35]. Ma et al. [18,19] proposed probabilistic factor analysis frameworks which fused the users' tastes and their trusted friends' favors together to perform recommendations for items. Yang et al. [34] proposed friendship-interest propagation (FIP) to address interest recommendation and user link prediction tasks. Chen et al. [7] proposed a method of making tweet personalized recommendations based on collaborative ranking to capture personal interests, which considers tweet topic level factors, user social relation factors, authority of the publisher and quality of the tweet. Ye et al. [35] proposed a probabilistic generative model to model the decision making of item selection which indicate that the social influence from friends can be captured quantitatively. Besides, Cha et al. [6] revealed social connecting between users in social media. We can see that it is beneficial to incorporate social context information into topic modeling. To our best knowledge, we are the first to incorporate user expression trend and user social circles in topic modeling and it is also the first time to be applied in user clustering.

3. Our model

In this section, we describe our proposed clustering model. We start with preliminaries in Section 3.1, provide an overview of the model in Section 3.2, describe the proposed topic model for user clustering in Section 3.1 and show our way for user clustering with social network information in streams in Section 3.4.

3.1. Preliminaries

Before we detail the proposed model, we summarize the main notations used in this paper in Table 1.

The goal of the dynamic social network topic model is to infer the dynamically changing interests $\theta_{t,u}$ of users $u \in U_t$ by using their posts $d_{t,u}$ and social circle information f_u, in twitter context at every time slice $\mathbf{t}$ where time slice can be one day, one month or one quarter and so forth, $U_t = \{u_{t,1}, u_{t,2}, \ldots, u_{t,n}\}$, $d_t = \{d_{t,u} | u \in U_t\}$ and $\theta_{t,u}$ is topic distribution.

We then use $\theta_{t,u}$ to represent a user's dynamically changing features and cluster users by k-means.

3.2. Overview of the method

The proposed user clustering with dynamic social network topic model (DSM) consists of the following main steps and is presented in Algorithm 1.

Algorithm 1: Overview of the algorithm for user clustering by using short text streams and social information

input : U_t, D, f_u
output: Cluster of each user $C_{t,u}$

Step 1: Evaluate user-friends closeness distribution $\psi_{t,u}$

Step 2: Scan the expression trend e of every word in each tweet from each user

Step 3: Extract interests distribution $\theta_{t,u}$ of user based on dynamic social network topic model (DSM)

Step 4: Cluster users by using their interests distribution $\theta_{t,u}$ as feature

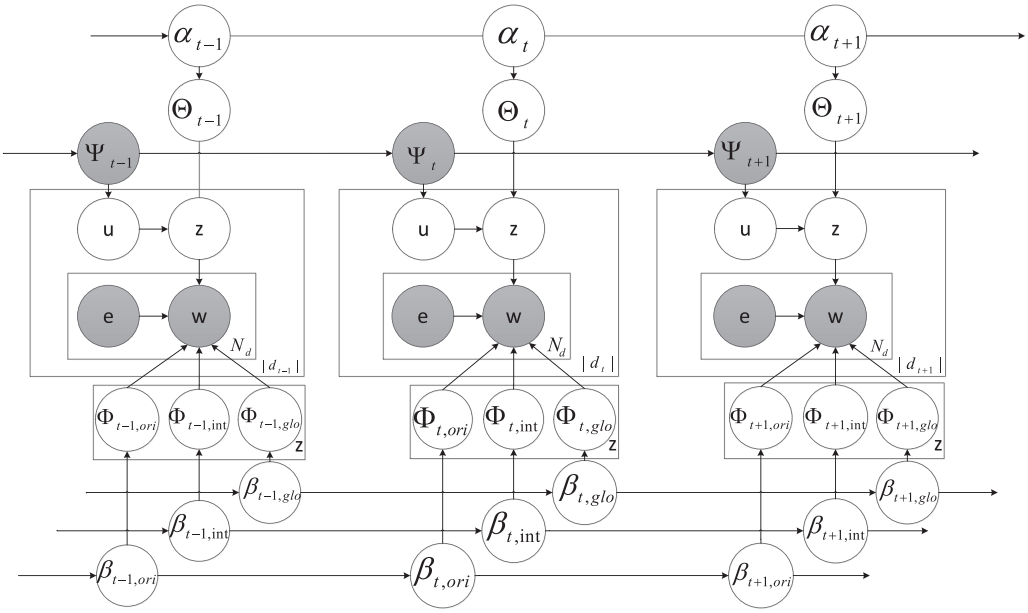

Fig. 1. Our topic model for short text streams in a social network.

Because short text does not provide enough information, we incorporate other context knowledge into our topic models. In this paper we take the view of "social circle influence" from friends to explain the topic sharing among users. Twitter is a typical short text stream application containing plenty of social circles, so we use it as our user clustering research setting. We propose dynamic social network topic model to cluster users in social circles based on their interests. To achieve this, we take latent topic distribution to represent user' s interest distribution. First, we evaluate user-friends closeness distribution in his social circle (see Step 1 in Algorithm 1). Second, to assure which kind of topic-word user takes in tweets we scan the expression trend of every word in every tweet (see Step 2 in Algorithm 1). Third, before clustering user, we use Gibbs sampling to sample all tweets with the above two results and then infer $\theta_{t,u}$ to represent user (see Step 3 in Algorithm 1). Finally, with the features of user, we use k-means to cluster users in time slice **t** (see Step 4 in Algorithm 1).

3.3. User topic model with social network

Traditional topic models do not consider dynamic user social circle information. To overcome this, we introduce user-friends closeness distribution and expression trend into our proposed dynamic social network topic model in Fig. 1 and motivations of the new two factors are explained as follows.

User-friends closeness (motivation: to measure the contribution of friends in composing a tweet).

As we all know, every twitter user has his own timeline that is a list or stream of tweets from all the people they follow, which gets updated in real time. Any follower including strangers can comment, like or repost any post from his followee. That is to say a user can get information from tweets published by his friends and others' comments before posting a tweet. So we may assume that every user in twitter setting posts a tweet will be influenced by some kinds of factor. Here we categorized them into three types, including user's own original interests, his friends interests and other factors like the comments on his tweets from his followers or strangers, where all these factors affect the composition of a new tweet.

In general, a user can have many followees but actually the user is interested in them by different degrees and we call this user closeness differentiation. In other words, for a user, his followees have different extents of impact on the user which depends on his closeness between each of them and him.

If we want to measure how much extent of impact that followee exerts on the user we should calculate the two person' s closeness that can be observed by their interactions and history tweets. The more similar their content of tweets are, the closer they are. Therefore, we can solve the closeness distribution ψ of user u by calculating lexical similarity of their tweets. Once upon we get the closeness distribution $\psi_{t,u}$, we can use it to represent how much contribution a friend provides to him in posting a tweet. In fact we regard a user's all friends as co-authors of his tweets and they have different contributions to each of the user' s tweets.

User expression trends (motivation: to track the different latent topic-word trend in time slice **t**. In different trends the topics are expressed in different ways of initiative habit, interactive habit and global habit, respectively).

Traditional topic models consider only one kind of topic-word distribution, which cannot model the social circle effect on topic-word distribution during the same time period. Therefore we propose that the topic-word exists in three fashions in every time slice, which includes initiative habit, interactive habit and global habit. These trends can cover all variations of topic-word in dynamic text stream. The initiative habit tracks user's own initiative composition fashion. The interactive habit captures topic-word distribution drifting cause by user-friends interactions, which explains how the relations among user and friends affect on expressing the topic of a tweet and choice of word. And the global habit describes something except above mentioned two expression trends in complex text streams which is dedicated to model the effect from other factors except friends and himself.

Topic of short text

In our daily operations on social media like Twitter, we assume that user can only express one main topic on his one post because of the restricted length of a tweet (140-character limit). If he wants to express more, he has to post another tweet. On the basis of what we have observed so far, people indeed always post a tweet for one topic. Hence we just sample the tweet by one topic, which says all words in same tweet are assigned to one topic. Therefore in our sampling phase we only sample one topic for all words in a tweet as Fig. 1 shows.

3.4. Clustering users in short text streams

Problem definition: given a corpus $D = \{d_{\mathbf{t}} | \mathbf{t} \in T\}$ with the addition of user-friends relations streaming in time $T = \{\mathbf{t_1}, t_2, \ldots, \mathbf{t_n}\}$, where each document $d_t \in d_{\mathbf{t}} = \{(text, uid, timestamp)_t | t \in \mathbf{t}\}$ contains time-stamp (timestamp) and user ID (uid), to cluster the users by semantic analysis of their posts in this corpus.

In other words, we divide the whole corpus into some pieces by time slice, which spans either one day, one week, one month or a half of year and so forth, then number those time pieces to get a time sequence set and cluster the users during every time slice **t** base on their topic features extracted from their posted tweets. Suppose there is a set of users U_t, and our first task is to infer their topic distributions $\Theta_t = \{\theta_{t,u} | u \in U_t\}$ and then use k-mean or other cluster methods to cluster these user features. Obviously inferring user's dynamic topic distribution is our mainly task here.

Evaluating the closeness distribution: to decide $\Theta_{\mathbf{t}}$, we need first infer the user closeness distribution $\Psi_t = \{\psi_{t,u} | u \in U_t\}$ at time **t**. The user closeness between two users can be modeled from two aspects, their interaction frequency and their time-line content similarity. The user interaction can be formalized as frequency of reposts, likes, comments and the lingering time in one' s certain tweet or homepage, where a high user interaction frequency of two users indicates high closeness between them. But most users are inclined to be silent user who just spend time lingering in others tweets and almost never comment or repost. For the limit of twitter streaming API we cannot obtain just the corpus without user reposts, likes, comments and lingering time. But we still can model the closeness of two users by computing $tf \times idf$ of two documents to measure the content similarity between them.

Introduction to TF-IDF [20]: assuming that there is a corpus A and we want to formalize each word in this as an algebraic value that represents its semantic. $tf(w, d)$ is the frequency of term w in document d, namely term frequency, which can be calculated as follows:

$$tf(w, d) = \frac{count(\mathrm{w})}{length(d)} \tag{1}$$

Inverse document frequency $idf(w)$ is the inverse document frequency of word w in document, like

$$idf(w, A) = \log \frac{|A|}{|\{d | w \in d \wedge d \in A\}|} \tag{2}$$

We organize all words in A by lexicographical ordering which forms a word partial set $W = \{w_1, w_2, \ldots, w_n\}$, where n is the number of words in corpus A. $tf(w, d)$ can be seen as the weight of word w in document d and $idf(w, A)$ represent the "topic" of w in corpus A. Therefore the value of $tf(w, d) \times idf(w, A)$ reflects the topic that word w indicates in document d in corpus thus a document d can be represented as a vector $vec(d)$ like $[tf(w_1) \times idf(w_1), tf(w_2) \times idf(w_2), \ldots, tf(w_n) \times idf(w_n)], w_i \in W$ using formulas (1) and (2).

Similarity: the similarity of two documents can be computed by computing the cosine distance of two vectors as follows,

$$sim(d_i, d_j) = \frac{vec(d_i) \cdot vec(d_j)}{|vec(d_i)| \times |vec(d_j)|}, d_i, d_j \in A \tag{3}$$

And to calculate the lexical similarity between two users, we use their timelines as a corpus and then get their content similarity by (3).

Closeness distribution: according to content similarity between and user u and each of his friend and the friends list f_u we can define the closeness distribution $\psi_{t,u}$ by normalizing their similarity vectors.

$$\psi_{t,u,f} = \frac{sim(f,u)}{\sum_{u' \in f_u \cup \{u\}} sim(u,u')} \tag{4}$$

$$\psi_{t,u} = [\psi_{t,u,u'}], u' \in \{u, f_1, f_2, \ldots, f_n\} \tag{5}$$

User expression trends: as mentioned in Section 3.3, user expression trends with topics and words can be formalized as topic-word distribution $\Phi_{t,e} = \{\phi_{t,e,z} | e \in style, z \in \mathbf{z}\}$ to reveal the generation of words in short text streams, where $\phi_{t,e,z}$ is a distribution of words under a certain topic which is a multinomial distribution. We assume that there exist three styles of topic-word distributions representing users' initiative expression habit (ori), interactive expression habit (int) and global circumstance expression habit (glo) respectively which can be formalized as $style = \{ori, int, glo\}$. That's to say given $e \in style$, $z \in \mathbf{z}$, we have $\sum_v \phi_{t,e,z,v} = 1$, where $\phi_{t,e,z,v}$ is a probability of word v under given the specific condition value of t, e and z, namely $w_d | \phi_{t,z_d,e} \sim Multinomial(\phi_{t,z_d,e})$.

We assume that when a user expresses one topic z using a word w he will be affected by those three expression trends under that topic z. For instance there is a user without observing any followee he may express a topic about "praying" by typing a tweet that "John is praying on the ground before game" but when all of his followees (his opinion leaders) start spreading a buzzword "tebowing" [39] in the tebowing fad), the user may use "tebowing" instead of "praying" by a certain probability that depends on what extent his friends affected him by that hot word, where "tebowing" was originated from the Denver Broncos' quarterback Tim Tebow who was photographed in the position as early as December 2010 and the pose itself is similar to the body building A Christian. He is known for praying in this stance before games therefore "tebowing" is defined as "to get down on a knee and start praying, even if everyone else around you is doing something completely different". In the twitter environment, the above mentioned phenomenon is prevailing and common and in fact users are always exposed to the tweet feeds posted by all his followees with some kinds of buzzword and then may absorb the buzzword in creating his own tweets. That is to say users expression trend can drift over time with his followees. Formally speaking, a word belonging to topic z_j can fall in the other topic z_i owing to some prevailing expression styles.

We split the expression trend into three kinds below. When observing a word in user' s tweet if this word appears in his friends tweets but not his own, we regard this as mostly influenced by his friend, so his expression style falls in interaction style. If word w is in his own, then we call this original style. If it is in neither of two the case, it falls in global category. By dividing expression style into different categories, we can infer topic-word in more detailed ways to express users' expression behavior.

Inferring user's interests distribution in time slice **t**: We assume that every user's interest will last for some time. In more technical words, this means that each user keeps his interests distribution $\theta_{t,u}$ unchanged until he posts new tweets in time slice $t+1$. In order to infer current interest of users in time slice **t**, we track topic distribution of users $\theta_{t,u}$ based on their previous interests $\theta_{t-1,u}$.

As mentioned before, we deem user as topic distribution $\theta_{t,u}$ with form of multinomial distribution, $\sum_z \theta_{t,u,z} = 1$ which indicates $z_d \sim Multinomial(\theta_{t,u})$. And the multinomial distribution is a conjugate distribution of Dirichlet distribution so we set the prior distribution of $\theta_{t,u}$ as a Dirichlet distribution with parameter $\theta_{t-1,u} \cdot \alpha_{t,u}$ for inference convenience, namely $\theta_{t,u} \sim Dirichlet(\theta_{t-1,u} \cdot \alpha_{t,u})$ and therefore the probability of $\theta_{t,u}$ can be obtained by the following:

$$P(\theta_{t,u} | \theta_{t-1,u}, \alpha_{t,u}) = \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \prod_{z=1}^{Z} \theta_{t,u,z}^{\alpha_{t,u,z} \cdot \theta_{t-1,u,z} - 1} \tag{6}$$

where $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$ is Gamma function and $P(\theta_{t,u} | \theta_{t-1,u}, \alpha_{t,u})$ is of Dirichlet distribution form with parameter of $\theta_{t-1,u} \cdot \alpha_{t,u}$

In contrast to traditional model, our Dirichlet prior parameter is $\theta_{t-1,u} \cdot \alpha_{t,u}$ rather than the static parameter α.

And in the same way the three kinds of topic-word distribution $\Phi = \{\phi_{ori}, \phi_{int}, \phi_{glo}\}$ defined in Section 3.1 can be obtained in (7), where we use the prior distribution parameter $\beta_{t,e,z}$ of $\phi_{t,e,z}$ and its previous distribution $\phi_{t-1,e,z}$ to track its dynamics, namely $\phi_{t,e,z} | \beta_{t,z,e} \cdot \phi_{t-1,z,e} \sim Dirichlet(\beta_{t,z,e} \cdot \phi_{t-1,z,e})$. To solve the short text sparsity, we assume that all words in a tweet share same topic with their affiliated tweet but different expression trend as Fig. 1 shows.

$$P(\phi_{t,z,e} | \phi_{t-1,z,e}, \beta_{t,z,e}) = \frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} - 1} \tag{7}$$

Once we have got the all expression trends of every word in tweet d and the dominant co-author u_d mentioned in Section 3.3 and the additional information like α_{t,u_d}, θ_{t-1,u_d}, $\boldsymbol{\beta}_{\mathbf{t,e}}$, $\phi_{t-1,e}$, we can infer the topic z_d of tweet as follows given a user u from $\{u_d\} \cup F_{u_d}$ as author of document d and expression trends for words in the document d. We deem all kinds of expression trends contribute independently to the probability of the topic for document d which consists of different

percentage of expression trends.

$$\begin{aligned} P(z_d = z|\mathbf{z}_{\mathbf{t},\neg \mathrm{d}}, \alpha_{t,u,}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e}) &= \frac{P(\mathbf{z}_{\mathbf{t},\mathrm{d}}, \mathbf{d_t}|\alpha_{t,u,}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e})}{P(\mathbf{z}_{\mathbf{t},\neg \mathrm{d}}, \mathbf{d_t}|\alpha_{t,u,}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e})} \\ &\propto \frac{P(\mathbf{z}_{\mathbf{t},\mathrm{d}}, \mathbf{d_t}|\alpha_{t,u,}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e})}{P(\mathbf{z}_{\mathbf{t},\neg \mathrm{d}}, \mathbf{d}_{\mathbf{t},\neg \mathrm{d}}|\alpha_{t,u,}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e})} \end{aligned} \tag{8}$$

where $e = \{ori, int, glo\}$, $\mathbf{d_t}$ represents documents in time slice $\mathbf{t}$, $\mathbf{z_t}$ represents topics of each document in time slice $\mathbf{t}$, $\mathbf{d}_{\mathbf{t},\neg \mathbf{d}}$ means the document d is removed from $\mathbf{d_t}$ and $\mathbf{z}_{\mathbf{t},\neg \mathbf{d}}$ means the topic of document d is remove from $\mathbf{z_t}$. And now we need infer joint distribution $P(\mathbf{z_t}, \mathbf{d_t}|\alpha_{t,u_d,}, \theta_{t-1,u_d}, \beta_{\mathrm{t,e}}, \phi_{t-1,e})$ and we assume that each document shows up independently therefore we can obtain joint distribution of $\mathbf{z_t}$ and $\mathbf{d_t}$ as follows.

$$\begin{aligned} P(\mathbf{z_t}, \mathbf{d_t}|\alpha_{t,u}, \theta_{t-1,u}, \beta_{\mathrm{t,e}}, \phi_{t-1,e}) &= P(\mathbf{d_t}|\mathbf{z_t}, \phi_{t-1,e}, \beta_{t,e}) P(\mathbf{z_t}|\theta_{t-1,u}, \alpha_{t,u}) \\ &= \int \prod_{d=1}^{|\mathbf{d_t}|} \prod_{i=1}^{N_{d,e}} P(w_{d_i}|\phi_{t,e,z_{d_i}}) \prod_{z=1}^{Z} P(\phi_{t,z,e}|\phi_{t-1,z,e}, \beta_{t,e}) d\phi_{t,ze} \\ &\quad \times \int \prod_{d=1}^{|\mathbf{d_t}|} P(z_{t,d}|\theta_{t,u}) P(\theta_{t,u}|\theta_{t-1,u}, \alpha_{t,u}) d\theta_{t,u} \end{aligned} \tag{9}$$

where $N_{d,\,e}$ represents the number of words with expression trend e in the d^{th} document in $\mathbf{d_t}$. And then we expand $P(\theta_t|\theta_{t-1,u_d}, \alpha_{t,u_d})$ and $P(\phi_{t,z,e}|\phi_{t-1,z,e}, \beta_{t,z,e})$ in (9) by applying its definition of Dirichlet distribution (6) and (7), respectively, therefore it becomes

$$\begin{aligned} &= \int \prod_{z=1}^{Z} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{n_{t,z,e,v}} \prod_{z=1}^{Z} P(\phi_{t,z,e}|\phi_{t-1,z,e}, \beta_{t,e}) d\phi_{t,z,e} \times \int \prod_{z=1}^{Z} \theta_{t,u,z}^{m_{t,u,z}} P(\theta_{t,u}|\theta_{t-1,u}, \alpha_{t,u}) d\theta_{t,u} \\ &= \int \prod_{z=1}^{Z} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{n_{t,z,e,v}} \prod_{z=1}^{Z} \left(\frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}-1} \right) d\phi_{t,z,e} \\ &\quad \times \int \prod_{z=1}^{Z} \theta_{t,u,z}^{m_{t,u,z}} \left(\frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \prod_{z=1}^{Z} \theta_{t,u,z}^{\alpha_{t,u,z} \cdot \theta_{t-1,u,z}-1} \right) d\theta_{t,u} \\ &= \prod_{z=1}^{Z} \frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{z=1}^{Z} \int \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}-1+n_{t,z,v,e}} d\phi_{t,z,e} \\ &\quad \times \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \int \prod_{z=1}^{Z} \theta_{t,u,z}^{\alpha_{t,u,z} \cdot \theta_{t-1,u,z}-1+m_{t,u,z}} d\theta_{t,u} \end{aligned} \tag{10}$$

where $m_{t,\,u,\,z}$ represents number of documents in topic z generated by user u in time slice $\mathbf{t}$, $n_{t,\,z,\,v,\,e}$ represents the number of occurrences of the vth word in word set V given topic z and expression trend e in time slice $\mathbf{t}$ and w_{d_i} represents the ith word in document d. And then we can simplify (10) with the help of Euler's integral so it becomes

$$\begin{aligned} &= \prod_{z=1}^{Z} \frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{z=1}^{Z} \left(\frac{\prod_{v=1}^{|V|} \Gamma(n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\Gamma(\sum_{v=1}^{|V|} (n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}))} \right) \\ &\quad \times \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \left(\frac{\prod_{z=1}^{Z} \Gamma(m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \right) \end{aligned} \tag{11}$$

And then applying (11) into (8) with the help of $\Gamma(x) = (x-1)\Gamma(x-1)$ and $\Gamma(x+m) = \prod_{i=1}^{m}(x+i-1)\Gamma(x)$ we can easily obtain the conditional distribution for Gibbs sampling with a quite elegant form as follows.

$$\begin{aligned} P(z_d = z|\mathbf{z}_{\mathbf{t},\neg \mathrm{d}}, \alpha_{t,u,}, \theta_{t-1,u}, \beta_{t,e}, \phi_{t-1,e}) \propto & \frac{\prod_{v \in d} \prod_{i=1}^{n_{t,d,e,v}} (n_{t,z_d,e,v,\neg d} + \beta \cdot \phi + i - 1)}{\prod_{i=1}^{n_{t,d,e}} (n_{t,z_d,e,\neg d} + \sum_{v=1}^{|V|} \beta \cdot \phi + i - 1)} \\ & \cdot \frac{m_{t,u,z_d} + \alpha_{t,u,z_d} \cdot \theta_{t-1,u,z_d} - 1}{\sum_{z=1}^{Z} m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z} - 1} \end{aligned} \tag{12}$$

where $n_{t,\,d,\,e}$ represents the number of words in document d with expression trend e in time slice $\mathbf{t}$, $n_{t,\,d,\,e,\,v}$ represents the number of occurrences of the vth word in word sorted set V with expression trend e in document d in time slice $\mathbf{t}$, m_{t,u,z_d} represents number of documents in topic z_d which is specific to document d referred to user u in time slice $\mathbf{t}$ and β, ϕ are abbreviations of $\beta_{t,\,e,\,z,\,v}$ and $\phi_{t-1,z,e,v}$ respectively for space limit concern.

So far we have inferred the user-topic distribution (6), topic-word distribution (7) and (12) for Gibbs sampling. According to the topic model in Fig. 1, we can sample the newly arriving documents as below.

Generative process:

1. For each topic $z \in [1, Z]$ and each expression trend $e \in \{ori, int, glo\}$:
   (a) Draw a multinomial distribution $\phi_{t,e,z}$ from Dirichlet prior $\beta_{t,e,z} \cdot \phi_{t-1,e,z}$.
2. For each tweet $d \in \mathbf{d_t}$ and observations $\mathbf{e}_{w \in d}$, its topic author u can be regarded that he is chosen from the set $\{u_d\} \cup F_{u_d}$:
   (a) Draw an "author" u' from user-friends closeness distribution $\Psi_{t,u}$;
   (b) Draw a multinomial distribution of topic $\theta_{t,u'}$ from its Dirichlet prior $\alpha_{t,u'} \cdot \theta_{t-1,u'}$;
   (c) Draw a topic z from topic distribution $\theta_{t,u'}$;
   (d) For each word $w \in d$: draw a word w from topic-word distribution $\phi_{t,e_w,z}$.

We use Gibbs sampling method to estimate Θ_t and Φ_t and the sampling process is given in Algorithm 2. After an iteration of sampling, we apply a fixed-point iteration for estimating the two Dirichlet prior parameters $\alpha_{t,u}$ and $\beta_{t,e}$ by maximizing the joint distribution $P(\mathbf{z_t}, \mathbf{d_t} | \alpha_{t,u_d}, \theta_{t-1,u_d}, \beta_{t,e}, \phi_{t-1,e})$ (11). We choose to maximize the logarithm rather than directly the joint distribution for derivation convenience.

$$\begin{aligned}
\log P(\mathbf{z_t}, \mathbf{d_t} | \alpha_{t,u_d,}, \theta_{t-1,u_d}, \beta_{t,e}, \phi_{t-1,e}) = {} & \partial_{z=1}^{Z} \sum \log \Gamma \left(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} \right) \\
& - \sum_{z=1}^{Z} \log \Gamma \left(\sum_{v=1}^{|V|} (\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} + n_{t,z,v,e}) \right) \\
& + \sum_{z=1}^{Z} \sum_{v=1}^{|V|} \log \Gamma (n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}) \\
& - \sum_{z=1}^{Z} \sum_{v=1}^{|V|} \log \Gamma \left(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} \right) \\
& + \log \Gamma \left(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z} \right) - \log \Gamma \left(\sum_{z=1}^{Z} (m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z}) \right) \\
& + \sum_{z=1}^{Z} \log \Gamma (m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z}) - \sum_{z=1}^{Z} \log \Gamma (\alpha_{t,u,z} \cdot \theta_{t-1,u,z})
\end{aligned} \tag{13}$$

Then we obtain the update rules of α and β by applying inequality scaling [21]: for any $x \in \mathbb{R}^+$, $n \in \mathbb{Z}^+$ and $\tilde{x}$ is an estimation of x, we have

$$\log \Gamma (\tilde{x}) - \log \Gamma (\tilde{x} + n) \geq \log \Gamma (x) - \log \Gamma (x + n) + (\Psi (x + n) - \Psi (x))(x - \tilde{x}) \tag{14}$$

and

$$\log \Gamma (\tilde{x} + n) - \log \Gamma (\tilde{x}) \geq \log \Gamma (x + n) - \log \Gamma (x) + x(\Psi (x + n) - \Psi (x))(\log \tilde{x} - \log x) \tag{15}$$

And then we let $\alpha_{t,u,z}^{new}$ as optimal value of $\alpha_{t,u,z}$ in the next fixed-point iteration, we will have the following inequality by using (14) and (15).

$$\begin{aligned}
& \log P(\mathbf{z_t}, \mathbf{d_t} | \{\alpha_{t,u,1}, \ldots, \alpha_{t,u,z}^{new}, \ldots, \alpha_{t,u,Z}\}, \theta_{t-1,u}, \beta_{t,e}, \phi_{t-1,e}) \geq F(\alpha_{t,u,z}^{new}) \\
& = \alpha_{t,u,z} \theta_{t-1,u,z} (\Psi (m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z}) - \Psi (\alpha_{t,u,z} \cdot \theta_{t-1,u,z})) \log (\alpha_{t,u,z}^{new} \theta_{t-1,u,z}) \\
& \quad - \alpha_{t,u,z}^{new} \theta_{t-1,u,z} \left(\Psi \left(\sum_{z=1}^{Z} m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z} \right) - \Psi \left(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z} \right) \right) + const.
\end{aligned} \tag{16}$$

And then we take $\frac{\partial}{\alpha_{t,u,z}^{new}}$ to $\alpha_{t,u,z}^{new}$ for $F(\alpha_{t,u,z}^{new})$ therefore we obtain the follows.

$$\begin{aligned}
\frac{\partial F(\alpha_{t,u,z}^{new})}{\partial \alpha_{t,u,z}^{new}} = {} & \frac{\alpha_{t,u,z} \theta_{t-1,u,z} (\Psi (m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z}) - \Psi (\alpha_{t,u,z} \cdot \theta_{t-1,u,z}))}{\alpha_{t,u,z}^{new}} \\
& - \theta_{t-1,u,z} \left(\Psi \left((r_{z=1}^{Z} \sum m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z} \right) - \Psi \left(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z} \right) \right)
\end{aligned} \tag{17}$$

And then we let (17) be zero then solve it to have the update rule for prior parameter $\alpha_{t,u,z}$ like following.

$$\alpha_{t,u,z}^{new} \leftarrow \frac{\alpha_t' (\Psi (\alpha_t' \theta_{t-1}' + m_{t,u,z}) - \Psi (\alpha_t' \theta_{t-1}'))}{\Psi (\sum_{z=1}^{Z} \alpha_t' \theta_{t-1}' + m_{t,u,z}) - \Psi (\sum_{z=1}^{Z} \alpha_t' \theta_{t-1}')} \tag{18}$$

Algorithm 2: Gibbs Sampling for DSM at time t

input : $\alpha_t, \theta_{t-1}, \beta_t, \phi_{t-1}, \phi_t, E_t$
output: $\alpha_{t+1}, \theta_t, \beta_{t+1}, \phi_t$

Initializing phase:

$n_{e_w,z,w} \leftarrow n_{t-1,e_w,z,w}$;
$n_{e_w,z} \leftarrow n_{t-1,e_w,w}$;
$m_{u,z} \leftarrow m_{t-1,u,z}$;
$M_u \leftarrow M_{t-1,u}$;
foreach $tweet \in \mathbf{d_t}$ **do**
  sample z, u randomly;
  $m_{u,z} \leftarrow m_{u,z} + 1$;
  $M_u \leftarrow M_u + 1$;
  foreach $w \in tweet$ **do**
    if $w \in doc_{u,t-1}$ **then**
      $e_w = ori$;
    else if $w \in doc_{f_u,t-1}$ **then**
      $e_w = int$;
    else
      $e_w = glo$;
    $n_{e,z,w} \leftarrow n_{e,z,w} + 1$;
    $n_{e,z} \leftarrow n_{e,z} + 1$;

Sampling phase:

for $iteration \leftarrow 1$ **to** N **do**
  for $d \leftarrow 1$ **to** $|\mathbf{d_t}|$ **do**
    foreach $w \in tweet_d$ **do**
      $n_{e,z,v} \leftarrow n_{e,z,v} - 1$;
      $n_{e,z} \leftarrow n_{e,z} - 1$;
    $m_{u,z} \leftarrow m_{u,z} - 1$;
    $M_u \leftarrow M_u - 1$;
    foreach $u' \in u \cup f_u$ **do**
      calculate $sim(u, u')$
    obtain ψ_t by normalizing $sim(u, u'), u' \in u \cup f_u$ and draw u_d from ψ_t ;
    $m_{u_d,z} \leftarrow m_{u_d,z} - 1$;
    $M_{u_d} \leftarrow M_{u_d} - 1$;
    draw z_d from $P(z_d|\mathbf{z_{t,-d}}, \mathbf{d_t}, \alpha_{t,u_d}, \theta_{t-1,u_d}, \beta_t, \phi_{t-1}, \mathbf{e_d})$;
    foreach $w \in tweet_d$ **do**
      $n_{e_w,z_d,w} \leftarrow n_{e_w,z_d,w} + 1$;
      $n_{e_w,z_d} \leftarrow n_{e_w,z_d} + 1$;
    /*for real author u plus the new topic z_d*/;
    $m_{u,z_d} \leftarrow m_{u,z_d} + 1$;
    $M_u \leftarrow M_u + 1$;
    /*for co-author u' plus the old topic z*/;
    $m_{u_d,z} \leftarrow m_{u_d,z} + 1$;
    $M_{u_d} \leftarrow M_{u_d} + 1$;
  update $n_{z,e,v}, n_{e,v}, m_{u,z}, M_{u,z}$;
  update α_t, β_t;
foreach $u \in U_t$ **do**
  update $\theta_{t,u}$
foreach $w \in tweet_d$ **do**
  update ϕ_{t,z_d,e_w,v_w};

where α'_t and θ'_{t-1} are $\alpha_{t,\,u,\,z}$ and $\theta_{t-1,u,z}$ for short, respectively, and $\Psi(x) = \frac{\partial \log \Gamma(x)}{\partial x}$ which is digamma function. And using the same way we can get the update rule for $\beta_{t,\,z,\,e,\,v}$ as below.

$$\beta^{new}_{t,z,e,v} \leftarrow \frac{\sum_{z=1}^{Z} \beta'_t \phi'_{t-1} (\Psi(n_{t,z,e,v} + \beta'_t \phi'_{t-1}) - \Psi(\beta'_t \phi'_{t-1}))}{\sum_{z=1}^{Z} \phi'_{t-1} (\Psi(\sum_{v=1}^{|V|} (n_{t,z,e,v} + \beta'_t \phi'_{t-1})) - \Psi(\sum_{v=1}^{|V|} \beta'_t \phi'_{t-1}))} \tag{19}$$

where β'_t and ϕ'_{t-1} are $\beta_{t,\,z,\,e,\,v}$ and $\phi_{t-1,z,e,v}$ for short, respectively. Given documents $\mathbf{d}_{\mathbf{t},\,\mathrm{u}}$ of user u in time slice $\mathbf{t}$ and list of expression trend for words in $\mathbf{d}_{\mathbf{t},\,\mathbf{u}}$ and applying Bayes' rule yields:

$$\begin{aligned}
P(\theta_{t,u} | \mathbf{z}_{\mathbf{d}_{\mathbf{t},\mathrm{u}}}, \alpha_{t,u} \cdot \theta_{t-1,u}) &= \frac{P(\theta_{t,u}, \mathbf{z}_{\mathbf{d}_{\mathbf{t},\mathrm{u}}} | \alpha_{t,u} \cdot \theta_{t-1,u})}{P(\mathbf{z}_{\mathbf{d}_{\mathbf{t},\mathrm{u}}} | \alpha_{t,u} \cdot \theta_{t-1,u})} \\
&= \frac{P(\mathbf{z}_{\mathbf{d}_{\mathbf{t},\mathrm{u}}} | \theta_{t,u}) P(\theta_{t,u} | \alpha_{t,u} \cdot \theta_{t-1,u})}{\int P(\mathbf{z}_{\mathbf{d}_{\mathbf{t},\mathrm{u}}} | \theta^*_{t,u}) P(\theta^*_{t,u} | \alpha_{t,u} \cdot \theta_{t-1,u}) d\theta^*_{t,u}} \\
&= \frac{\prod_{d=1}^{|d_{t,u}|} P(z_{t,d} | \theta_{t,u}) P(\theta_{t,u} | \alpha_{t,u} \cdot \theta_{t-1,u})}{\int \prod_{z=1}^{Z} (\theta^*_{t,u,z})^{m_{t,u,z}} \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \prod_{z=1}^{Z} (\theta^*_{t,u,z})^{\alpha_{t,u,z} \cdot \theta_{t-1,u,z} - 1} d\theta^*_{t,u}} \\
&= \frac{\prod_{z=1}^{Z} \theta_{t,u,z}^{m_{t,u,z}} \cdot \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \prod_{z=1}^{Z} \theta_{t,u,z}^{\alpha_{t,u,z} \cdot \theta_{t-1,u,z} - 1}}{\frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z})} \cdot \frac{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z} + m_{t,u,z})}{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z} + m_{t,u,z})}} \\
&= \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,u,z} \cdot \theta_{t-1,u,z} + m_{t,u,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,u,z} \cdot \theta_{t-1,u,z} + m_{t,u,z})} \prod_{z=1}^{Z} \theta_{t,u,z}^{m_{t,u,z} + \alpha_{t,u,z} \cdot \theta_{t-1,u,z} - 1} \\
&= Dirichlet(\theta_{t,u} | \{\alpha_{t,u,z} \cdot \theta_{t-1,u,z} + m_{t,u,z}\}_{z=1}^{Z})
\end{aligned} \tag{20}$$

$$\begin{aligned}
P(\phi_{t,z,e} | \mathbf{w}_{\mathrm{t,e}}, \mathbf{z}_{\mathbf{d_t}}, \beta_{t,z,e} \phi_{t-1,z,e}) &= \frac{P(\phi_{t,z,e}, \mathbf{w}_{\mathrm{t,e}} | \mathbf{z}_{\mathbf{d_t}}, \beta_{t,z,e} \phi_{t-1,z,e})}{P(\mathbf{w}_{\mathrm{t,e}} | \mathbf{z}_{\mathbf{d_t}}, \beta_{t,z,e} \phi_{t-1,z,e})} \\
&= \frac{\prod_{w \in \mathbf{w}_{\mathrm{t,e,z}}} P(w | \phi_{t,z,e}) P(\phi_{t,z,e} | \beta_{t,e} \phi_{t-1,e})}{\int \prod_{w \in \mathbf{w}_{\mathrm{t,e,z}}} P(w | \phi^*_{t,z,e}) P(\phi^*_{t,z,e} | \beta_{t,z,e} \phi_{t-1,z,e}) d\phi^*_{t,z,e}} \\
&= \frac{\prod_{v=1}^{|V|} \phi_{t,z,e,v}^{n_{t,z,e,v}} \cdot \frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} - 1}}{\frac{\Gamma(\sum_{v=1}^{|V|} \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\prod_{v=1}^{|V|} \Gamma(\beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \cdot \frac{\prod_{v=1}^{|V|} \Gamma(n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})}{\Gamma(\sum_{v=1}^{|V|} (n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}))}} \\
&= \frac{\Gamma(\sum_{v=1}^{|V|} (n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}))}{\prod_{v=1}^{|V|} \Gamma(n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v})} \prod_{v=1}^{|V|} \phi_{t,z,e,v}^{n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v} - 1} \\
&= Dirichlet(\phi_{t,z,e} | \{n_{t,z,v,e} + \beta_{t,z,e,v} \cdot \phi_{t-1,z,e,v}\}_{v=1}^{|V|})
\end{aligned} \tag{21}$$

where $\theta^*_{t,u}$ represents any kind of latent topic distribution of u and $\phi^*_{t,z,e}$ represents any kind of topic-word distribution for topic z with expression trend e. And using the expectation of the Dirichlet distribution on (20) and (21), respectively, yields:

$$\theta_{t,u,z} = \frac{m_{t,u,z} + \alpha_{t,u,z} \theta_{t-1,u,z}}{\sum_{z=1}^{Z} (m_{t,u,z} + \alpha_{t,u,z} \theta_{t-1,u,z})} \tag{22}$$

$$\phi_{t,z,e,v} = \frac{n_{t,z,e,v} + \beta_{t,z,e,v} \phi_{t-1,z,e,v}}{\sum_{v=1}^{|V|} (n_{t,z,e,v} + \beta_{t,z,e,v} \phi_{t-1,z,e,v})} \tag{23}$$

When the sampling process is convergent, we can count the topic distribution for user $u \in U_t$ by (22) and topic-word distribution for each expression trend by (23).

4. Experimental setup

In this section, we provide our experimental setup. We post our research questions in Section 4.1, describe the data set used in our experiments in Section 4.2, list the state-of-the-art baseline methods for performance comparisons in Section 4.3, and show the evaluation metrics in Section 4.4.

Table 2
Statistics of our data set before preprocessing.

	Total number of users	Average number of followees followed by each user	Average number of tweet posted by each user	Average length of tweet	Total number of words (including proper nouns)
Value	1414	≈ 847	≈ 2567	≈ 15	763,834

Table 3
Statistics of our data set after preprocessing.

	Total number of users	Average number of followees followed by each user	Average number of tweet posted by each user	Average length of tweet	Total number of words (including proper nouns after stemming)
Value	511	≈ 3.4	≈ 2776	≈ 7.3	223,461

4.1. Research questions

The following research questions guide the remainder of the paper:
RQ1 Does our dynamic user clustering with social network topic model outperform state-of-the-art baseline methods?
RQ2 What is the impact of the different time slices in our dynamic user clustering method?
RQ3 What is the quality of the topical representation inferred by our user clustering topic model?

4.2. Data set

Similar to [17,38], we work with a data set crawled from Twitter by using twitter streaming API to answer our research questions. The data set contains 511 active users with their current followees from time 2007-2-2 to 2015-5-21 (see Table 2) and they appear in Table 3 after data preprocessing. And we use this data set as our short text streaming context for experiments and then we manually judge the cluster of the users for ground truth by their contents of published tweets in different time periods like a week, a month, a quarter, half of a year and a year and with the number of clusters varying from 46 to 55, 41 to 51, 37 to 48, 26 to 31 and 28 to 30, respectively.

For our data set, the data preprocessing methods we take include: (1) convert all letters in the text corpus into lowercase; (2) Remove non-Latin characters and stop words; (3) Perform stemming for words with the nltk[2] python packages; (4) Remove words with length smaller than 2 or larger than 15; (5) Remove the users who do not have any followee or follower in our data set. (6) Remove the users' friends who are not included in our data set.

4.3. State-of-the-art baseline methods

We compare our proposed DSM user clustering algorithm with the following baseline methods:

k-*means with TF-IDF.* It is a traditional clustering algorithm [14]. We use TF-IDF to extract word statistical information from text as user features and categorize them into different clusters based on their TF-IDF vector similarities.

Latent Dirichlet Allocation (LDA) [4]. It is a probabilistic document model that assumes each document is a mixture of latent topics.

Dynamic topic model (DTM) [3]. It can model topic distributions of long text streams by using a Gaussian distribution for inferring.

Topic tracking model (TTM) [13]. It is a probabilistic consumer purchase behavior model for tracking the interests of individual consumer and trends in each topic which can be used to model long text streams.

Collapsed Gibbs Sampling algorithm for the Dirichlet Multinomial Mixture model (GSDMM) [36]. It is a Dirichlet multinomial mixture model-based approach for short text clustering, which represents each short document through a single topic.

User Clustering Topic model (UCT) [38]. This model captures dynamic topic distribution over time in short text streams but ignores social relations among users.

In our experimental setup for LDA, DTM, TTM, we represent users by utilizing their averaged topic distributions extracted from their posted tweets respectively and set $\alpha = 0.1$ and $\beta = 0.01$ for LDA. Besides, we set the number of topics $K = 50$ and the number of topics is equal to the number of clusters.

4.4. Evaluation metrics

For quantitative evaluation, we adopt four clustering evaluation metrics, precision, NMI, ARI and H-score for evaluating the performance of our proposed method and the baselines, where those evaluation metrics are widely used in [17,33,36,38].

To facilitate description, we predefined two cluster sets, a ground truth set $\mathbb{C} = \{c_1, c_2, \ldots, c_P\}$ where $P = |\mathbb{C}|$ and output set $\Omega = \{\omega_1, \omega_2, \ldots, \omega_Q\}$, where $Q = |\Omega|$, respectively.

[2] http://www.nltk.org/.

Precision. Precision is a measure that allows us to evaluate the output cluster's percentage of correctly clustered users. For every element ω_i in output set Ω, we first assign it to one ground truth cluster c_j which has max intersection with ω_i. Then we use $\frac{\max_j |\omega_i \cap c_j|}{|\omega_i|}$ to measure the output set ω_i's precision, which reflects the correct number of users clustered. Therefore the precision of output set Ω based on ground truth set $\mathbb{C}$ can be calculated by sum up all output cluster's precision and normalized by division of $|\Omega|$,

$$Precision(\mathbb{C}, \Omega) = \frac{1}{|\Omega|} \sum_{i=1}^{|\Omega|} \frac{\max_j |\omega_i \cap c_j|}{|\omega_i|} \tag{24}$$

Normalized Mutual Information (NMI) [27,33]. We observe that a high precision is easy to achieve. For instance, when $|\mathbb{C}| = 1$ every output cluster gets its own cluster, which makes precision=1 no matter how many clusters output. So precision cannot trade off the quality of the clustering against the number of clusters. But NMI is a measure that allows us to make this trade off.

$$\begin{aligned} NMI(\mathbb{C}, \Omega) &= \frac{I(\Omega, \mathbb{C})}{(E(\Omega) + E(\mathbb{C}))/2} \\ &= \frac{\sum_{i,j} \frac{|\omega_i \cap c_j|}{N} \log \frac{|\omega_i \cap c_j|}{|\omega_i||c_j|}}{(-\sum_i \frac{|\omega_i|}{N} \log \frac{|\omega_i|}{N} - \sum_j \frac{|c_j|}{N} \log \frac{|c_j|}{N})/2} \end{aligned} \tag{25}$$

where $I(\Omega, \mathbb{C})$ is mutual information between set Ω and $\mathbb{C}$. It measures the amount of information by which our knowledge about the classes increases when we are told what the clusters are.

$$\begin{aligned} I(\Omega, \mathbb{C}) &= \sum_k \sum_j P(\omega_k \cap c_j) \log \frac{P(\omega_k \cap c_j)}{P(\omega_k) p(c_j)} \\ &= \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|} \end{aligned} \tag{26}$$

where $P(\omega_k)$, $P(c_j)$, and $P(\omega_k \cap c_j)$ are the probabilities of a user being in cluster ω_k, class c_j, and in the intersection of ω_k and c_j, respectively. $E(\Omega)$ is entropy of Ω.

$$E(\Omega) = -\sum_k P(\omega_k) \log P(\omega_k) = -\sum_k \frac{|\omega_k|}{N} \log \frac{|\omega_k|}{N} \tag{27}$$

Adjusted Rand Index (ARI) [12,27,33,38]. Consider user clustering as a series of pair-wise decisions. If two user both in the same class and are assigned to the same cluster, or both in different classes are assigned to different clusters, the decision is considered to be correct, otherwise it is false. Rand index measures the percentage of decisions that are correct. Adjusted Rand index is the corrected-for-chance version of Rand index, whose expected value is 0, while the maximum value is also 1 for exact match.

$$ARI(\mathbb{C}, \Omega) = \frac{\sum_{i,j} \binom{|\omega_i \cap c_j|}{2} - [\sum_i \binom{|\omega_i|}{2} \sum_j \binom{|c_j|}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{|\omega_i|}{2} + \sum_j \binom{|c_j|}{2}] - [\sum_i \binom{|\omega_i|}{2} \sum_j \binom{|c_i|}{2}]/\binom{n}{2}} \tag{28}$$

H-score [33,38]. Because our clustering method is to capture the topic representation of each user in social network, we need a metric to evaluate how well our topic representations are. The intuition is that if the average inter-cluster distance is small compared to the average intra-cluster distance, the topical representation of users agrees well with human labeled clusters, which can be formalized as follows.

$$H - score(\Omega) = \frac{IntraDis(\Omega)}{InterDis(\Omega)} \tag{29}$$

where the average intra-cluster distance $IntraDis(\Omega)$ and average inter-cluster distance $InterDis(\Omega)$ are computed as:

$$IntraDis(\Omega) = \frac{1}{|\Omega|} \sum_{|\Omega|} \sum_{\substack{u_i, u_j \in \omega_q \\ i \neq j}} \frac{dis(u_i, u_j)}{\binom{|\omega_q|}{2}} \tag{30}$$

$$InterDis(\Omega) = \frac{1}{|\Omega|(|\Omega| - 1)} \sum_{\substack{\omega_m, \omega_n \in \Omega \\ m \neq n}} \Big[\sum_{\substack{u_i \in \omega_m \\ u_j \in \omega_n}} \frac{dis(u_i, u_j)}{|\omega_m||\omega_n|} \Big] \tag{31}$$

where $dis(u_i, u_j)$ is the symmetric Kullback–Leibler distance of topic distributions of user u_i and user u_j.

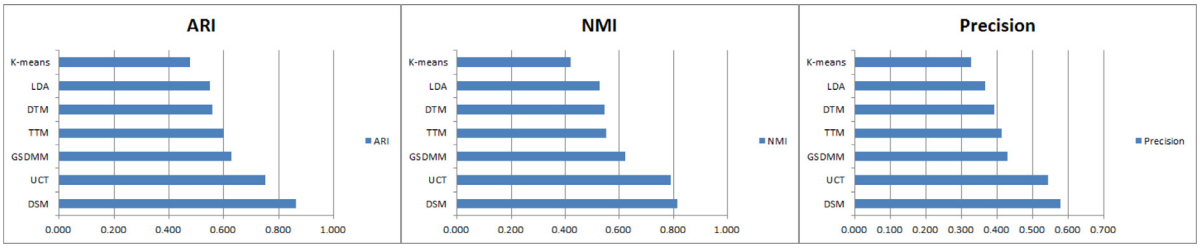

Fig. 2. The ARI, NMI, precision results of DSM and baselines in time slice of a quarter.

For all these metrics we use sklearn[3] as implementation tool in our experimental study.

5. Experimental results and analysis

In this section we report our experimental outcomes and give analysis for the results.

5.1. Effectiveness of DSM

We analyze our experiment result with regarding to precision, NMI, ARI scores. We set time slice a quarter and conduct our comparison experiment with other 6 baselines, *k*-means with TF-IDF, LDA, DTM, TTM, GSDMM and UCT. As we mentioned before, LDA is designed for static long text, and DTM, TTM for dynamic long text streams, GSDMM for short text topic modeling and UCT for dynamic short text streams.

We demonstrate our experimental results in 4 dimensions.

1. Method effectiveness. We can see all topic models are better than *k*-means with TF-IDF method in three sets of evaluation, which demonstrates that topic modeling is better than TF-IDF in text modeling for clustering. The reason behind this result is obvious—topic model is a probabilistic model which analogues the real generative process of text by utilizing probability theory but TF-IDF is only based on statistical information like word frequency which doesn't capture semantic meanings.

2. Time factor. We can find that DTM and TTM are better than LDA, and UCT is better than GSDMM, which shows that the models considering time factor suit more for dynamic text streams than those ignoring dynamically changing topic distribution. We also see that our model DSM outperforms all static models because we take in the time factor in our topic model.

3. Short text factor. Owing to assignment of only one topic for short text, models like GSDMM, UCT and DSM outperform others, which demonstrates that single topic assignment is better than multiple topics for short text.

4. Social relation factor. In Fig. 2 we can see that DSM model significantly outperforms other methods in NMI, ARI and precision. Especially, DSM surpass UCT in all metrics. This is because DSM considers social relations but UCT does not. This result demonstrates the value of user social relations for the user clustering task.

In summary, our DSM model outperforms all baselines in three evaluation metrics for cluster quality. This validates our method's effectiveness in user clustering with shot text streams in social networks.

5.2. Length of time slices

In this part, we analyze precision, NMI, ARI scores of all baselines mentioned above in different time slice settings. For understanding the effect exerted by different time slices on DSM, we compare the performance of those models in different time slices: a week, a month, a quarter, half a year and a year, respectively, and we illustrate the precision, NMI, ARI scores in Fig. 3.

From Fig. 3 we can observe that no matter what time slice we set, our DSM model still works better than all other competitors in user clustering task. This is because our model utilizes rich context information from the short text streams like friend relation, user expression trend and topic dependency of previous topics but others do not. We also find that as the length of the time slice goes longer, the three kinds of scores of DSM go higher. This can be explained that the longer the time slice is, the more user interactions can be found in their tweets as their friendship goes strong. In our model, the more user interactions we captured, the better topic distribution it gets.

Additionally, it is also found that the performances of all methods increase as the time slice expands. This is because that in our data set, the number of tweets in a week is too few to be used for inferring. With the alleviating of sparsity of data set as time slice goes longer, all methods can utilize more reliable data to support the Gibbs sampling and therefore its performance correspondingly increases.

[3] http://scikit-learn.org/.

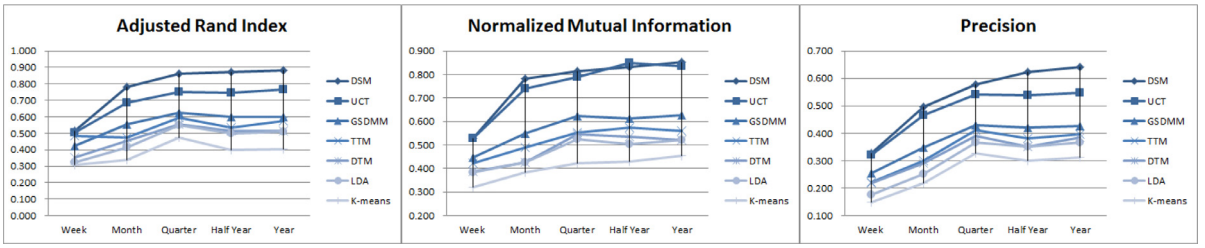

Fig. 3. The ARI, NMI, precision results of DSM and baselines in different time slice settings.

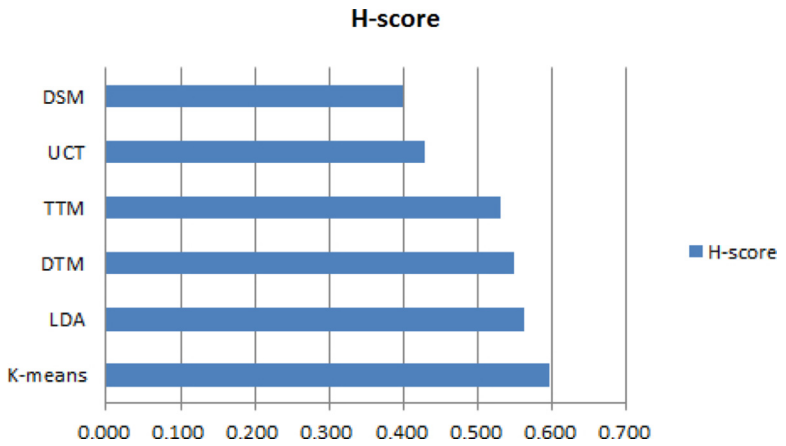

Fig. 4. The *H*-score results of DSM and baselines in time slice of a quarter.

5.3. Quality of topic representations

As mentioned in Section 4.4, the metric *H*-score can be used for evaluating the topical representations. Therefore, to assess topical representation quality of DSM, we compare our model with other baselines. The results shown in Fig. 4 illustrate that DSM gets better scores than UCT, TTM, DTM and LDA. A smaller *H*-score representing a higher intra-distance indicates that the clustering result is more consistent with the ground truth and also reflects a better quality of topic representation. Because DSM has the smallest *H*-score among all models, it achieves the highest quality of topic representations.

6. Conclusion

In this paper, we have proposed Dynamic Social Network Topic Model for user clustering (DSM) with capturing the dynamic topic distributions in the social network context. Unlike the traditional topic models, our model not only considers the dynamics of topic distribution of users but also incorporates the social relations among users, where the former consideration can solve the concept-drift problem in text streams and the latter can address the drifting of topic-word distribution among users. Then to cope with the sparsity of short text, we assign the same topic for all words in one short text during Gibbs sampling for the inference. And we have proposed a new collapsed Gibbs sampling algorithm for our proposed DSM.

For quantitative evaluation of our clustering method's effectiveness, we conduct comparative experiments of performance with different metrics between our model and state-of-the-art models including traditional models *k*-means with TF-IDF, LDA, DTM, TTM, GSDMM and the very recent model User Clustering Topic (UCT). Our experimental results shows that our DSM achieves a better effectiveness and yields higher quality of topic representation than other topic modeling methods.

Acknowledgment

This work is supported by Research Initiative Grant of Sun Yat-sen University under Project 985 and Australian Research Council Discovery Projects funding DP150104871.

References

[1] I. Arapakis, M. Lalmas, G. Valkanas, Understanding within-content engagement through pattern analysis of mouse gestures, in: Proceedings of the Twenty-third ACM International Conference on Conference on Information and Knowledge Management, ACM, 2014, pp. 1439–1448.

[2] K. Balog, M. de Rijke, Finding similar experts, in: W. Kraaij, A.P. de Vries, C.L.A. Clarke, N. Fuhr, N. Kando (Eds.), Proceedings of the Thirtieth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2007), ACM, Amsterdam, The Netherlands, July 23–27, 2007, pp. 821–822, doi:10.1145/1277741.1277926.

[3] D.M. Blei, J.D. Lafferty, Dynamic topic models, in: Proceedings of the Twenty-third International Conference on Machine learning, ACM, 2006, pp. 113–120.
[4] D.M. Blei, A.Y. Ng, M.I. Jordan, Latent Dirichlet allocation, J. Mach. Learn. Res. 3 (Jan. 2003) 993–1022.
[5] G. Buscher, R.W. White, S. Dumais, J. Huang, Large-scale analysis of individual and task differences in search result page examination strategies, in: Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, ACM, 2012, pp. 373–382.
[6] M. Cha, H. Haddadi, F. Benevenuto, P.K. Gummadi, Measuring user influence in Twitter: the million follower fallacy, in: Proceedings of the 2010 International Conference on Web and Social Media ICWSM, 10, 2010, p. 30.
[7] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, Y. Yu, Collaborative personalized tweet recommendation, in: Proceedings of the Thirty-fifth International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2012, pp. 661–670.
[8] W. Chen, J. Wang, Y. Zhang, H. Yan, X. Li, User based aggregation for biterm topic model, in: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), Beijing, China, July 26–31, 2015, pp. 489–494.
[9] W. Chen, J. Wang, Y. Zhang, H. Yan, X. Li, User based aggregation for biterm topic model, in: Proceedings of the Fifty-third Annual Meeting of the Association for Computational Linguistics and the Seventh International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, 2, The Association for Computer Linguistics, Beijing, China, July 26–31, 2015, pp. 489–494. Short Papers.
[10] C.V. Gysel, M. de Rijke, M. Worring, Unsupervised, efficient and semantic expertise retrieval, in: J. Bourdeau, J. Hendler, R. Nkambou, I. Horrocks, B.Y. Zhao (Eds.), Proceedings of the Twenty-fifth International Conference on World Wide Web, WWW 2016, ACM, Montreal, Canada, April 11–15, 2016, pp. 1069–1079, doi:10.1145/2872427.2882974.
[11] T. Hofmann, Probabilistic latent semantic indexing, in: Proceedings of the Twenty-second Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 1999, pp. 50–57.
[12] L. Hubert, P. Arabie, Comparing partitions, J. Classif. 2 (1) (1985) 193–218.
[13] T. Iwata, S. Watanabe, T. Yamada, N. Ueda, Topic tracking model for analyzing consumer purchase behavior., in: Proceedings of the 2009 International Joint Conference on Artificial Intelligence (IJCAI), 9, 2009, pp. 1427–1432.
[14] A.K. Jain, Data clustering: 50 years beyond k-means, Pattern Recognit. Lett. 31 (8) (2010) 651–666.
[15] I.T.Y. Li, Q. Yang, K. Wang, Classification pruning for web-request prediction, in: Proceedings of the Tenth World Wide Web Conference (WWW Posters), 2001.
[16] Y. Li, S.M. Chung, J.D. Holt, Text document clustering based on frequent word meaning sequences, Data Knowl. Eng. 64 (1) (2008) 381–404.
[17] S. Liang, E. Yilmaz, E. Kanoulas, Dynamic clustering of streaming short documents, in: B. Krishnapuram, M. Shah, A.J. Smola, C. Aggarwal, D. Shen, R. Rastogi (Eds.), Proceedings of the Twenty-second ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, San Francisco, CA, USA, August 13–17, 2016, pp. 995–1004, doi:10.1145/2939672.2939748.
[18] H. Ma, I. King, M.R. Lyu, Learning to recommend with social trust ensemble, in: Proceedings of the Thirty-second international ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2009, pp. 203–210.
[19] H. Ma, D. Zhou, C. Liu, M.R. Lyu, I. King, Recommender systems with social regularization, in: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, ACM, 2011, pp. 287–296.
[20] C.D. Manning, P. Raghavan, H. Schütze, Scoring, term weighting and the vector space model, Introd. Inf. Retr. 100 (2008) 2–4.
[21] T.P. Minka, Estimating a Dirichlet distribution. Technical Report, (2000; revised 2003, 2009, 2012).
[22] B. Mobasher, R. Cooley, J. Srivastava, Creating adaptive web sites through usage-based clustering of urls, in: Proceedings of the 1999 Workshop on Knowledge and Data Engineering Exchange, (KDEX'99), IEEE, 1999, pp. 19–25.
[23] M. Pennacchiotti, F. Silvestri, H. Vahabi, R. Venturini, Making your interests follow you on Twitter, in: Proceedings of the Twenty-first ACM International Conference on Information and Knowledge Management, ACM, 2012, pp. 165–174.
[24] A. Rangrej, S. Kulkarni, A.V. Tendulkar, Comparative study of clustering techniques for short text documents, in: Proceedings of the Twentieth International Conference Companion on World Wide Web, ACM, 2011, pp. 111–112.
[25] Z. Ren, S. Liang, E. Meij, M. de Rijke, Personalized time-aware tweets summarization, in: Proceedings of the Thirty-sixth International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2013, pp. 513–522.
[26] M. Rosen-Zvi, T. Griffiths, M. Steyvers, P. Smyth, The author-topic model for authors and documents, in: Proceedings of the Twentieth Conference on Uncertainty in Artificial Intelligence, AUAI Press, 2004, pp. 487–494.
[27] H. Schütze, Introduction to information retrieval, in: Proceedings of the 2008 International Communication of Association for Computing Machinery Conference, 2008.
[28] J. Srivastava, R. Cooley, M. Deshpande, P.-N. Tan, Web usage mining: discovery and applications of usage patterns from web data, ACM SIGKDD Explor. Newslett. 1 (2) (2000) 12–23.
[29] X. Wang, A. McCallum, Topics over time: a non-Markov continuous-time model of topical trends, in: Proceedings of the Twelfth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2006, pp. 424–433.
[30] X. Wei, W.B. Croft, Lda-based document models for ad-hoc retrieval, in: Proceedings of the Twenty-ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2006, pp. 178–185.
[31] X. Wei, J. Sun, X. Wang, Dynamic mixture models for multiple time-series., in: Proceedings of the 2007 International Joint Conference on Artificial Intelligence (IJCAI), 7, 2007, pp. 2909–2914.
[32] Z. Xu, Y. Zhang, Y. Wu, Q. Yang, Modeling user posting behavior on social media, in: Proceedings of the Thirty-fifth International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2012, pp. 545–554.
[33] X. Yan, J. Guo, Y. Lan, X. Cheng, A biterm topic model for short texts, in: Proceedings of the Twenty-second International Conference on World Wide Web, ACM, 2013, pp. 1445–1456.
[34] S.-H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, H. Zha, Like like alike: joint friendship and interest propagation in social networks, in: Proceedings of the Twentieth International Conference on World Wide Web, ACM, 2011, pp. 537–546.
[35] M. Ye, X. Liu, W.-C. Lee, Exploring social influence for recommendation: a generative model approach, in: Proceedings of the Thirty-fifth International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, 2012, pp. 671–680.
[36] J. Yin, J. Wang, A Dirichlet multinomial mixture model-based approach for short text clustering, in: Proceedings of the Twentieth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2014, pp. 233–242.
[37] G. Yu, R. Huang, Z. Wang, Document clustering via dirichlet process mixture model with feature selection, in: Proceedings of the Sixteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2010, pp. 763–772.
[38] Y. Zhao, S. Liang, Z. Ren, J. Ma, E. Yilmaz, M. de Rijke, Explainable user clustering in short text streams, in: R. Perego, F. Sebastiani, J.A. Aslam, I. Ruthven, J. Zobel (Eds.), Proceedings of the Thirty-ninth International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2016, ACM, Pisa, Italy, July 17–21, 2016, pp. 155–164, doi:10.1145/2911451.2911522.
[39] B. Zimmer, C.E. Carson, Among the new words, Am. Speech 87 (4) (2012) 491–510.