Inferring Dynamic User Interests in Streams of Short Texts for User Clustering

SHANGSONG LIANG, University College London
ZHAOCHUN REN, Data Science Lab, JD.com
YUKUN ZHAO, Shandong University
JUN MA, Shandong University
EMINE YILMAZ, University College London
MAARTEN DE RIJKE, University of Amsterdam

User clustering has been studied from different angles. In order to identify shared interests, behavior-based methods consider similar browsing or search patterns of users, whereas content-based methods use information from the contents of the documents visited by the users. So far, content-based user clustering has mostly focused on static sets of relatively long documents. Given the dynamic nature of social media, there is a need to dynamically cluster users in the context of streams of short texts. User clustering in this setting is more challenging than in the case of long documents, as it is difficult to capture the users' dynamic topic distributions in sparse data settings. To address this problem, we propose a dynamic user clustering topic model (UCT). UCT adaptively tracks changes of each user's time-varying topic distributions based both on the short texts the user posts during a given time period and on previously estimated distributions. To infer changes, we propose a Gibbs sampling algorithm where a set of word pairs from each user is constructed for sampling. UCT can be used in two ways: (1) as a short-term dependency model that infers a user's current topic distribution based on the user's topic distributions during the previous time period only, and (2) as a long-term dependency model that infers a user's current topic distributions based on the user's topic distributions during multiple time periods in the past. The clustering results are explainable and human-understandable, in contrast to many other clustering algorithms. For evaluation purposes, we work with a dataset consisting of users and tweets from each user. Experimental results demonstrate the effectiveness of our proposed short-term and long-term dependency user clustering models compared to state-of-the-art baselines.

CCS Concepts: • **Information systems → Clustering;**

Additional Key Words and Phrases: Diversity, ad hoc retrieval, data streams

This research was supported by the UCL Big Data Institute, Elsevier, Natural Science Foundation of China (61272240, 61672322), Ahold Delhaize, Amsterdam Data Science, the Bloomberg Research Grant program, the Dutch national program COMMIT, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 312827 (VOX-Pol), the Microsoft Research Ph.D. program, the Netherlands Institute for Sound and Vision, the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.001.116, HOR-11-10, CI-14-25, 652.002.001, 612.001.551, 652.001.003, and Yandex. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

Authors' addresses: S. Liang, Department of Computer Science, University College London, UK; email: shangsong.liang@ucl.ac.uk; Z. Ren (corresponding author), Data Science Lab, JD.com, China; email: renzhaochun@jd.com; M. de Rijke, Informatics Institute, University of Amsterdam, The Netherlands; email: derijke@uva.nl; Y. Zhao and J. Ma, Department of Computer Science, Shandong University, China; emails: yukun.zhao.sdu@gmail.com, majun@sdu.edu.cn; E. Yilmaz, Department of Computer Science, University College London, UK; and The Alan Turing Institute, UK; email: emaine.yilmazAucl.ac.uk.

DOI: http://dx.doi.org/10.1145/3072606

ACM Reference Format:
Shangsong Liang, Zhaochun Ren, Yukun Zhao, Jun Ma, Emine Yilmaz, and Maarten de Rijke. 2017. Inferring dynamic user interests in streams of short texts for user clustering. ACM Trans. Inf. Syst. 36, 1, Article 10 (July 2017), 37 pages.
DOI: http://dx.doi.org/10.1145/3072606

1. INTRODUCTION

Microblogging services such as Twitter[1] and Sina Weibo[2] gained popularity for communicating and distributing ideas, news content, and advertisements [12, 36, 44]. Through short documents, users can express their dynamic interests in real time. A good understanding of users' dynamic preferences is important for the design of applications that cater for users of such microblogging services, such as personalized microblog search [72], twitter summarization [61], and semantic entity linking for microblogs [49]. In this article, we study the problem of *user clustering in the context of streams of short texts*. Here, users are understood to be people who post messages on a microblogging platform. Our goal is to infer users' topic distributions over time and dynamically cluster users based on their topic distributions in such a way that users in the same cluster share similar interests while users in different clusters differ in their interests. In addition, we aim at making the clustering results explainable and understandable.

Previous work on user clustering [13, 39, 52] mainly clusters users who exhibit similar patterns when accessing information such as clicked documents. For instance, the user clustering method proposed by Mobasher et al. [52] constructs vector matrices for URLs and users and then utilizes K-means [46] to cluster users based on browsing vectors. Such methods are designed to work with collections of static, long documents and they often make the assumption that users' interests do not change over time. Unlike previous work, we focus on clustering users at a certain point in time, in the context of streams of short documents.

Accordingly, we propose a dynamic multinomial Dirichlet mixture user clustering topic model (UCT) to tackle the problem of dynamic user clustering in streams of short texts. UCT models time-varying topic distributions using a short-term or long-term dependency model over sequential short texts posted by users at different points in time. The *short-term* dependency UCT model infers a user's topic distributions at time T, i.e., the current topic distributions, based on the topic distributions at time period $T - 1$ and the content of newly arriving short documents at time T. In contrast, the *long-term* dependency UCT model infers a user's current topic distributions based on their topic distributions at time $(T-1), (T-2), \ldots, (T-L)$, plus the content of newly arriving short documents. Here, L is the length of the history that we want to consider for the inference of the current topic distributions. Obviously, the short-term dependency UCT model is a special case of the long-term dependency UCT model if we set $L = 1$ in the long-term dependency UCT model.

Traditional topic models such as probabilistic latent semantic indexing [28, PLSI], latent Dirichlet allocation [8, LDA], author topic models [63, 76], or the user interest topic model [40, 43], have been widely used to uncover topics of documents and users. However, these topic models ignore time information underlying the documents and only work well in static corpora. In contrast, dynamic topic models such as dynamic topic model [7, DTM], dynamic mixture model [75, DMM], and topic tracking model [31, TTM] work in the context of long document streams. These dynamic topic models, however, do not work well in the context of streams of short texts due to the problem

[1] http://www.twitter.com.
[2] http://www.weibo.com.

of sparsity. How to infer users' dynamic topic distributions for user clustering in the context of short document streams is still an open problem.

Inspired by previous work [13, 14, 41, 42, 44, 59, 61, 77], to alleviate the sparsity problem, in our UCT model, we extract word pairs in each tweet and form a word pair set for each user to explicitly capture word co-occurrence patterns. That is, UCT infers each user's interests with hidden topics while topics are captured from the word pair set of the users. In addition, to track the dynamics of a user's interests, UCT infers a user's current interests by integrating the interests at previous time periods, either short-term or long-term, with newly observed data in text streams. It then utilizes users' current interests for clustering. Thus, the result of user clustering is time-varying and users in the same cluster share similar interests at the current time, although their interests may differ at previous times. To the best of knowledge, we are the first to perform dynamic user clustering in streams of short texts based on the distributions of users' interests during a given time period.

Our main research questions are whether UCT outperforms state-of-the-art user clustering methods, whether the long-term dependency UCT model outperforms the short-term dependency UCT model, and whether our clustering results are explainable and understandable in contrast to results of other methods. We conduct our experiments on a Twitter dataset and demonstrate the effectiveness of UCT.

The main contributions of our work can be summarized as:

(1) We propose the task of dynamically clustering users in the context of streams of short texts.
(2) We propose a dynamic multinomial Dirichlet mixture UCT model to address the user clustering task, where users' time-varying topic distributions can be captured in the context of streams of short texts.
(3) We propose a collapsed Gibbs sampling algorithm for the inference of dynamic users' topic distributions in the context of streams of short texts, where we tackle the problem of word co-occurrence sparsity.
(4) UCT models temporal dynamics using either short-term or long-term dependencies over sequential data. The short-term dependency UCT model infers a user's dynamic interests based on the content of short documents arriving at the current time and topic distributions inferred over the immediately preceding time period. The long-term dependency UCT model infers a user's dynamic interests based on the content of short documents arriving at the current time and the user's topic distributions over multiple time periods in the past.
(5) Our proposed clustering model can effectively cluster previously seen users as well as users who just newly arrive in the streams.
(6) We provide a thorough analysis of UCT and of the impact of its key ingredients and parameters and find that it significantly outperforms state-of-the-art algorithms in terms of both clustering and topic modeling oriented evaluation metrics that capture different evaluation criteria.

The remainder of the article is organized as follows: Section 2 discusses related work; Section 3 details the problem; Section 4 describes the proposed model for user clustering in streams; Section 5 describes our experimental setup; Section 6 is devoted to our experimental results, and we conclude the article in Section 7.

2. RELATED WORK

Three major types of research relate to our work: user clustering, text clustering, and topic modeling.

2.1. User Clustering

State-of-the-art research on user clustering mainly focuses on web user clustering [10, 39, 52, 66]. These papers study users' access information from logged server data including query and click data, and then uncover clusters of these users that exhibit similar information needs. For instance, Buscher et al. [10] cluster users based on user interaction information, including clicks, scrolls, and cursor movements for search queries on long text documents. Another line of work, which mostly focuses on content-based similarity, groups users by expertise [5, 27]. Recent advances in distributed representation learning have given rise to new types of joint topic and entity representations [70, 71], but, so far, these have not been used for user clustering yet.

With the arise of social media, user clustering on social media [11, 36] has attracted lots of attention. Cha et al. [11] investigate the dynamics of user influence across topics and time based on an in-depth comparison of three measures of influence: in-degree, retweets, and mentions. Sakaki et al. [64] consider each Twitter user as a sensor and apply Kalman filtering and particle filtering. Personalized tweets ranking has also been proposed by leveraging the use of retweet behavior [69]. User classification on social text streams has also been addressed recently: Pennacchiotti and Popescu [55] infer the values of user attributes such as political orientation or ethnicity by leveraging observable information such as the user behavior, network structure, and the linguistic content of the user. Al Zamal et al. [2] evaluate the extent to which features present in a Twitter user and her immediate neighbors can improve the inference of attributes possessed by the user herself. Community detection is another relevant topic based on user clustering [37, 48]. Mislove et al. [51] propose a community detection method to infer the attributes of the remaining users given attributes for some fraction of the users in an online community. McAuley and Leskovec [48] propose the node clustering problem on a user's ego-network, and develop a model for detecting circles that combines network structure as well as user profile information.

To the best of our knowledge, existing content-based user clustering algorithms work with long documents and do not consider clustering users in the context of streams of short texts such as Twitter or Weibo. In this article, we aim at inferring users' dynamic interests and dynamically clustering them in streams of short texts. This article extends our previous work [83], in which we propose a dynamic clustering method for user clustering in the context of streams of short texts. Our earlier article only focuses on short term histories of users when inferring a user's current interests for clustering users. In this extension, we incorporate long term histories of users when inferring their current dynamic interests for clustering.

2.2. Text Clustering

Another relevant line of work is text clustering. Yu et al. [80] and Huang et al. [29] propose a Dirichlet process mixture with feature selection model (DPMFS) and a Dirichlet process mixture with feature partition model (DPMFP) for long document clustering, respectively. They compare DPMFP with four other clustering models: EM (Expectation-Maximization) text classification [53, EM-TC], K-means [32, 46], LDA [8], and exponential-family approximation of the Dirichlet compound multinomial distribution [18, EDCM]; they find that DPMFP performs best. Using entity linking and knowledge graph representations, clustering of entities has received an increase of attention [23, 57]. Green et al. [23] consider clustering entity mentions across languages without *a priori* knowledge of the quantity or types of entities from a knowledge base.

In the context of short text documents, Rangrej et al. [58] compare three clustering algorithms including K-means, Singular Value Decomposition, and Affinity Propagation [19] on a small set of tweets and find that Affinity Propagation outperforms the other two, but the complexity of Affinity Propagation is quadratic in the number of documents. He et al. [26] propose a co-regularized non-negative matrix factorization model for clustering user comments. Tsur et al. [68], Yin [78], and Yu et al. [80] focus on the problem of online clustering of a stream of tweets. All of those methods use an incremental clustering framework that first groups a number of tweets into clusters, then assigns the newly arriving tweets to these clusters. Yin and Wang [79] introduce a collapsed Gibbs sampling algorithm for the Dirichlet multinomial mixture model for short text clustering in a static set of short documents. They do not model documents with a distribution of topics. Instead, they assign a single topic to each document, then cluster the documents based on the topic assignments. Liang et al. [45] consider the problem of dynamically clustering a streaming corpus of short documents by proposing a new clustering topic model to effectively handle the dynamic nature of topics across time. All of these algorithms aim at clustering short documents—the problem of dynamically clustering users in the context of streams of short texts has so far been ignored, however.

2.3. Topic Modeling

Probabilistic topic models, such as PLSI [28] and LDA [8], aim to analyze latent topics of documents by using latent topic distribution to represent each document. Various LDA-type topic models have been proposed. The author topic model [63] has been proposed to uncover latent topics of authors; each author is associated with a multinomial distribution over topics and each topic is associated with a multinomial distribution over words. This suggests a method for clustering users in streams of short texts: model users as distributions over topics inferred from their tweets and then cluster users based on their topic distributions. The entity-topic model detects and links an entity to a latent topic in a document [25]. However, for data with topic evolution, the underlying “bag of words” representation may be insufficient. To analyze topic evolution, other models have been proposed, such as the Dynamic Topic Model [7], Dynamic Mixture Models [75], and the Topic Tracking Model [31]. Topic models have not yet been considered very frequently in the setting of Twitter. Twitter-LDA is an interesting exception; it classifies latent topics into “background” topics and “personal” topics [82], while an extension of Twitter-LDA has been proved to be effective in burst detection [17]. Topic models have been extended to sentiment analysis task successfully. For instance, Paul et al. [54] propose a topic model to distinguish topics into two contrastive categories; and Li et al. [38] propose a sentiment-dependency LDA model by considering dependency between adjacent words.

Various dynamic topic models have been proposed to track changes of topics in streams. The DTM [7] analyzes the time evolution of topics in document collections, in which a document is assumed to have one timestamp. Since DTM uses a Gaussian distribution for the dynamics, the inference is intractable because of the non-conjugacy of the Gaussian and multinomial distributions. The DMM [75] considers a single dynamic sequence of documents, which corresponds to a single topic over time. The TTMI [31] focuses on tracking time-varying consumer behavior, in which consumers’ interests change over time. The topic over time model [74, ToT] assumes that each topic is associated with a continuous distribution over timestamps, and the topic distribution of a document is influenced by both word co-occurrences and the document’s timestamp. The distributed author-topic over time [76, D-AToT] topic model combines the merits of author topic model [63] and ToT model. Specifically, it can automatically detect latent topics, users’ interests, and their changing patterns from large-scale social network information. Gerrish and Blei [21] propose a dynamic topic model for quantifying and qualifying the impact of documents within the collection and use the changes in the thematic content of documents over time to measure the importance of the documents.

Ren et al. [60] propose a dynamic topic model to monitor viewpoints on social media. All of these models assume that the context of the documents is rich enough to infer a topic distribution for the documents, which may not work well for documents in streams of short texts.

Exploiting external knowledge to enrich the representation of short texts has been proposed to improve the performance of topic modeling for short texts. Phan et al. [56] train latent topics from large external resources. Jin et al. [33] learn hidden topics on short texts via transfer learning from auxiliary long text data. Ren et al. [62] apply a document expansion method that consists of entity linking and sentence extraction. Chen and Liu [14] retain the results learned in the past and use them to help future learning. Yan et al. [77] extract bi-terms in each tweet to capture word co-occurrence explicitly for enhancing the performance of short text topic modeling. Again, unlike our UCT, these algorithms aim at working with a static collection of documents only.

Recently, non-parametric topic models, such as the hierarchical Dirichlet process [67, HDP], have seen an increase in attention. Non-parametric topic models are aimed at handling infinitely many topics. For instance, to capture the relationship between latent topics, nested Chinese restaurant processes generate tree-like topical structures over documents [6]. To describe the whole life cycle of a topic, Ahmed and Xing [1] propose an infinite dynamic topic model on temporal documents. Instead of assuming that a vocabulary is known *a priori*, Zhai and Boyd-Graber [81] propose an extension of the Dirichlet process to add and delete terms over time. Non-parametric topic models have also been applied to explore personalized topics and time-aware events in social text streams [16]. Traditional non-parametric topic models do not explicitly address diversification among latent variables during clustering. To tackle this issue, Kulesza and Taskar [34, 35] propose a stochastic process named structured determinantal point process (SDPP), where diversity is explicitly considered. As an application in text mining, Gillenwater et al. [22] propose a method for topic modeling based on SDPPs. As far as we know, the determinantal point process has not been integrated with other non-parametric models yet.

We work with streams of short texts and propose a dynamic Dirichlet multinomial mixture UCT model, either short-term or long-term dependency, by which we capture a multinomial distribution of topics specific to each user over time in Twitter and then dynamically cluster users based on their dynamic topic distributions. To enhance the performance of the inference in our proposed Gibbs sampling for our topic model, we extract word pairs in tweets and form a word pair set for each user to explicitly capture word co-occurrence patterns. To the best of our knowledge, this is the first attempt to use a topic model to infer dynamic user interests in the context of streams of short texts for user clustering.

3. NOTATION AND TASK

In this section, we introduce the main notation used in the article as well as the task that we address.

Table I summarizes our main notation. Term $u \in \mathbf{U}_t$ indicates a user, while $\mathbf{U}_t = \{u_1, u_2, \ldots, u_m\}$ is a set of users at time period $t \in \{\ldots, (T-1), T\}$ with T being the most recent time period, and the length of each time period t can be a week, a month, a quarter, half a year, and a year. Also, z is a topic and K is the number of topics we infer in our UCT model; w is a word in a tweet and b represents an unordered word pair (w_i, w_j) extracted from a tweet.

We extract a set of word pairs $\mathbf{B}_{t,u}$ for each user u from their published tweets $\mathbf{D}_{t,u}$ at time period t, and we aggregate all users' word pair sets as $\mathbf{B}_t$. We use $\mathbf{B}_t$ as input to monitor each user's interest in the UCT model. The parameters α_t and β_t are Dirichlet priors for our topic model at time t. $z_{t,u,b}$, $m_{t,u,z}$, and $n_{t,z,w}$, which are used in the topic

Table I. Main Notation used in the Article

Notation	Gloss	Notation	Gloss
u	user	t	time slice
z	topic	K	number of topics
w	word	b	word pair
L	length of terms in long-term dependency UCT		
$\boldsymbol{\omega}_{t,u}$	cluster user u belonging to at time t		
$\mathbf{U}_t$	a set of users at time t		
$\mathbf{D}_t$	a text stream at time t		
$\mathbf{D}_{t,u}$	texts published by user u at time t		
$\mathbf{B}_{t,u}$	a set of word pairs published by user u at time t		
$\mathbf{B}_t$	a set of word pairs published at time t		
V_t	total number of distinct words in document stream $\mathbf{D}_t$		
α_t	the parameter of user topic distribution Dirichlet prior		
β_t	the parameter of token topic distribution Dirichlet prior		
$\theta_{t,u}$	multinomial distribution of topics specific to user u at time t		
$\phi_{t,z}$	multinomial distribution of words specific to topic z at time t		
$z_{t,u,b}$	topic assignment on b for user u at time t		
$m_{t,u,z}$	number of word pairs published by u assigned to topic z at time t		
$n_{t,z,w}$	number of times word w is assigned to topic z at time t		

model training process at time t, represent the topic assignment on word pair b for user u, the number of word pairs published by user u assigned to topic z, and the number of times w is assigned to topic z at time t, respectively. $\boldsymbol{\omega}_{t,u}$ is a cluster to which user u belongs at time t, and the cluster $\boldsymbol{\omega}_{t,u}$ can be changed over time as the user's interest $\theta_{t,u} = \{\theta_{t,u,z}\}_{z=1}^{K}$ is time-varying in streams.

The task we address is to dynamically track users' interests and cluster them over time in the context of streams of short texts such that users in the same cluster share similar interests. Specifically, for each time period t, given a set of users $\mathbf{U}_t = \{u_1, u_2, \dots, u_{|\mathbf{U}_t|}\}$ at time t with $|\mathbf{U}_t|$ being the number of users in $\mathbf{U}_t$ and a short text stream $\mathbf{D}_t$ up to t, we focus on uncovering the clusters of users in $\mathbf{U}_t$, with $\boldsymbol{\omega}_{t,u}$ being the cluster to which user u belongs at t.

4. METHOD

We start by providing an overview of our method in Section 4.1. We then detail each of the three main steps of the proposed user clustering method: preprocessing in Section 4.2, UCT model in Section 4.3, and user clustering in Section 4.4.

4.1. Overview

We use Twitter as our default setting of streams of short texts and provide a general scenario of our method for dynamically clustering users in tweet streams in Algorithm 1. We assume that each user's interest is represented by topics, and each user's interests may drift over time. Formally, given a time period $t \in \{\dots, (T-1), T\}$, the interest of each user $u \in \mathbf{U}_t$ is represented as a multinomial distribution $\theta_{t,u}$ over topics. The distribution $\theta_{t,u}$ is inferred from our proposed dynamic user topic model. Because documents in streams of short texts are short and sparse, we propose a preprocessing step to extract word pairs (see step 1 in Algorithm 1), where a word pair contains two words sharing the same topic. We enrich the context by considering co-occurring words in word pairs instead of documents.

Next, we propose a dynamic Dirichlet multinomial mixture UCT model to capture each user's dynamic interests $\theta_{t,u}$, at time slice t, in the context of streams of short texts (see step 2 in Algorithm 1). Each user's interests $\theta_{t,u}$ are computed after the sampling

ALGORITHM 1: Overview of the Algorithm for user Clustering in Streams of Short Texts.

```
Input: A set of users $\mathbf{U}_t$ along with their published tweets $\mathbf{D}_t$
Output: cluster of each user $\boldsymbol{\omega}_{t,u}$
1 Extract a set of word pairs $\mathbf{B}_{t,u}$ for each user $u$, see Section 4.2
2 Use UCT model, either short-term or long-term dependency, to track each user's interests
  at time $t$ as $\theta_{t,u}$, see Section 4.3
3 Cluster users based on their interests distribution $\theta_{t,u}$ at time $t$, see Section 4.4
```

process has finished. Our UCT can be either a short-term dependency model that infers users' current topic distributions based on only the users' topic distributions at the previous step or a long-term dependency one that infers users' current topic distributions based on users' topic distributions at the previous long steps.

Based on the multinomial distribution $\theta_{t,u}$, we cluster users using K-means clustering [32, 46] (see step 1 in Algorithm 1). With the time period t moving forward, the result of clustering users changes dynamically.

4.2. Extracting Word Pairs

Traditional topic models [8, 31, 74] detect topics from a document based on word co-occurrences in documents. The topics are represented as groups of correlated words, while the correlation is revealed by word co-occurrence patterns in documents. In this article, we do not directly use the words in tweets (our documents) to directly infer topics for users due to the limited length of tweets.

Since topics are basically groups of correlated words and the correlation is revealed by word co-occurrence patterns in documents [15], in order to tackle the lack of context in modeling users' interests and the short, ambiguous nature of documents in streams, we explicitly consider word correlations via co-occurring words in a word pair instead of each individual word in a whole tweet, where a word pair is a set of two (order-exchangeable) words assigned to the same topic. Specifically, after removing stop words and applying Porter stemming, we obtain each user's tweet set $\mathbf{D}_{t,u}$ (the tweets user u published at the current time period t). Following [14, 77], we regard each tweet as an individual context unit, in which word pairs in a tweet can be assigned to different topics but the two words in a word pair share the same topic. Then, the method to extract word pairs from each tweet $d \in \mathbf{D}_{t,u}$ is as follows:

$$\mathbf{B}_d = \{(w_i, w_j) \mid w_i, w_j \in d, i \neq j\}. \tag{1}$$

Each word pair $b \in \mathbf{B}_d$ contains two different unordered words (w_i, w_j) in tweet d. For example, from the tweet "bananas and apples are all fruit" we extract three word pairs, i.e., "banana apple", "banana fruit" and "apple fruit" after removing stop words and stemming. Then, we aggregate all word pairs extracted from tweets $\mathbf{D}_{t,u}$ for user u:

$$\mathbf{B}_{t,u} = \bigcup_{d \in \mathbf{D}_{t,u}} \mathbf{B}_d. \tag{2}$$

Thus, for each user u, the set $\mathbf{D}_{t,u}$ of their published tweets at time period t is processed to a set of word pairs $\mathbf{B}_{t,u}$.

We sample the topic assignment z for each word pair instead of each independent word. In other words, the word correlation constructed to infer topics does not rely on the co-occurrence in tweets but in word pairs. The next section shows how to use the word pair set $\mathbf{B}_{t,u}$ to model the user's interests.

4.3. Dynamic UCT Model

In this section, we first provide the preliminaries of the proposed models in Section 4.3.1 and then detail our dynamic multinomial Dirichlet mixture UCT model. Specifically, we propose our short-term dependency UCT in Section 4.3.2 and our long-term dependency UCT in Section 4.3.3.

4.3.1. Preliminaries. UCT aims at capturing a user's interests at time t as $\theta_{t,u}$, i.e., a multinomial distribution over mixtures of topics. We use $t \in \{\ldots, (T-1), T\}$ to represent a time period, the length of which can be a day, a week, a month, a quarter, or even a year. In a fully Bayesian non-dynamic topic model such as LDA [8], there is an underlying assumption that the current topic distributions are independent of the past distributions and have a Dirichlet prior with a static set of parameters $\kappa = \{\kappa_z\}_{z=1}^{K}$, with $\kappa_z > 0$. With this assumption, if we do not consider the past distribution for the current topic distribution in a user clustering scenario, we have:

$$P(\theta_{t,u}|\kappa) \propto \prod_{z=1}^{K} \theta_{t,u,z}^{\kappa_z - 1}. \tag{3}$$

Similarly, the per-topic word distribution $\phi_{t,z} = \{\phi_{t,z,v}\}_{v=1}^{V_t}$ also has a Dirichlet prior with a static set of parameters $\gamma = \{\gamma_v\}_{v=1}^{V_t}$, with $\gamma_v > 0$ and V_t being the total number of distinct words in the document stream $\mathbf{D}_t$:

$$P(\phi_{t,z}|\gamma) \propto \sum_{v=1}^{V_t} \phi_{t,z,v}^{\gamma_v - 1}. \tag{4}$$

The assumptions made in Equations (3) and (4) are not realistic in a streaming setting, where the distributions at time t are dependent on past distributions. In the following sections, we infer $\theta_{t,u}$ and $\phi_{t,z}$ by short-term dependency and long-term dependency UCT models, respectively.

4.3.2. Short-term Dependency UCT. To model the temporal dependencies of the topics and the issue of short documents in streams, and following work of past dynamic topic models [7, 31, 74], we propose a *short-term-dependency* UCT model, in which the topic distribution at time t is the same as the one at previous time $t-1$ if no newly arriving short documents are observed at the current time t; otherwise, it is updated by the newly arriving short documents observed at time t based on the one at $t-1$.

Figure 1 shows the graphical representation of our short-term dependency UCT model, where shaded and unshaded nodes indicate observed and latent variables, respectively. At time t, we sample word pairs $(w_i, w_j) \in \mathbf{B}_{t,u}$ for users $u \in \mathbf{U}_t$ based on the current topic-word distributions $\mathbf{\Phi}_t$, and infer current users' interests $\mathbf{\Theta}_t$. From Figure 1, we see that a dependency is assumed to exist between two adjacent time periods.

We track the dynamics of a user's interests based on the assumption that a user's interests during the current time period t are the same as those during the preceding time period $t-1$ unless interests are changed by newly observed data at t. We infer a user's current interests by combining their topic distribution obtained during the previous time slice $t-1$ as prior knowledge and newly arriving observed data. In particular, we use both the user's previous interests $\theta_{t-1,u}$ and current Dirichlet prior α_t as a new Dirichlet prior. Then, the K-dimensional variable $\theta_{t,u} = \{\theta_{t,u,z}\}_{z=1}^{K}$ has the

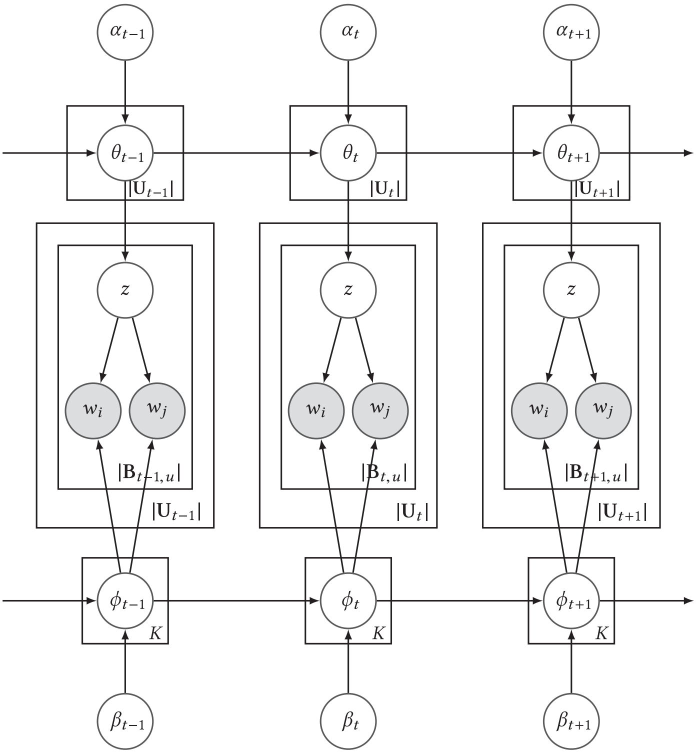


Fig. 1. Graphical representation of our dynamic UCT model. Shaded nodes represent observed variables, while unshaded ones represent latent variables. Note that the current distributions only depend on the previous timestep distributions, e.g., the distributions θ_{t+1} and ϕ_{t+1} only depend on θ_t and ϕ_t, respectively.

following probability density:

$$P(\theta_{t,u} \mid \theta_{t-1,u}, \alpha_t) = \frac{\Gamma(\sum_z \theta_{t-1,u,z} + \alpha_{t,z})}{\prod_z \Gamma(\theta_{t-1,u,z} + \alpha_{t,z})} \cdot \prod_{z=1}^{K} \theta_{t,u,z}^{\theta_{t-1,u,z}+\alpha_{t,z}-1}, \tag{5}$$

where $\Gamma(x)$ is a Gamma function. In contrast with static topic models [8, 63], the Dirichlet prior changes from κ in Equation (3) to $\theta_{t-1,u} + \alpha_t$ in Equation (5), where the added term $\theta_{t-1,u}$ represents the influence of previously inferred interests.

To model the dynamics of topics over words, we infer topic-word distributions $\phi_{t,z} = \{\phi_{t,z,v}\}_{v=1}^{V_t}$ at the current time period t by using the following Dirichlet distribution:

$$P(\phi_{t,z} \mid \phi_{t-1,z}, \beta_t) = \frac{\Gamma(\sum_v \phi_{t-1,z,v} + \beta_{t,v})}{\prod_v \Gamma(\phi_{t-1,z,v} + \beta_{t,v})} \cdot \prod_{v=1}^{V_t} \phi_{t,z,v}^{\phi_{t-1,z,v}+\beta_{t,v}-1}. \tag{6}$$

The topic-words distributions in the short-term dependency UCT model are inferred through priors $\phi_{t-1,z}$ and β_t. We estimate α_t and β_t for each time period instead of simply using symmetric priors. Given all users' word pairs set $\mathbf{B}_t = \bigcup_{u \in \mathbf{U}_t} \mathbf{B}_{t,u}$, where $\mathbf{B}_{t,u}$ is the set of word pairs specific to user u, from Figure 1, we know that topic z is related with a distribution of words with the multinomial distribution $\phi_{t,z} = \{\phi_{t,z,v} \mid v \in \mathbf{V}_t\}$. In UCT, the multinomial distribution specific to the user u is used to select a topic; thereafter, a word in a word pair is generated according to the distribution $\phi_{t,z}$ specific to that chosen topic z. According to the graphical model, we sample each topic $z_{t,u,b}$ for each word pair $b \in \mathbf{B}_{t,u}$. The distributions, $\mathbf{\Theta}_{t-1}$ and $\mathbf{\Phi}_{t-1}$, at the previous time period $t-1$ and the two priors, β_t and α_t, that are obtained via fixed-point iterations (see Equation (10)) are utilized for inferring the current distributions $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$. Assuming that we know the topic distribution at time period $t-1$, $\mathbf{\Theta}_{t-1}$, and the word distribution over topics at time period $t-1$, $\mathbf{\Phi}_{t-1}$, the proposed short-term dependency UCT model is a generative model that depends on $\mathbf{\Theta}_{t-1}$ and $\mathbf{\Phi}_{t-1}$. At time period $t = 0$, we initialize the means of the two distributions to $\theta_{0,z} = 1/K$ and $\phi_{0,z,v} = 1/V$. The generative process is as follows:

(i) Draw K multinomials $\phi_{t,z}$ from Dirichlet priors β_t and $\mathbf{\Phi}_{t-1}$, one for each topic z;
(ii) For each user u, draw a multinomial distribution $\theta_{t,u}$ from Dirichlet priors α_t and $\theta_{t-1,u}$; then, for each word pair b in the word pairs set $b \in B_{t,u}$:
   (a) Draw a topic $z_{t,u,b}$ from multinomial $\theta_{t,u}$;
   (b) Draw a word $w_i \in b$ from multinomial $\phi_{t,z_{t,u,b}}$;
   (c) Draw another word $w_j \in b$ from multinomial $\phi_{t,z_{t,u,b}}$.

The parameterization of the proposed UCT topic model is as follows:

$$\begin{aligned} \phi_{t,z} \mid \mathbf{\Phi}_{t-1} + \beta_t &\sim \text{Dirichlet}(\mathbf{\Phi}_{t-1} + \beta_t) \\ \theta_{t,u} \mid \theta_{t-1,u} + \alpha_t &\sim \text{Dirichlet}(\theta_{t-1,u} + \alpha_t) \\ z_{t,u,b} \mid \theta_{t,u} &\sim \text{Multinomial}(\theta_{t,u}) \\ w_i \in b \mid \phi_{t,z_{t,u,b}} &\sim \text{Multinomial}(\phi_{t,z_{t,u,b}}) \\ w_j \in b \mid \phi_{t,z_{t,u,b}} &\sim \text{Multinomial}(\phi_{t,z_{t,u,b}}) \end{aligned}$$

We sample word pairs instead of words as shown in the above generative process. Then, the probability of generating a word pair $b = (w_i, w_j)$ given a topic z at t is represented as:

$$P(b \mid t, z) = P(w_i \mid t, z) P(w_j \mid t, z), \tag{7}$$

and the probability of generating a word pair b at t is represented as:

$$P(b \mid t) = \sum_z P(z \mid t) P(w_i \mid t, z) P(w_j \mid t, z). \tag{8}$$

Inference is intractable in this short-term dependency UCT. Following the work of Blei et al. [8], Griffiths and Steyvers [24], Iwata et al. [31], Wei et al. [75], Yin and Wang [79], we propose a collapsed Gibbs sampling method to perform approximate inference. We adopt a conjugate prior (Dirichlet) for the multinomial distributions, and thus, we easily integrate out $\mathbf{\Phi}_t$ and $\mathbf{\Theta}_t$, analytically capturing the uncertainty associated with them. In this way, we facilitate the sampling, i.e., we need not sample $\mathbf{\Phi}_t$ and $\mathbf{\Theta}_t$ at all.

The proposed collapsed Gibbs sampling algorithm for the UCT model is shown in Algorithm 2 (recall that our main notation is shown in Table I). The input of our Gibbs sampling algorithm is $\mathbf{B}_t$ (which consists of a set of word pairs at time slice t for every user) and the output consists of all users' interest distributions over topics at the current time t. For the initialization of our Gibbs sampling, we randomly sample a topic $z = z_{t,u,b}$ from a multinomial distribution with parameter $1/K$ and assign it to each word pair $b \in \mathbf{B}_{t,u}$ and update $m_{t,u,z}$ and $n_{t,z,w}$ (to be defined below) accordingly.

In the Gibbs sampling procedure above at time slice t, we need to calculate the conditional distribution:

$$P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t),$$

where $\mathbf{Z}_{t,-(u,b)}$ represents all topics assignments except the current word pair b from user u. We begin with $P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t | \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$, i.e., the joint probability of the

ALGORITHM 2: Gibbs Sampling for Short-term Dependency UCT Model

```
**Input:** $K, N, t, \mathbf{B}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_{t-1}, \beta_{t-1}$
**Output:** multinomial parameter $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$
**Initialize** $m_{t,u,z}, n_{t,z,w}$ as zero and
**for** *each user* $u \in \mathbf{U}_t$ **do**
    **for** *each word pair* $b \in \mathbf{B}_{t,u}$ **do**
        sample a topic $z \sim$ Multinomial$(1/K)$ randomly:
        $z_{t,u,b} \leftarrow z$
        $m_{t,u,z} \leftarrow m_{t,u,z} + 1$
        while word pair $b$ contains two words $w_i$ and $w_j$
        $n_{t,z,w_i} \leftarrow n_{t,z,w_i} + 1$
        $n_{t,z,w_j} \leftarrow n_{t,z,w_j} + 1$
**Sampling Phase**
**for** *iteration* $= 1, \ldots, N$ **do**
    **for** *each user* $u \in \mathbf{U}_t$ **do**
        **for** *each word pair* $b \in \mathbf{B}_{t,u}$ **do**
            record the current topic, $z = z_{t,u,b}$
            $m_{t,u,z} \leftarrow m_{t,u,z} - 1$
            while word pair $b$ contains two words $w_i$ and $w_j$:
            $n_{t,z,w_i} \leftarrow n_{t,z,w_i} - 1$
            $n_{t,z,w_j} \leftarrow n_{t,z,w_j} - 1$
            draw $z_{t,u,b} = z$ from $P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$ for the word
              pair $b$; see Equation (9).
            update $m_{t,u,z}, n_{t,z,w_i}, n_{t,z,w_j}$ as:
            $m_{t,u,z} \leftarrow m_{t,u,z} + 1$
            $n_{t,z,w_i} \leftarrow n_{t,z,w_i} + 1$
            $n_{t,z,w_j} \leftarrow n_{t,z,w_j} + 1$
    obtain optimal $\alpha_t$ and $\beta_t$ with fixed-point iterations; see Equations (10) and (17) for
      short-term dependency model.
compute the distributions $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$ using Equation (11).
```

current word pair set $\mathbf{B}_t$, the topic assignments $\mathbf{Z}_t$, and the user set $\mathbf{U}_t$ given the previous distributions $\mathbf{\Phi}_{t-1}$ and $\mathbf{\Theta}_{t-1}$, and two Dirichlet priors α_t and β_t. The joint probability is as follows:

$$
\begin{aligned}
&P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t) \\
&\quad = \left(\prod_z \left(\frac{\Gamma(\sum_v (\Upsilon_b))}{\prod_v \Gamma(\Upsilon_b)} \frac{\prod_v \Gamma(\Upsilon_a)}{\Gamma(\sum_v \Upsilon_a)}\right)\right)^2 \times \prod_u \frac{\Gamma(\sum_z (\Upsilon_2))}{\prod_z \Gamma(\Upsilon_2)} \frac{\prod_z \Gamma(\Upsilon_1)}{\Gamma(\sum_z \Upsilon_1)},
\end{aligned}
$$

where Υ_1, Υ_2, Υ_a, and Υ_b are defined as:

$$
\begin{aligned}
\Upsilon_1 &= m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1, \quad & \Upsilon_2 &= \theta_{t-1,u,z} + \alpha_{t,z}, \\
\Upsilon_a &= n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v} - 1, \quad & \Upsilon_b &= \phi_{t-1,z,v} + \beta_{t,v}.
\end{aligned}
$$

Here, $m_{t,u,z}$ represents the number of word pairs published by user u and assigned to topic z at time t, and $n_{t,z,w}$ represents the number of times word w is assigned to topic z at time t. Details of how to obtain the joint probability are provided in Appendix A and B. Using the chain rule, we obtain the conditional probability conveniently as follows:

$$\begin{aligned} &P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t) \\ &\propto \frac{n_{t,z,w_i} + \phi_{t-1,z,w_i} + \beta_{t,w_i} - 1}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v}) - 1)} \\ &\quad \times \frac{n_{t,z,w_j} + \phi_{t-1,z,w_j} + \beta_{t,w_j} - 1}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v}) - 1)} \\ &\quad \times \frac{m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1}{\sum_{z=1}^{K}(m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z}) - 1}. \end{aligned} \tag{9}$$

A detailed derivation of Gibbs sampling for our proposed short-term dependency UCT model is provided in Appendix A. The two Dirichlet priors α_t and β_t are estimated by maximizing the joint distribution, $P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t|\mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$, in the sampling at each iteration. We use the following update rules in fixed-point iterations for obtaining these two Dirichlet priors:

$$\begin{aligned} \alpha_{t,z} &\leftarrow \frac{\alpha_{t,z} \sum_u (\Psi(\Upsilon_1) - \Psi(\Upsilon_2))}{\sum_u(\Psi(\sum_z \Upsilon_1) - \Psi(\sum_z \Upsilon_2))}, \\ \beta_{t,v} &\leftarrow \frac{\beta_{t,v} \sum_z (\Psi(\Upsilon_a) - \Psi(\Upsilon_b))}{\sum_z(\Psi(\sum_v \Upsilon_a) - \Psi(\sum_v \Upsilon_b))}, \end{aligned} \tag{10}$$

where $\Psi(x) = \frac{\partial \log \Gamma(x)}{x}$ is a Digamma function. The derivation of the update rules for $\alpha_{t,z}$ and $\beta_{t,v}$, and the two bounds used in deriving the updating rules can be found in Appendix B.

Once the Gibbs sampling has been done, with the fact that a Dirichlet distribution is conjugate to a multinomial distribution, we then conveniently infer the following distributions for $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$ in our short-term dependency UCT model, respectively:

$$\begin{aligned} \theta_{t,u,z} &= \frac{m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1}{\sum_{z=1}^{K}(m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z}) - 1}, \\ \phi_{t,z,v} &= \frac{n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v} - 1}{\sum_{v=1}^{V_t}(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v}) - 1}. \end{aligned} \tag{11}$$

4.3.3. Long-term Dependency UCT. So far we have modeled the most recent topic distribution $\theta_{t,u}$ and $\phi_{t,u}$ to be only dependent on the previous topic distributions, i.e., the topic distributions $\theta_{t-1,u}$ and $\phi_{t-1,u}$, respectively. Past research has shown that topic distributions may depend on longer histories [7, 31, 74]. Accordingly, we propose a *long-term dependency* UCT model for user clustering in the context of short document streams. Instead of inferring users' interests at time t based only on the immediately preceding topic distributions and the newly arriving short documents, our long-term dependency UCT model infers the topic distributions $\theta_{t,u}$ and $\phi_{t,u}$ at time t based on the topic distributions from multiple time-periods in the past, i.e., the topic distributions at times $(t-1), (t-2), \ldots, (t-L)$, as well as the newly arriving short documents. Here, L is the dependency length, i.e., the number of time periods under consideration for inferring the current topic distributions at time t. Obviously, the short-term dependency UCT model is a special case of the long-term dependency UCT model if we set $L = 1$ in the long-term dependency model. The main difference between the short-term and long-term dependency UCT models is the following. The short-term dependency UCT

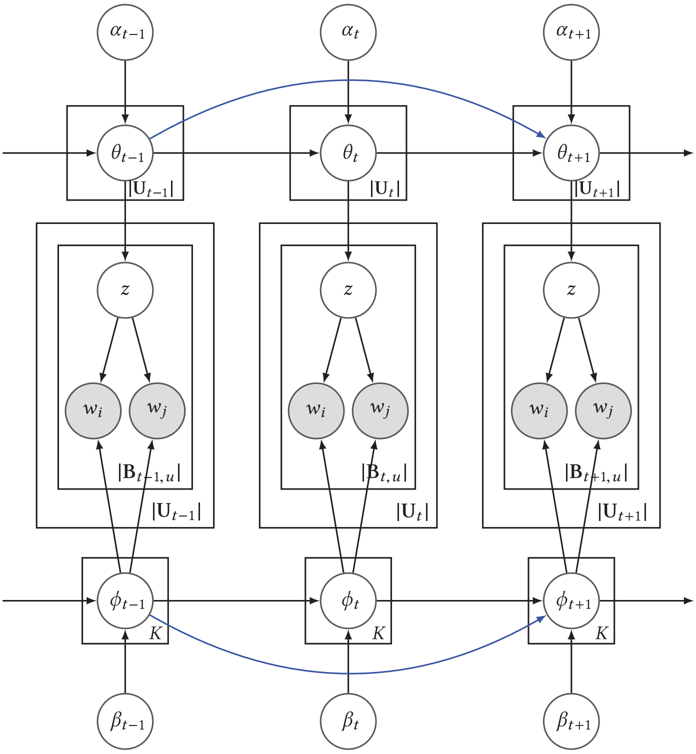


Fig. 2. Graphical representation of our long-term dependency dynamic UCT model. Shaded nodes represent observed variables, while unshaded ones represent latent variables. Note that compared to the graphical representation of short-term dependency UCT in Figure 1, the short-term dependency UCT model excludes the two blue curved lines while the long-term dependency UCT model does include them. The figure is for the long-term dependency UCT model with $L = 2$. The figure is best viewed in color.

infers topic distributions at time t based on the topic distributions at time period $t-1$ and the content of newly arriving short documents at time t, whereas the long-term dependency UCT model infers topic distributions at time t based not only on the topic distributions at time period $t-1$, but also on those at time periods $(t-2), \ldots, (t-L)$ and the content of the documents streaming in during these time periods.

Figure 2 shows a graphical representation of our long-term dependency UCT model, where shaded and unshaded nodes indicate observed and latent variables, respectively. Again, at time t, we sample word pairs $(w_i, w_j) \in \mathbf{B}_{t,u}$ for users $u \in \mathbf{U}_t$ based on the current topic-word distributions $\mathbf{\Phi}_t$, and infer current users' interests $\mathbf{\Theta}_t$. From Figure 2, we see that a dependency is assumed to exist between multiple adjacent time periods, e.g., the distribution $\mathbf{\Theta}_t$ not only depends on $\mathbf{\Theta}_{t-1}$ but also $\mathbf{\Theta}_{t-2}$ so that we consider a sequence of two time periods.

Accordingly, in our long-term dependency UCT model, we model long-term dependencies, using a sequence of L time periods, as follows:

$$P(\theta_{t,u} \mid \{\theta_{t-l,u}, \alpha_{t,l}\}_{l=1}^{L}) \propto \prod_{z=1}^{K} \theta_{t,u,z}^{(\sum_{l=1}^{L} \theta_{t-l,u,z} + \alpha_{t,z,l})-1}, \tag{12}$$

which can be further represented as:

$$P(\theta_{t,u} \mid \{\theta_{t-l,u}, \alpha_{t,l}\}_{l=1}^{L}) \\ = \frac{\Gamma\left(\sum_z \left(\sum_{l=1}^{L} \theta_{t-l,u,z} + \alpha_{t,z,l}\right)\right)}{\prod_z \Gamma\left(\sum_{l=1}^{L} \theta_{t-l,u,z} + \alpha_{t,z,l}\right)} \cdot \prod_{z=1}^{K} \theta_{t,u,z}^{(\sum_{l=1}^{L} \theta_{t-l,u,z}+\alpha_{t,z,l})-1}, \tag{13}$$

where L is the dependency length, i.e., the number of previous time periods under consideration for the inference of the current topic distributions.

That is, the mean of user u's topic distribution $\theta_{t,u}$ is proportional to the weighted sum of the past L "topic trends" in the short documents, and $\alpha_{t,l}$ represents how the topics at time t are related to the l-previous topics. See Equations (5) and (13) for a comparison between the short-term and the long-term dependency UCT models. In contrast to the short-term dependency UCT model, the long-term dependency UCT model reduces the information loss and the bias of the inference due to multiple estimates.

Similar to Equation (12), in the long-term dependency UCT model, the Dirichlet prior of the topic trends $\phi_{t,z}$ at the current time t can be revised such that $\phi_{t,z}$ depends on the past L topic trends $\{\phi_{t-l,z}\}_{l=1}^{L}$ as well. By doing so, we can make the inference more robust and thus have:

$$P(\phi_{t,z} \mid \{\phi_{t-l,z}, \beta_{t,l}\}_{l=1}^{L}) \propto \prod_{v=1}^{V_t} \phi_{t,z,v}^{(\sum_{l=1}^{L} \phi_{t-l,z,v}+\beta_{t,v,l})-1}, \tag{14}$$

which can be further represented as:

$$P(\phi_{t,z} \mid \{\phi_{t-l,z}, \beta_{t,l}\}_{l=1}^{L}) \\ = \frac{\Gamma\left(\sum_v \left(\sum_{l=1}^{L} \phi_{t-l,z,v} + \beta_{t,v,l}\right)\right)}{\prod_v \Gamma\left(\sum_{l=1}^{L} \phi_{t-l,z,v} + \beta_{t,v,l}\right)} \cdot \prod_{v=1}^{V_t} \phi_{t,z,v}^{(\sum_{l=1}^{L} \phi_{t-l,z,v}+\beta_{t,v,l})-1}. \tag{15}$$

The topic-words distributions in the long-term dependency UCT model are considered to be inferred through priors $\{\phi_{t-l,z}\}_{l=1}^{L}$ and $\{\beta_{t,l}\}_{l=1}^{L}$. Again, we estimate α_t and β_t for each time period instead of using simple symmetric priors. According to the graphical representation of the long-term dependency UCT model in Figure 2, we sample each topic $z_{t,u,b}$ for each word pair $b \in \mathbf{B}_{t,u}$. The distributions $\{\mathbf{\Theta}_{t-l}\}_{l=1}^{L}$, $\{\mathbf{\Phi}_{t-l}\}_{l=1}^{L}$ at the previous time periods and the priors $\{\beta_{t,l}\}_{l=1}^{L}$, $\{\alpha_{t,l}\}_{l=1}^{L}$ are utilized for inferring the current distributions $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$. The generative process in the long-term dependency UCT model is as follows:

(i) Draw K multinomials $\phi_{t,z}$ from Dirichlet priors $\{\beta_{t,l}\}_{l=1}^{L}$ and $\{\mathbf{\Phi}_{t-l}\}_{l=1}^{L}$, one for each topic z;
(ii) For each user u, draw a multinomial distribution $\theta_{t,u}$ from Dirichlet priors $\{\alpha_{t,l}\}_{l=1}^{L}$ and $\{\theta_{t-l,u}\}_{l=1}^{L}$; then for each word pair b in the word pairs set $b \in B_{t,u}$:
  (a) Draw a topic $z_{t,u,b}$ from multinomial $\theta_{t,u}$;
  (b) Draw a word $w_i \in b$ from multinomial $\phi_{t,z_{t,u,b}}$;
  (c) Draw another word $w_j \in b$ from multinomial $\phi_{t,z_{t,u,b}}$.

The parameterization of the proposed long-term dependency UCT topic model is as follows:

$$\begin{aligned}
\phi_{t,z} \left| \sum_{l=1}^{L} \mathbf{\Phi}_{t-l} + \beta_{t,l} \right. &\sim \text{Dirichlet}\left(\sum_{l=1}^{L} \mathbf{\Phi}_{t-l} + \beta_{t,l}\right)\\
\theta_{t,u} \left| \sum_{l=1}^{L} \theta_{t-l,u} + \alpha_{t,l} \right. &\sim \text{Dirichlet}\left(\sum_{l=1}^{L} \theta_{t-l,u} + \alpha_{t,l}\right)\\
z_{t,u,b} \mid \theta_{t,u} &\sim \text{Multinomial}(\theta_{t,u})\\
w_i \in b \mid \phi_{t,z_{t,u,b}} &\sim \text{Multinomial}(\phi_{t,z_{t,u,b}})\\
w_j \in b \mid \phi_{t,z_{t,u,b}} &\sim \text{Multinomial}(\phi_{t,z_{t,u,b}}).
\end{aligned}$$

Inference for the long-term dependency UCT is quite similar to that for the short-term dependency UCT. The parameters $\theta_{t,u}$ and $\phi_{t,z}$ in Equations (13) and (15) in the long-term dependency UCT can be integrated in the exact same way as before (since priors are still Dirichlet distributed) and $\theta_{t,u}$ and $\phi_{t,z}$ at time t can be inferred using the proposed Gibbs sampling in Algorithm 2. The only difference lies in the way we sample the latent topic for each word pair (step 19 in Algorithm 2), the update rules for the priors (step 24 in Algorithm 2), and the way of obtaining $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$ (step 25 in Algorithm 2). Similar to Equation (9), we sample a latent topic for a word pair b by:

$$\begin{aligned}
&P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \{\mathbf{\Phi}_{t-l}, \mathbf{\Theta}_{t-l}, \alpha_{t,l}, \beta_{t,l}\}_{l=1}^{L})\\
&\quad\propto \frac{n_{t,z,w_i} + \{\phi_{t-l,z,w_i} + \beta_{t,w_i,l}\}_{l=1}^{L} - 1}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \{\phi_{t-l,z,v} + \beta_{t,v,l})\}_{l=1}^{L} - 1)}\\
&\quad\times \frac{n_{t,z,w_j} + \{\phi_{t-l,z,w_j} + \beta_{t,w_j,l}\}_{l=1}^{L} - 1}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \{\phi_{t-l,z,v} + \beta_{t,v,l}\}_{l=1}^{L}) - 1)}\\
&\quad\times \frac{m_{t,u,z} + \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L} - 1}{\sum_{z=1}^{K}(m_{t,u,z} + \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L}) - 1}.
\end{aligned} \tag{16}$$

The derivation of Equation (16) is similar to that of Equation (9) (see Appendix A). For obtaining optimal priors $\alpha_{t,z,l}$ and $\beta_{t,v,l}$, we again apply the fixed-point iterations using the two bounds in the work of Minka [50], resulting in the update rules for $\alpha_{t,z,l}$ and $\beta_{t,v,l}$ in Equation (16) as:

$$\begin{aligned}
\alpha_{t,z,l} &\leftarrow \frac{\alpha_{t,z,l}\sum_u(\Psi(\Upsilon_3) - \Psi(\Upsilon_4))}{\sum_u \Psi \sum_z(\Upsilon_3) - \Psi(\sum_z \Upsilon_4)},\\
\beta_{t,v,l} &\leftarrow \frac{\beta_{t,v,l}\sum_z(\Psi(\Upsilon_c)) - \Psi(\Upsilon_d))}{\sum_z \Psi \sum_v(\Upsilon_c) - \Psi(\sum_v \Upsilon_d)},
\end{aligned} \tag{17}$$

where Υ_3, Υ_4, Υ_c, and Υ_d are defined as:

$$\begin{aligned}
\Upsilon_3 &= m_{t,u,z} + \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L} - 1, \quad \Upsilon_4 = \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L},\\
\Upsilon_c &= n_{t,z,v} + \{\phi_{t-l,z,v} + \beta_{t,v,l}\}_{l=1}^{L} - 1, \quad \Upsilon_d = \{\phi_{t-l,z,v} + \beta_{t,v,l}\}_{l=1}^{L}.
\end{aligned}$$

The derivations of the update rules for $\alpha_{t,z,l}$ and $\beta_{t,v,l}$ in the long-term dependency UCT model are quite similar to those for $\alpha_{t,z}$ and $\beta_{t,v}$ in the short-term dependency UCT model (see Appendix B).

Again, once the Gibbs sampling has been done, with the fact that a Dirichlet distribution is conjugate to a multinomial distribution, we then conveniently infer the following distributions for $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$ in the long-term dependency UCT model with L-steps (step 25 in Algorithm 2), respectively:

$$\begin{aligned}\theta_{t,u,z} &= \frac{m_{t,u,z} + \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L} - 1}{\sum_{z=1}^{K}(m_{t,u,z} + \{\theta_{t-l,u,z} + \alpha_{t,z,l}\}_{l=1}^{L}) - 1},\\ \phi_{t,z,v} &= \frac{n_{t,z,v} + \{\phi_{t-l,z,v} + \beta_{t,v,l}\}_{l=1}^{L} - 1}{\sum_{v=1}^{V_t}(n_{t,z,v} + \{\phi_{t-l,z,v} + \beta_{t,v,l}\}_{l=1}^{L}) - 1}.\end{aligned} \tag{18}$$

4.4. Clustering Users

Users in document streams can be classed into two categories, i.e., those who have existed before the current time period t—previously seen users—and those who just emerge in the streams at the current time period t—previously unseen users. In the following, we detail the way we cluster these two categories of users.

Clustering Previously Seen Users. After we have inferred and determined each user's $u \in \mathbf{U}_t$ dynamic topic distribution at time t, $\theta_{t,u}$ (obtained by either Equation (11) in short-term dependency UCT or Equation (18) in long-term dependency UCT), we use K-means [32, 46] to compute the clusters of these users based on each user's topic distribution $\theta_{t,u}$. Obviously, as time progresses, the clusters of these users dynamically change.

Clustering Previously Unseen Users. However, in some cases, we do not have users' interests $\theta_{t,u_{\text{new}}}$ for new arriving users $u_{\text{new}} \notin \mathbf{U}_{t-1}$. We then infer each new user's interests from their published tweets at time period t, where tweets are preprocessed into a word pair set $\mathbf{B}_{t,u_{\text{new}}}$ as discussed in Section 4.2. We compute the probability of the user u_{new} being interested in topic z at time t, i.e., $\theta_{t,u_{\text{new}},z}$, as:

$$P(z \mid t, u_{\text{new}}) = \sum_{b \in \mathbf{B}_{t,u_{\text{new}}}} P(z \mid t, b) P(b \mid t, u_{\text{new}}), \tag{19}$$

where $P(z \mid t, b)$ is computed as:

$$\begin{aligned}P(z \mid t, b) &= \frac{P(w_i \mid t, z)P(w_j \mid t, z)P(z \mid t)}{P(b \mid t)}\\ &= \frac{P(w_i \mid t, z)P(w_j \mid t, z)P(z \mid t)}{\sum_{z'} P(z' \mid t)P(w_i \mid t, z')P(w_j \mid t, z')}\\ &= \frac{P(z \mid t)\phi_{t,z,w_i}\phi_{t,z,w_j}}{\sum_{z'} P(z' \mid t)\phi_{t,z',w_i}\phi_{t,z',w_j}},\end{aligned} \tag{20}$$

where $P(w \mid t, z)$ is the probability of word w associated with topic z at t, i.e., $\phi_{t,z,v}$, and $P(z \mid t)$ is the probability of topic z at t. We obtain $P(z \mid t)$ for Equation (20) as:

$$P(z \mid t) = \frac{n_t(z, w)}{n_t(w)}, \tag{21}$$

where we use $n_t(z, w)$ and $n_t(w)$ to denote the total number of words assigned to topic z and the total number of words at time t, respectively.

Then, we estimate $P(b \mid t, u_{\text{new}})$ in Equation (19) as:

$$P(b \mid t, u_{\text{new}}) = \frac{n_{t,u_{\text{new}}}(b)}{\sum_b n_{t,u_{\text{new}}}(b)}, \tag{22}$$

where $n_{t,u_{\text{new}}}(b)$ is the frequency of word pair b in $\mathbf{B}_{t,u_{\text{new}}}$.

Finally, after applying Equations (20), (21), and (22) to Equation (19), we obtain the new user's interests $\theta_{t,u_{\text{new}}}$. We then group this user into a cluster $\boldsymbol{\omega}_{t,u_{\text{new}}}$ where they share most interests with other users in the cluster:

$$\boldsymbol{\omega}_{t,u_{\text{new}}} = \arg\max_{\boldsymbol{\omega}_{t,u}} \sum_{u \in \boldsymbol{\omega}_{t,u}} \frac{\cos(\theta_{t,u}, \theta_{t,u_{\text{new}}})}{|\boldsymbol{\omega}_{t,u}|}, \tag{23}$$

and update the user set $\mathbf{U}_t$ as $\mathbf{U}_t \leftarrow \mathbf{U}_t \cup \{u_{\text{new}}\}$.

5. EXPERIMENTAL SETUP

In this section, we describe our experimental setup. Section 5.1 lists our research questions; Section 5.2 describes our dataset; Section 5.3 and Section 5.4 list the baselines and metrics for evaluation, respectively.

5.1. Research Questions

We seek to answer the following research questions that guide the remainder of the article:

RQ1 Does our dynamic user clustering method UCT outperform state-of-the-art baseline methods? (See Section 6.1)
RQ2 What is the impact of the different time slices in our dynamic user clustering method? (See Section 6.2)
RQ3 What is the user clustering performance of long-term dependency UCT model compared to that of the short-term dependency UCT model? (See Section 6.3)
RQ4 What is the quality of the topical representation inferred by our UCT model? (See Section 6.4)
RQ5 Can we capture the dynamics of users' interests and make the clustering results produced by our proposed dynamic user topic model explainable? (See Section 6.5)
RQ6 What is the generalization performance of the UCT topic model compared to other baseline topic models? (See Section 6.6)
RQ7 What is the contribution of modeling word pairs rather than each individual word in our UCT topic model? (See Section 6.7)

5.2. Dataset

In order to answer our research questions, we work with a dataset collected by UCL's Big Data Institute from Twitter.[3] The data set contains 1,375 active users that were randomly sampled from Twitter and their tweets that were posted from the beginning of their registration up to May 31, 2015. In total, we have 3.78 million tweets with each tweet having its own timestamp. The average length of the tweets is 12 words. Due to the crawling restrictions imposed by Twitter, we cannot obtain the follower-followee relationships for each user. So we ignore the possibility of using users' relationships to improve the performance; we leave this as part of our future work.

We use this dataset as our streams of short texts and manually judge the clusters of the 1,375 users based on the content of their published tweets. We obtain ground truth clusters for five different partitions of time periods, i.e., a week, a month, a quarter, half a year, and a year. In the ground truth clusters for time periods of a week, the users are manually clustered through their published tweets during a week, resulting in 48 to 60 clusters. We also create ground truth for time periods of a month, a quarter, half a year, and a year, with the number of clusters varying from 43 to 52, 40 to 46, 28 to 30, and 28 to 30, respectively. For each partition of time periods, we have 18 human annotators to label categories after examining the content of the tweets. All of them

[3] The dataset is publicly available from https://bitbucket.org/sliang1/uct-dataset/get/UCT-Dataset.zip.

own intermediate or high English level certifications. We apply the Open Directory Project (ODP) category system[4] during our annotation process. Thus, in our dataset each user's interests are represented as ratings on 15 categories from the ODP system. The ratings are integral scores from 0 to 5 according to the relevance to the categories. A rating score 0 on a category indicates that the content of the tweets posted by the user, i.e., the user's interests, is not relevant to that category, whereas a rating score 5 indicates the highest relevance to that category.

Given a specific time period and the 15-dimensional ground truth vectors representing the users' interests, to reduce annotator workload for the cluster labeling task, we first apply a K-means clustering algorithm to get the initial clustering results. After that, within each cluster, we calculate the cosine distance between each user in the cluster and the centroid of the cluster. Our human annotators identify users far from the centroid and re-cluster to other clusters. On average, among the users that need to be re-clustered, inter-annotator agreement on these users' new cluster assignments is 87.3%. For pre-processing, we remove stop words and apply Porter stemming using the Lemur toolkit.[5]

5.3. Baselines

We compare our proposed method UCT[6] with the following baselines and state-of-the-art clustering strategies in our experiments:

Trivial-All This is a trivial baseline that assigns all users to the same cluster.

Trivial-Each This is a trivial baseline that assigns each user to its own cluster, i.e., one cluster per user.

Trivial-Random This is a trivial baseline that randomly assigns users to K clusters.

K-means This is a traditional clustering algorithm [32, 46]. It represents users by TF-IDF vectors and categorizes them into different clusters based on their TF-IDF vector similarities.

GSDMM This is a collapsed Gibbs Sampling algorithm for Dirichlet multinomial mixture model (GSDMM) to short text clustering [79]. It represents each short document through a single topic and groups each user into a cluster that contains most of her tweets.

LDA This model infers topic distributions specific to each document via the LDA model.

Author Topic Model (AuthorT) This model [63] infers topic distributions specific to each user in a static dataset, and then clusters the users into different clusters based on the similarities of their topic distributions.

DTM This model [7] utilizes a Gaussian distribution for inferring topic distribution of long text documents in streams.

Topic over time model (ToT) This model [74] normalizes timestamps of long documents in a collection and then infers topics distribution for each document.

TTM This model [31] captures the dynamic topic distribution of long documents arriving during time period t in streams based on the content of the documents and the previous estimated distributions.

For the LDA, DTM, ToT, and TTM baselines, we use the averaged topic distribution of all the documents a user posted before generated by LDA, DTM, ToT, and TTM, respectively, to represent this user, and cluster users based on their topic distribution similarities. For static baseline topic models, i.e., GSDMM, LDA, and AuthorT, we set $\alpha = 0.1$ and $\beta = 0.01$. We set the number of topics $K = 50$ and the number of clusters

[4] http://www.dmoz.org.

[5] http://www.lemurproject.org.

[6] The code is publicly available from https://github.com/yukunZhao/UCT.

equal to the number of topics. For the baseline dynamic topic models, i.e., DTM, ToT, and TTM, we set $\alpha = 0.1$ and $\beta = 0.01$ at time period $t = 0$. Again, we set the number of topics $K = 50$ and the number of clusters equal to the number of topics. We found that when the number of topics in all the baselines is large enough ($K \geq 20$), the experimental outcomes are qualitatively the same. Wallach et al. [73] and Asuncion et al. [4] demonstrated that the tuning of hyperparameters is crucial to the performance of topic models. Other ways of tuning the hyperparameters in our baseline models may help to improve the performance. We leave this as future work.

5.4. Evaluation Metrics

Given the number of clusters P in the ground truth and the number of output clusters Q, we set $\mathcal{C} = \{c_1, \ldots, c_j, \ldots, c_P\}$ as a set of ground-truth clusters and $\Omega = \{\omega_1, \ldots, \omega_i, \ldots, \omega_Q\}$ as the set of output clusters at time slice t, respectively. We use the following metrics that capture different evaluation criteria, orienting clustering quality and topic representation quality, to evaluate our experimental results, all of which are widely used in the literature [3, 47, 77, 79].

Precision. At time slice t, each output cluster ω_i is assigned to a ground-truth cluster c_j *iff* the intersection of the two clusters $\omega_i \cap c_j$ owns the largest number of users. In case of a draw, we randomly assign the output cluster ω_i to one of the ground-truth clusters that call the draw, as the random assignment does not result in different evaluation performance. Then, the *precision* of this assignment is measured by counting the number of user-pairs in the intersection correctly assigned and divided by the total number of user-pairs in the output cluster ω_i:

$$\text{Precision}(\mathcal{C}, \Omega) = \frac{1}{Q} \sum_{i=1}^{Q} \frac{\binom{\max_j |\omega_i \cap c_j|}{2}}{\binom{|\omega_i|}{2}},$$

where $\binom{|\omega_i \cap c_j|}{2}$ and $\binom{|\omega_i|}{2}$ are the number of two-combinations from a given set $\omega_i \cap c_j$ and ω_i, respectively. Obviously, a higher precision indicates better user clustering performance.

Purity. To compute purity, each output cluster ω is assigned to the ground-truth cluster that is most frequent in the cluster, and the the accuracy of this assignment is measured by counting the number of correctly assigned users and dividing by N. Here, N is the total number of users in $\mathcal{C}$. Formally, it is defined as:

$$\text{Purity}(\mathcal{C}, \Omega) = \frac{1}{N} \sum_{i=1}^{Q} \max |\omega_i \cap c_j|,$$

where $|\omega_i \cap c_j|$ is the number of users in the intersection $\omega_i \cap c_j$. Larger purity value indicates better clustering performance.

Normalized Mutual Information (NMI). NMI is a measure that allows us to make the tradeoff between the quality of the clustering and the number of clusters. It is an entropy-based metric that explicitly measures the amount of statistical information shared by the variables representing the output clusters and the ground truth clusters of users. Let $I(\Omega; \mathcal{C})$ denote the mutual information of the output cluster set Ω and the ground-truth cluster set $\mathcal{C}$. NMI avoids the value biasing to large number of clusters by using entropy of Ω and $\mathcal{C}$, i.e., $E(\Omega)$ and $E(\mathcal{C})$:

$$\begin{aligned}\mathrm{NMI}(\mathcal{C},\Omega) &= \frac{I(\Omega;\mathcal{C})}{[E(\Omega)+E(\mathcal{C})]/2}\\ &= \frac{\sum_{i,j}\frac{|\omega_i\cap c_j|}{N}\log\frac{N|\omega_i\cap c_j|}{|\omega_i||c_j|}}{\left(-\sum_i\frac{|\omega_i|}{N}\log\frac{|\omega_i|}{N}-\sum_j\frac{|c_j|}{N}\log\frac{|c_j|}{N}\right)/2}.\end{aligned}$$

Note that when $\mathcal{C}$ is equal to Ω, NMI reaches 1, i.e., its maximum value. Larger NMI value indicates better clustering performance.

Adjusted Rand Index (ARI). Consider clustering users based on a series of pair-wise decisions. If two users both in the same cluster are aggregated into the same cluster and two users in different classes are aggregated into different clusters, the decision is considered to be correct. The Rand index shows the percentage of decisions that are correct while the adjusted Rand index is the corrected-for-chance version of the Rand index [30]. The maximum value is one for exact match; larger values mean better performance for clustering. $\mathrm{ARI}(\mathcal{C},\Omega)$ is computed as follows, where N is the total number of users:

$$\mathrm{ARI}(\mathcal{C},\Omega)=\frac{\sum_{i,j}\binom{|\omega_i\cap c_j|}{2}-\left[\sum_i\binom{|\omega_i|}{2}\sum_j\binom{|c_j|}{2}\right]/\binom{N}{2}}{\frac{1}{2}\left[\sum_i\binom{|\omega_i|}{2}+\sum_j\binom{|c_j|}{2}\right]-\left[\sum_i\binom{|\omega_i|}{2}\sum_j\binom{|c_j|}{2}\right]\Big/\binom{N}{2}}.$$

A large ARI value indicates better clustering performance.

The four metrics introduced so far, Precision, Purity, NMI, and ARI, are for evaluating the performance of user clustering, whereas the following two metrics are for evaluating the quality of topic representations of users in clusters.

H-score. As our UCT model builds on topic modeling, we consider to evaluate the quality of the topic representation of each user using the H-score [9, 77] metric, which is computed as:

$$\text{H-score}(\mathcal{C})=\frac{\mathrm{IntraDis}(\mathcal{C})}{\mathrm{InterDis}(\mathcal{C})},$$

where the average intra-cluster distance $\mathrm{IntraDis}(\mathcal{C})$ and average inter-cluster distance $\mathrm{InterDis}(\mathcal{C})$ are computed as:

$$\mathrm{IntraDis}(\mathcal{C})=\frac{1}{P}\sum_{p}\sum_{\substack{u_i,u_j\in\mathcal{C}_p\\ i\neq j}}\frac{\mathrm{dis}(u_i,u_j)}{\binom{|\mathcal{C}_k|}{2}},$$

$$\mathrm{InterDis}(\mathcal{C})=\frac{1}{P(P-1)}\sum_{\substack{\mathcal{C}_k,\mathcal{C}_{k'}\in\mathcal{C}\\ k\neq k'}}\left[\sum_{\substack{u_i\in\mathcal{C}_k\\ u_j\in\mathcal{C}_{k'}}}\frac{\mathrm{dis}(u_i,u_j)}{|\mathcal{C}_k||\mathcal{C}_{k'}|}\right],$$

where $\mathrm{dis}(u_i,u_j)$ is the symmetric Kullback-Leibler divergence [47] of topic distributions of user u_i and user u_j, and is computed as:

$$\mathrm{dis}(u_i,u_j)=\frac{1}{2}(D_{KL}(u_i||u_j)+D_{KL}(u_j||u_i)).$$

Here, $D_{KL}(u_i||u_j)$ is the Kullback-Leibler divergence between u_i's topic distribution $\{u_{i,z}\}_{z=1}^{K}$ and u_j's topic distribution $\{u_{j,z}\}_{z=1}^{K}$, and is computed as:

$$D_{KL}(u_i||u_j) = \sum_{z=1}^{K} u_{i,z} \log \frac{u_{i,z}}{u_{j,z}}.$$

In a very small number of cases, we have $u_{j,z} = 0$ in the above equation, which results in $u_{i,z} \log \frac{u_{i,z}}{0}$ being undefined, and thus, the Kullback-Leibler divergence is undefined. One way to solve this problem is to apply the smoothed probabilities of $u_{j,z}$, denoted as $\hat{u}_{j,z}$, which is defined as [20]:

$$\hat{u}_{j,z} = \frac{\epsilon + u_{j,z}}{\epsilon * K + \sum_{z=1}^{K} u_{j,z}},$$

where ϵ can be set to be $\epsilon = \frac{1}{2*P}$ [20]. The intuition behind the H-score is that if the average inter-cluster distance is small compared to the average intra-cluster distance, the topical representation of users reaches good performance. Obviously, a lower H-score indicates better topic representation of users.

Perplexity. The perplexity, used by convention in language modeling and the evaluation of many topic models, is monotonically decreasing in the likelihood of the test data, and is algebraically equivalent to the inverse of the geometric mean per-word likelihood. A lower perplexity score indicates better generalization performance in the topic models. Formally, in our setting of streams of short texts, given a sequentially arriving corpus of documents $\{\ldots, \mathbf{D}_{T-1}, \mathbf{D}_T\}$ in time slices $\{\ldots, (T-1), T\}$ that are generated by users in $\{\ldots, \mathbf{U}_{T-1}, \mathbf{U}_T\}$, respectively, the perplexity of our UCT model can be obtained as:

$$\text{Perplexity}(\{\ldots, \mathbf{D}_{T-1}, \mathbf{D}_T\}) = \exp\left\{-\frac{\sum_{t=1}^{T}\sum_{d\in\mathbf{D}_t} \log P(\mathbf{v}_d \mid t)}{\sum_{t=1}^{T} |\mathbf{D}_t|}\right\},$$

where $P(\mathbf{v}_d|t)$ is the probability of generating the content of document d, $\mathbf{v}_d$, at time t by a topic model, which can be computed as:

$$\begin{aligned} P(\mathbf{v}_d \mid t) &= \prod_{v\in d} P(v \mid t) = \prod_{v\in d}\left(\sum_{z} P(v \mid t, z) P(z \mid t)\right) \\ &= \prod_{v\in d}\left(\sum_{z} \phi_{t,z,v} P(z \mid t)\right) \\ &= \prod_{v\in d}\left(\sum_{z} \phi_{t,z,v} \sum_{u\in\mathbf{U}_t} P(z \mid t, u) P(u \mid t)\right) \\ &= \prod_{v\in d}\left(\sum_{z} \phi_{t,z,v} \sum_{u\in\mathbf{U}_t} \theta_{t,u,z} \frac{1}{|\mathbf{U}_t|}\right). \end{aligned}$$

We report the Precision, Purity, NMI, and ARI scores of all seven baselines listed above and our UCT model to evaluate the clustering performance. Importantly, to evaluate the quality of topical representations, we report H-scores of our UCT model and all baseline methods except GSDMM, and report the perplexity of our UCT model and all the baseline topic models. We cannot compute H-scores for GSDMM as it assumes each document to be assigned a single topic; GSDMM clusters users based on topic assignments, not topic distributions. We evaluate the performance with the above metrics at each time period, and report the mean of the evaluation results. Statistical significance of observed differences between the performance of two user clustering models is tested using a two-tailed paired t-test and is denoted using ▲ (and ▼) for significant differences for $\alpha = .01$, or △ (and ▽) for $\alpha = .05$.

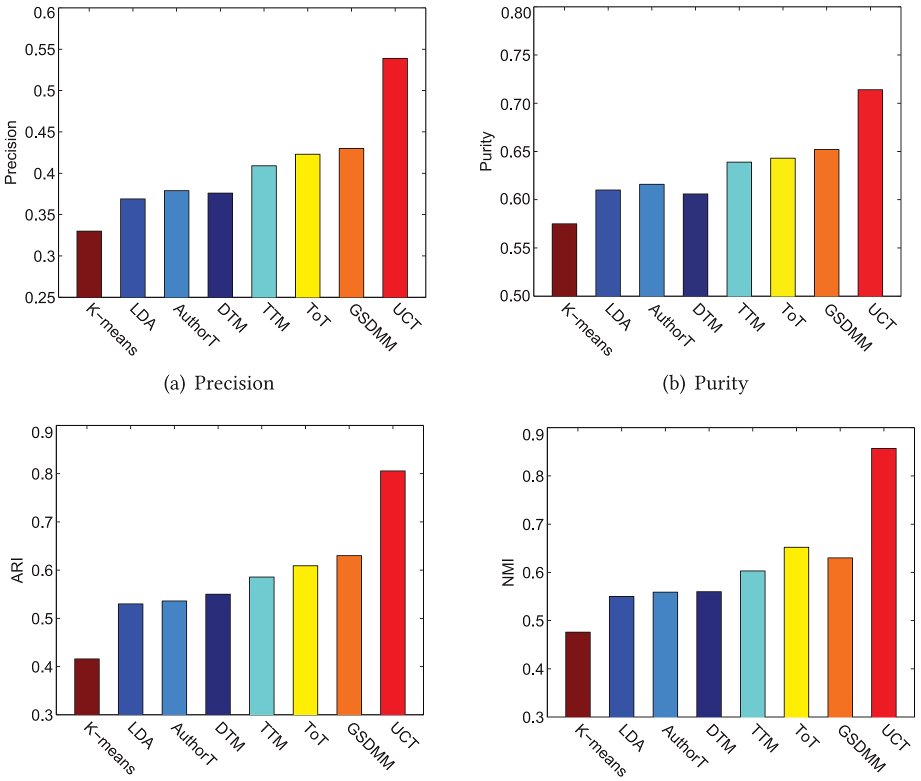
6. RESULTS AND ANALYSIS

In the following sections, we report on our experimental outcomes and formulate answers to our research questions.

6.1. Effectiveness of UCT

To begin, we address research question **RQ1**. We evaluate the performance of our UCT model in the context of streams of short texts, and compare UCT with LDA, an author topic model, AuthorT, a traditional clustering method, K-means, GSDMM, which is a state-of-the-art clustering topic model for short documents in a static set; three dynamic user clustering models, DTM, ToT, and TTM; and three trivial baselines, Trivial-All, Trivial-Each, and Trivial-Random (see Section 5.3). We use short-term dependency UCT as a representative for comparisons with the baselines, as the performance of long-term dependency UCT is better than or at least the same as that of the short-term dependency UCT. Following the training strategy in the work of Shokouhi [65], the training data we use for these eight models are all tweets published from the year 2013 to 2014, which we divide into two parts, each part containing tweets published during a year. We report the precision, purity, NMI, and ARI values of the eight methods by averaging the performance across the two parts.

Figure 3 shows the performance of UCT and the baselines, K-means, GSDMM, LDA, AuthorT, DTM, ToT, and TTM, in terms of cluster-oriented evaluation metrics—Precision, Purity, ARI, and NMI using a time period of a quarter. First, we see that UCT performs significantly better than K-means, GSDMM, and the five topic models, i.e., LDA, AuthorT, DTM, ToT, and TTM, on all the metrics, which demonstrates the effectiveness of our model for user clustering. UCT and the other five topic models outperform K-means, which attests to the merit of utilizing topic models for user clustering. UCT and GSDMM, which infer topic distributions and infer single topic assignments for short documents, respectively, outperform all other baselines in most cases and on all three evaluation metrics. This finding demonstrates that considering documents representing users as short texts rather than as long documents during inference helps to improve the performance on user clustering. UCT, ToT, TTM, and DTM, which infer topic distributions for documents in streams, outperform AuthorT and LDA, which infer topic distributions in static sets of documents. This finding demonstrates that inferring dynamic topic distributions of documents in the context of streams can help to enhance the performance of user clustering over considering documents as a set of static ones for the inference. UCT significantly outperforms all baselines, and this finding confirms that the way that UCT infers dynamic topic distributions of short documents in streams improves the performance of user clustering. Meanwhile, Table II shows the performance of one naive baseline, i.e., K-means, and the three trivial baselines, Trivial-All, Trivial-Each, and Trivial-Random. As can been seen from Table II, none of the trivial baselines can outperform K-means, the worst baseline among K-means, GSDMM, LDA, AuthorT, DTM, ToT, TTM, and our UCT, in terms of all the metrics except the case that Trivial-Each outperforms K-means in terms of purity metric. Because of this, we will not report the performance of these three trivial baselines in the remainder of the article.



(a) Precision (b) Purity

(c) ARI (d) NMI

Fig. 3. Clustering performance of UCT and the baselines using a time period of a quarter. The performance is evaluated using Precision, Purity, ARI, and NMI, respectively.

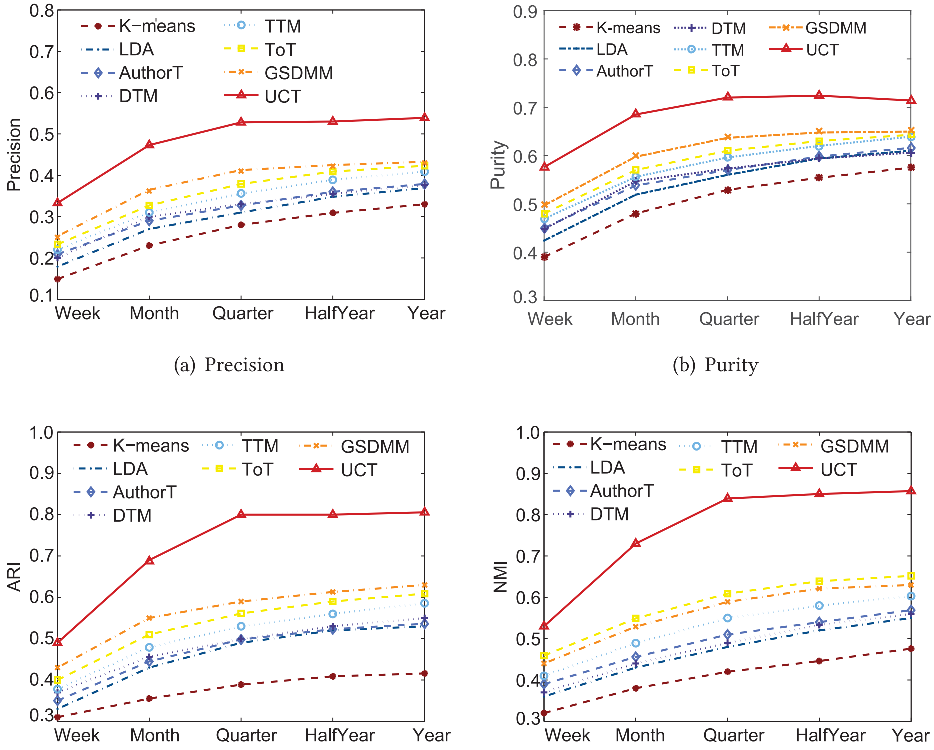
Table II. Performance Comparison Between the Naive Baseline K-means and Three Trivial Clustering Methods, Trivial-All, Trivial-Each, and Trivial-Random

	Precision	Purity	ARI	NMI
Trivial-All	.027	.153	0.000	0.000
Trivial-Each	NA	1.000	NA	.309
Trivial-Random	.145	.273	.235	.391
K-means	.330$^{\blacktriangle}$	.575$^{\blacktriangle}$	.416$^{\blacktriangle}$	.476$^{\blacktriangle}$

Statistically significant differences between K-means and the best trivial baseline per metric are tested using a two-tailed paired T-test and are marked in the upper right hand corner of the K-means scores, respectively.

6.2. Length of Time Periods

Next, we address research question **RQ2**. To understand the influence on UCT of the length of the time period that we use for evaluation, we compare the performance for different time periods: a week, a month, a quarter, half a year, and a year, respectively. Figure 4 shows the evaluation results in terms of Precision, Purity, ARI, and NMI for time periods of different lengths; we average the scores over periods of six weeks, six months, six quarters, four semi-years, and two years, respectively. Again, we use short-term dependency UCT as a representative for comparisons.



(a) Precision

(b) Purity

(c) ARI

(d) NMI

Fig. 4. User clustering performance of UCT and the baselines on time periods of a week, a month, a quarter, half a year, and a year. The performance is evaluated using Precision, Purity, ARI, and NMI, respectively.

As can be seen in Figure 4, UCT always outperforms LDA, AuthorT, DTM, TTM, ToT, and GSDMM for time periods of all lengths. This finding, again, confirms the fact that UCT, which infers topic distributions of short documents based on the previous distributions and arriving documents, works better than the state-of-the-art algorithms for user clustering in streams. When the length of the time period increases from a week to a month, both UCT and the baseline methods all obtain a big improvement, but UCT continues to outperform the other methods. Although the performance of UCT seems to level off on all four metrics when the length of the time period increases from a quarter to a year, it still significantly outperforms the baselines. These findings demonstrate that UCT's performance on the user clustering task is robust in the context of short document streams, and is able to maintain significant improvements over state-of-the-art algorithms.

To further understand why UCT and the baseline methods increase their performance when the length of the time period used for evaluation increases, we provide an analysis of word co-occurrence patterns in different time periods. Descriptive statistics of the tweets users published in different time periods are shown in Table III. On average, a user only publishes 16 tweets per week, which indicates that there are 16×12 word co-occurrence patterns if we assume the average length of each tweet to be 12 words. The number 16×12 is not comparable with the number 1,012, which is the number of word pairs. A larger number of word pairs helps to better infer topic

Table III. Descriptive Statistics, i.e., Average (Ave.), Median (Med.), and Standard Deviation (Std.), of Tweets per User Published Per Week, Month, Quarter, Half a Year, and Year, Respectively

		Med.			Med.			Med.	
	Ave.	week	Std.	Ave.	month	Std.	Ave.	quarter	Std.
#tweets	16	16	5.12	111	108.5	24.51	220	227	40.13
#word pairs	1,012	1,010	323.84	3,586	3,613	791.79	9,199	9,008	1,677.90
		half a year			year				
#tweets	418	413	48.15	744	751	64.28			
#word pairs	14,348	14,456.5	2,066.11	29,810	29,574	2,575.58			

Table IV. The Impact of Dependency Length L on User Clustering

	a week				a month			
	Precision	Purity	ARI	NMI	Precision	Purity	ARI	NMI
UCT-1	.333	.576	.490	.530	.473	.685	.690	.800
UCT-2	.352$^{\triangle}$	.584$^{\triangle}$	.523$^{\blacktriangle}$	.529	.487$^{\blacktriangle}$	.696$^{\triangle}$	.702	.809
UCT-3	.365$^{\blacktriangle}$	.604$^{\blacktriangle}$	.551$^{\blacktriangle}$	.533	.493$^{\blacktriangle}$	.703$^{\triangle}$	.705	.818$^{\triangle}$
UCT-4	.370$^{\blacktriangle}$	.608$^{\blacktriangle}$	.557$^{\blacktriangle}$	.552	.495$^{\blacktriangle}$	.705$^{\triangle}$	.706	.823$^{\triangle}$
UCT-5	.372$^{\blacktriangle}$	.610$^{\blacktriangle}$	.565$^{\blacktriangle}$	.568	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-6	.378$^{\blacktriangle}$	.610$^{\blacktriangle}$	.571$^{\blacktriangle}$	.582	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-7	.382$^{\blacktriangle}$	.613$^{\blacktriangle}$	.577$^{\blacktriangle}$	.602	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-8	.387$^{\blacktriangle}$	.615$^{\blacktriangle}$	.580$^{\blacktriangle}$	.618	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-9	.392$^{\blacktriangle}$	.618$^{\blacktriangle}$	.583$^{\blacktriangle}$	.637	.494$^{\blacktriangle}$	.703$^{\triangle}$	.705	.822$^{\triangle}$
UCT-10	.402$^{\blacktriangle}$	.620$^{\blacktriangle}$	.585$^{\blacktriangle}$	.645	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-11	.404$^{\blacktriangle}$	.621$^{\blacktriangle}$	.590$^{\blacktriangle}$	.662	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
UCT-12	.404$^{\blacktriangle}$	.622$^{\blacktriangle}$	.592$^{\blacktriangle}$	.671	.493$^{\blacktriangle}$	.703$^{\triangle}$	.704	.823$^{\triangle}$
	a quarter				half a year			
UCT-1	.528	.720	.800	.839	.530	.724	.800	.850
UCT-2	.530	.723	.803	.840	.531	.722	.801	.846
UCT-3	.533	.719	.793	.850$^{\triangle}$	.530	.726	.800	.851
UCT-4	.535	.721	.802	.853$^{\triangle}$	.532	.725	.801	.850

UCT-L is the long-term dependency UCT model with L being the length of the dependency under consideration. UCT-1 is the short-term dependency UCT model. Statistically significant differences between UCT-L when $L \geq 2$ and UCT-1 per metric are tested using a two-tailed paired T-test and are marked in the upper right-hand corner of the UCT-L scores, respectively.

distributions in Gibbs sampling. The longer the time period is, the more word pairs can be utilized in our Gibbs sampling for the topic inference.

6.3. Impact of Dependency Length

Next, we address research question **RQ3**. For comparison, we write UCT-L for the long-term dependency UCT model with L being the dependency length, i.e., the number of previous time periods under consideration for inferring the current topic distributions. To make things clear, in Table IV, we write UCT-1 for our short-term dependency UCT model. We examine the user clustering performance by varying the length L from 1 to 10 timesteps when the time slice is set to a week or a month, and from 1 to 5 timesteps when the time slice is set to be a quarter or half a year, respectively.

Table IV compares the performance of the short-term dependency UCT model, UCT-1, and the long-term dependency UCT models, UCT-L, when $L \geq 2$, using the Precision, Purity, NMI, and ARI evaluation metrics and using time periods of a week, a month, a quarter, and half a year, respectively. As can be seen in the table, when $L \geq 2$, UCT-L can statistically significantly outperform UCT-1 on all metrics when using a week as the time slice and $L \geq 2$. We do not show the performance of UCT-L using a week as

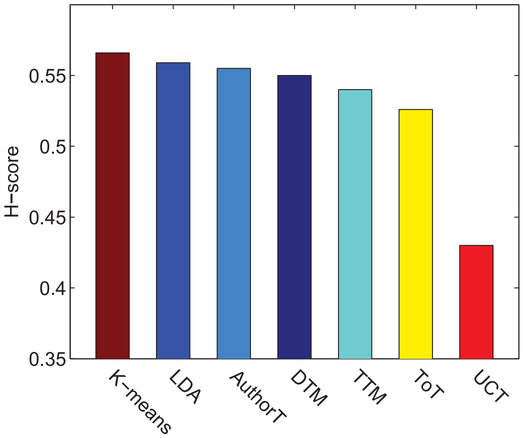


Fig. 5. Evaluation results for the quality of topic representations for UCT and the baselines, using the H-score metric and time periods of a quarter.

the time slice when $L \geq 12$ in the table as it is almost the same as that of the UCT-12. When $L \geq 2$, UCT-L can also statistically significantly outperform UCT-1 on almost all the metrics when using a month as the time slice. The performance of UCT-L using a month as the time slice when $L \geq 3$ is stable. These findings illustrate that the UCT model can enhance the performance of user clustering when more past information of users' distributions is integrated in the model, especially when the time slice is a week. In other words, the long-term dependency UCT model works better than the short-term dependency UCT, especially in terms of using a week and a month as time slices. When the time slices become a quarter and half a year, the performance of the long-term and short-term dependency UCT models is almost the same, which is mainly because the interests of the users inferred by both the long-term and short-term dependency UCT models seem to be the same when the time slices are sufficiently long. It is possible to propose a method to automatically obtain the optimal dependency length L for our UCT-L model such that its performance will be stable with a longer dependency length. However, we leave this as future work.

In the remainder of the experimental analysis, we will focus on the short-term dependency UCT model such that we can further study the performance of our dynamic user clustering model independently of the length of the dependency. The performance of long-term dependency UCT-L with $L \geq 2$ is at least as good as the performance of the short-term dependency UCT model.

6.4. Quality of Topical Representations

We now address research question **RQ4**. To assess the quality of topics extracted by UCT, we compare UCT and the baseline methods. Figure 5 shows the comparison of the performance of UCT and the baselines in terms of H-score. When computing the H-scores for evaluating the quality of topical representation in UCT and the baselines, we use the quarterly ground-truth user clustering results.

It is clear from Figure 5 that UCT obtains a significantly smaller H-score compared to the other six models,[7] which indicates that the average inter-cluster distance is small compared to the average intra-cluster distance. A smaller H-score means that the topical representation of users is more similar to that labeled manually (each

[7] Recall that we cannot calculate the H-score for GSDMM as it assumes that each document is assigned to only a single topic.

Table V. Top 25 Words Representing a Cluster and Two Users Extracted by UCT and ToT, Respectively

UCT	ToT
kid color community robbery spiritual immigration alert kicker education child violence star chocolate girl gay control privilege govt sentiment game jail ticket LGBT service story	media kid toast immigration dog campaign alert labour fan advertisement bet slide John image people school ticket law politics safe chief Mexico officer pride development
kid unit robbery ride arrest education star police civilian flat immigration internal rule city community govt game potty ticket LGBT guilty service law school control	kid education internal photo fan media ad community game score rio toast process http attitude victim politics Mexico law safe school ticket cost service football
immigration kid child kicker process community control color police officer rule city game violence gay LGBT cost sentiment school law nobody govt football safe chief	kid process control child media victim dog Mexico reason score song education robbery trial institution price development politics officer stake game cost school rugby story

Words in the first row represent a cluster, while words in the second and third rows represent two users in the cluster, respectively. Words marked blue represent the most coherent words for topics; those marked orange represent less coherent words and others represent irrelevant words.

cluster in the ground-truth clusters of users has lower average intra-cluster distance and higher inter-cluster distance), which demonstrates a better quality of the topics represented by UCT in contrast to state-of-the-art clustering models.

To further illustrate the quality of topic representations in UCT, we display the top-N words for an output cluster and two users from this cluster. The top-N words of a user are generated as follows. First, we rank the words in decreasing order of the probability $P(w \mid t, u)$, associated with the user, which is computed as $P(w \mid t, u) = \sum_z P(w \mid t, z) \cdot P(z \mid t, u)$; the words ranked within the top-N are then selected to represent the user. For generating the top-N words for a cluster, the words are ranked by the probabilities $P(w \mid t, c)$, associated with the cluster, which is computed by

$$P(w \mid t, c) = \frac{1}{|c|} \sum_{u \in c} \sum_{z} P(w \mid t, z) P(z \mid t, u),$$

i.e.,

$$P(w \mid t, c) = \frac{1}{|c|} \sum_{u \in c} \sum_{z} \phi_{t,z,w} \cdot \theta_{t,u,z}.$$

Then, the top-N words that obtain the highest probabilities $P(w \mid t, c)$ are selected to represent the cluster. Table V shows the top 25 words extracted from a cluster and two users in this cluster, where words in the first row represent a cluster, while words in the second and third rows represent two users in the cluster, respectively. We use ToT as a representative topic model for comparison as it is the best baseline (GSDMM cannot obtain representative words for users' interests and clustering results). We can see from the table that the two users in the same cluster generated by the UCT model share more similar interests represented by the words "kids," "immigration," "community," "education," and so on, from the topic "Kids" and the words "girl," "gay," "privilege," "LGBT," "law," and the like, from the topic "Society," compared to that generated by the ToT model. UCT is able to obtain representative words for a cluster more accurately than ToT. This again, the explainable and human-understandable clustering results further illustrates that the quality of UCT's topic representation is better than that of the baseline methods.

6.5. Dynamic Topic Representation of Users

We address research question **RQ5** in this section. As UCT captures each user's dynamic topic distribution, we investigate the content of the users' interests. We conduct

Table VI. Top 25 Words Representing Two Users' Interests Over Time, Covering Five Quarters from April 2014 to May 2015

Apr. 2014 to Jun. 2014	Jul. 2014 to Sep. 2014	Oct. 2014 to Dec. 2014	Jan. 2015 to Mar. 2015	Apr. 2015 to May 2015
promotion book email battle html prototype tweet feature iOS team coding perspective lane platform car course sdk mobile image offline code beginner app developer level	app kid store dog dialog browser design scenario book mobile element programming night inspirationgame tester creativity target awesome robot perl file system sdk	prototype android web internet design house breakfast iOS film tweet media social people Russia license html5 practice language mobile show game learner app execution workshop	Russia government email designer app photo problem engineer mobile strategy smartphone product team hardware iOS sdk html dress issue inspiration master hour idea ill de	prototype apple music iOS Mac video entrepreneur correlation point interaction task screen years amp reason slide mobile iPad interaction product course problem offline game test
center partner WalMart TXST mall David belt offer game player improvement blue enforcement county Oklahoma campus student shot person shirt football bowl utsa target home	TXST star guy fan game night campus sports basketball member grace feminism equality score post football record time note ticket appearance account damn player efforts	ESPN TXST sports dt StarNews student semester bowl game university star tonight traffic conference battle ballgame Alabama state campus county image review entertainment glimpse	TXST state respect feminism community Texas sexism opinion game campus podcast America nation government student tax nation role violence women body education basketball branch house	violence TXST victim responsibility police official opinion state campus respond columnist follow season lecture Texas college video women assault democrats safety level gender student staff

The first row shows the top 25 words per quarter to represent a user whose interests center on the design of apps. The second row shows the top 25 words per quarter to represent another users whose interests dramatically vary as time progresses. Words marked blue represent the most coherent words for topics; those marked orange represent less coherent words and others represent irrelevant words.

a qualitative analysis and see if the clustering result is explainable. As an example, we randomly choose two users and show their interests over five quarters. Specifically, we show each user's interests at each time period by using the top 25 words in Table VI, where the words are selected from the 10 most probable topics of the user and then the 20 most probable words for each topic. In Table VI, the first row shows the top 25 words per quarter to represent a user whose interests center on the design of apps, whereas the second row shows the top 25 words per quarter to represent another user's whose interests dramatically vary as time progresses.

As seen in Table VI, the first user is concerned with "book, promotion, prototyping, iOS, etc." in the second quarter of 2014 and this is slightly changed to "app, store, browser, dialog, design, etc." and "prototype, android, web, internet, etc." in the following two quarters, respectively. As time moves on in 2015, her interests change to "Russia, government, designer, app, problem, etc." and "prototype, Apple, iOS, music, Mac, etc." The user's interests are almost stable and mainly focus on the design of apps. In contrast, during the second quarter in 2014, the second user is interested in "center, partner, WalMart, game, player, Oklahoma," which are about business, politics, and some sports. Then, she talks more about college football and feminism and equality with words like "TXST, star, game, campus, feminism, equality, etc." in the third quarter of 2014. In the next quarter, this user mostly enjoys college football as represented by words "ESPN, TXST, star, bowl, game, etc." Then, this user is concerned with politics and society with "TXST, state, feminism, government, university" and "violence, victim, responsibility" in 2015. This example illustrates how UCT captures dynamic topic distributions to represent the interests of each user and that the result

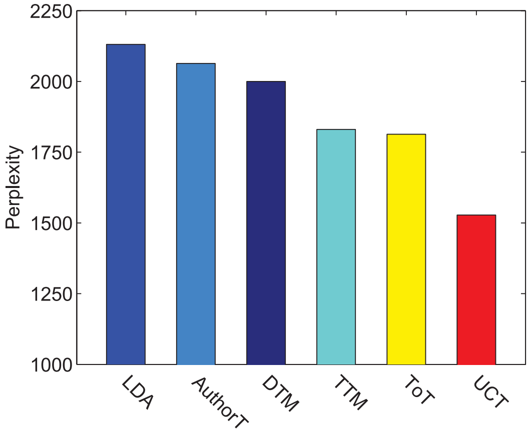


Fig. 6. Generalization performance of the compared six topic models on the perplexity metric using a quarter time slice.

Table VII. Performance Comparison Between UCT$_{LDA}$, Which Assigns One Topic Per Word in Each Document, and UCT, Which Assigns One Topic to Each Word Pair in Each Document

	Precision	Purity	ARI	NMI
UCT$_{LDA}$	.409	.639	.586	.603
UCT	.539$^{\blacktriangle}$	.714$^{\blacktriangle}$	.806$^{\blacktriangle}$	.857$^{\blacktriangle}$

Statistically significant differences between UCT$_{LDA}$ and UCT per metric are tested using a two-tailed paired T-test and are marked in the upper right-hand corner of the UCT scores, respectively.

of dynamic clustering is explainable and understandable in the context of streams of short texts.

6.6. Generalization Comparison

Here, we address our research question **RQ6**. We answer the research question by evaluating the generalization performance of UCT and the baseline topic models in terms of perplexity that is widely used as an effective evaluation metric in many topic models [7, 8, 61]. Note, again, that a lower perplexity score indicates better generalization performance in the topic models. Figure 6 shows the experimental results. As shown in the figure, UCT outperforms all the baseline methods in terms of perplexity metric, which illustrates that the generalization ability of our UCT model is better than that of the baseline topic models.

6.7. Contribution of Modeling Word Pairs

Finally, we address our final research question **RQ7**. We have demonstrated in Section 6.3 that long-term dependency UCT works better than short-term dependency UCT. To better understand the contribution of modeling word pairs rather than individual words to the improvement in user clustering performance, we compare the performance of short-term dependency UCT and that of UCT$_{LDA}$. Here, UCT$_{LDA}$ is a modified version of the short-term dependency UCT, and the only difference lies in the way UCT$_{LDA}$ models words—unlike UCT, which assigns one topic to each word pair, UCT$_{LDA}$ assigns one topic to each word. Table VII shows the performance of UCT$_{LDA}$ and UCT. As can be seen, UCT outperforms UCT$_{LDA}$ on all evaluation metrics, which demonstrates that the strategy of assigning one topic per word pair in topic modeling works better than that of assigning one topic per individual word in the context of streams of short texts for user clustering.

7. CONCLUSION

We have proposed a content-based method for user clustering. Previous work on content-based user clustering has mostly focused on long documents. In contrast, we have studied the problem of dynamically clustering users in the context of streams of short documents. We have proposed a dynamic Dirichlet multinomial mixture UCT model to dynamically cluster both previously seen and previously unseen users based on their interests. To better infer the dynamic topic distribution specific to each user, we have proposed to extract word pairs from each user and apply a Gibbs sampling algorithm for the inference. Besides the proposed short-term dependency UCT that models users' topic distributions at the current time to be dependent on the previous short-term distributions only, to enhance the clustering performance, we also have proposed a long-term dependency UCT that models users' current topic distributions to be dependent on the previous L-steps topic distributions.

For evaluation purposes, we have compared the performance of short-term and long-term dependency UCT to that of a traditional clustering algorithm, K-means, non-dynamic topic models, LDA, the author topic model, and GSDMM that works with short documents, and state-of-the-art dynamic topic models, viz., DTM, ToT, and TTM. Our experimental results demonstrated the clustering effectiveness of our short-term and long-term UCT for user clustering in the context of short document streams, and showed that long-term dependency UCT does work better than short-term dependency UCT. We have also found that UCT produces higher quality topic representations than competing methods, and it comes with the benefit of offering explanations of the clustering.

As to future work, we aim to incorporate other information such as users' social relations to collaboratively group users into clusters. Further research that we are keen to do concerns an evaluation of the similarity of topics, which can be used for automatic selection of K. Another line of work is to develop a more efficient user clustering model to utilize click information and query logs of users for inferring a user's current interests, and to improve efficiency of the Gibbs sampling algorithm. Like most previous work, it is difficult to obtain the ground-truth number of user clusters in our model. Thus, we leave this as future work as well. We plan to apply the proposed user clustering algorithm for user recommendation and for time-aware personalized tweet summarization that considers each user's social circles' interests. We also intend to investigate the effectiveness of our user clustering models at different levels of user activity, e.g., some users very actively post tweets while others do not.

APPENDIXES

A. GIBBS SAMPLING DERIVATION FOR UCT

We calculate the conditional distribution $P(z_{t,u,b} \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$ with the joint distribution $P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$. We can take advantage of conjugate priors to simplify the integrals as follows. All the symbols are defined in Sections 3 and 4. Here, we only provide the derivation for the short-term dependency UCT model, and the derivation for the long-term dependency UCT model is actually similar. Using the chain rule, we can obtain the conditional probability conveniently as:

$$\begin{aligned} &P(z_{t,u,b} \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t) \\ &\quad \propto \frac{P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)}{P(\mathbf{B}_{t,-(u,b)}, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)}. \end{aligned}$$

Thus, to obtain the conditional probability, we need to obtain the join distribution $P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t)$. The join distribution, then, can be calculated as:

$$
\begin{aligned}
&P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t) \\
&\quad = \int P(\mathbf{B}_t \mid \mathbf{Z}_t, \boldsymbol{\Phi}_t) P(\boldsymbol{\Phi}_t \mid \boldsymbol{\Phi}_{t-1}, \beta_t) d\boldsymbol{\Phi}_t \\
&\qquad \times \int P(\mathbf{Z}_t \mid \mathbf{U}_t, \boldsymbol{\Theta}_t) P(\boldsymbol{\Theta}_t \mid \boldsymbol{\Theta}_{t-1}, \alpha_t) d\boldsymbol{\Theta}_t \\
&\quad = \int \prod_{u=1}^{|\mathbf{U}_t|} \prod_{b=1}^{|B_{t,u}|} P(b \mid \phi_{t,z}) \prod_{z=1}^{K} P(\phi_{t,z} \mid \phi_{t-1,z}, \beta_t) d\phi_{t,z} \\
&\qquad \times \int \prod_{u=1}^{|\mathbf{U}_t|} \prod_{b=1}^{|B_{t,u}|} P(z \mid \theta_{t,a}) \prod_{u=1}^{|\mathbf{U}_t|} P(\theta_{t,a} \mid \theta_{t-1,a}, \alpha_t) d\theta_{t,a} \\
&\quad = \left(\int \prod_{z=1}^{K} \prod_{v=1}^{V_t} \phi_{t,z,v}^{n_{t,z,v}} \prod_{z=1}^{K} P(\phi_{t,z} \mid \phi_{t-1,z}, \beta_t) d\phi_t \right)^2 \\
&\qquad \times \int \prod_{u=1}^{|\mathbf{U}_t|} \prod_{z=1}^{K} \theta_{t,u,z}^{m_{t,u,z}} \prod_{u=1}^{|\mathbf{U}_t|} P(\theta_{t,u} \mid \theta_{t-1,u}, \alpha_t) d\theta_t \\
&\quad = \left(\prod_{z=1}^{K} \left(\frac{\Gamma(\sum_{v=1}^{V_t} (\phi_{t-1,z,v} + \beta_{t,w}))}{\prod_{v=1}^{V_t} \Gamma(\phi_{t-1,z,v} + \beta_{t,w})} \right) \right)^2 \\
&\qquad \times \left(\prod_{z=1}^{K} \frac{\prod_{v=1}^{V_t} \Gamma(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,w} - 1)}{\Gamma\left(\sum_{v=1}^{V_t} n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,w} - 1 \right)} \right)^2 \\
&\qquad \times \prod_{u=1}^{|\mathbf{U}_t|} \frac{\Gamma(\sum_{z=1}^{K} (\theta_{t-1,u,z} + \alpha_{t,z}))}{\prod_{z=1}^{K} \Gamma(\theta_{t-1,u,z} + \alpha_{t,z})} \\
&\qquad \times \prod_{u=1}^{|\mathbf{U}_t|} \frac{\prod_{z=1}^{K} \Gamma(m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1)}{\Gamma\left(\sum_{z=1}^{K} m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1 \right)}.
\end{aligned}
$$

Finally, we get:

$$
\begin{aligned}
&P(z_{t,u,b} \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t) \\
&\quad \propto \frac{n_{t,z,w_i} + \phi_{t-1,z,w_i} + \beta_{t,w_i} - 1}{(\sum_{v=1}^{V_t} (n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,w}) - 1)} \\
&\qquad \times \frac{n_{t,z,w_j} + \phi_{t-1,z,w_j} + \beta_{t,w_j} - 1}{(\sum_{v=1}^{V_t} (n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,w}) - 1)} \\
&\qquad \times \frac{m_{t,u,z} + \theta_{t-1,u,z+}\alpha_{t,z} - 1}{\sum_{z=1}^{K} (m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z}) - 1},
\end{aligned}
$$

where w_i and w_j represent the two words in word pair b, and $n_{t,u,z}$ is the number of word pairs published by user u assigned to topic z.

B. DERIVATION OF UPDATE RULES

We apply a fixed-point iteration for estimating the parameters α_t and β_t by maximizing the joint distribution $P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$. Again, here we only show the derivation of the updating rules for α_t and β_t in the short-term dependency UCT model. The derivation of the updating rules for these two parameters in the long-term dependency UCT model is similar to that in the short-term dependency UCT model. The joint probability is as follows:

$$\begin{aligned}
&P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)\\
&\quad= \int P(\mathbf{B}_t \mid \mathbf{Z}_t, \mathbf{\Phi}_t) P(\mathbf{\Phi}_t \mid \mathbf{\Phi}_{t-1}, \beta_t) d\mathbf{\Phi}_t\\
&\qquad \times \int P(\mathbf{Z}_t \mid \mathbf{U}_t, \mathbf{\Theta}_t) P(\mathbf{\Theta}_t \mid \mathbf{\Theta}_{t-1}, \alpha_t) d\mathbf{\Theta}_t\\
&\quad= \int \prod_{u=1}^{|\mathbf{U}_t|} \prod_{b=1}^{|\mathbf{B}_{t,u}|} P(b \mid \phi_{t,z}) \prod_{z=1}^{K} P(\phi_{t,z} \mid \phi_{t-1,z}, \beta_t) d\phi_{t,z}\\
&\qquad \times \int \prod_{u=1}^{|\mathbf{U}_t|} \prod_{b=1}^{|\mathbf{B}_{t,u}|} P(z \mid \theta_{t,u}) \prod_{u=1}^{|\mathbf{U}_t|} P(\theta_{t,u} \mid \theta_{t-1,u}, \alpha_t) d\theta_{t,u}\\
&\quad= \left(\prod_z \left(\frac{\Gamma(\sum_v (\Upsilon_b))}{\prod_v \Gamma(\Upsilon_b)} \frac{\prod_v \Gamma(\Upsilon_a)}{\Gamma(\sum_v \Upsilon_a)} \right) \right)^2 \times \prod_u \frac{\Gamma(\sum_z (\Upsilon_2))}{\prod_z \Gamma(\Upsilon_2)} \frac{\prod_z \Gamma(\Upsilon_1)}{\Gamma(\sum_z \Upsilon_1)},
\end{aligned}$$

where Υ_1, Υ_2, Υ_a, and Υ_b are shown as follows:

$$\begin{aligned}
&\Upsilon_1 = m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z} - 1,\ \Upsilon_2 = \theta_{t-1,u,z} + \alpha_{t,z}\\
&\Upsilon_a = n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v} - 1,\ \Upsilon_b = \phi_{t-1,z,v} + \beta_{t,v}.
\end{aligned}$$

Instead of maximizing the joint distribution directly, we try to maximize the following log-likelihood:

$$\begin{aligned}
&\log P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)\\
&\quad= 2 \sum_z \left(\log \Gamma \left(\sum_v (\Upsilon_b) \right) - \log \Gamma \left(\sum_v \Upsilon_a \right) \right) + 2 \sum_z \sum_v (\log \Gamma(\Upsilon_a) - \log \Gamma(\Upsilon_b))\\
&\qquad + \sum_u \left(\log \Gamma \left(\sum_z (\Upsilon_2) \right) - \log \Gamma \left(\sum_z \Upsilon_1 \right) \right) + \sum_u \sum_z (\log \Gamma(\Upsilon_1) - \log \Gamma(\Upsilon_2)).
\end{aligned}$$

Using the following two bounds from the work of Minka [50],

$$\begin{aligned}
&\log \Gamma(\hat{x}) - \log \Gamma(\hat{x} + n) \geq \log \Gamma(x) - \log \Gamma(x + n) + (\Psi(x + n) - \Psi(x))(x - \hat{x})\\
&\log \Gamma(\hat{x} + n) - \log \Gamma(\hat{x}) \geq \log \Gamma(x + n) - \log \Gamma(x) + x(\Psi(x + n) - \Psi(x))(\log \hat{x} - \log x),
\end{aligned}$$

and assuming that $\hat{x}$ is the optimal updating parameter in the next fixed-point iteration, we have:

$$\begin{aligned}
&\log P(\mathbf{B}_t, \mathbf{Z}_t, \mathbf{U}_t \mid \Phi_{t-1}, \Theta_{t-1}, \{\alpha_{t,1}, \ldots, \hat{\alpha}_{t,z}, \ldots, \alpha_{t,K}\}, \beta_t) \geq L(\hat{\alpha}_{t,z})\\
&\quad= \sum_u \left(\Psi \left(\sum_z \Upsilon_1 \right) - \Psi \left(\sum_z \Upsilon_2 \right) \right) (-\hat{\alpha}_{t,z}) + \alpha_{t,z} \sum_u (\Psi(\Upsilon_a) - \Psi(\Upsilon_b)) \log \hat{\alpha}_{t,z} + C,
\end{aligned}$$

where C is a function not containing the variable $\hat{\alpha}$. Then we let:

$$\frac{\partial L(\hat{\alpha}_{t,z})}{\partial \hat{\alpha}_{t,z}} = \frac{\alpha_{t,z} \sum_u (\Psi(\Upsilon_1) - \Psi(\Upsilon_2))}{\hat{\alpha}_{t,z}} - \sum_u \left(\Psi\left(\sum_z \Upsilon_1\right) - \Psi\left(\sum_z \Upsilon_2\right)\right) = 0,$$

which results in the following updating rule for the Dirichlet prior α_t:

$$\hat{\alpha}_{t,z} \leftarrow \frac{\alpha_{t,z} \sum_u (\Psi(\Upsilon_1) - \Psi(\Upsilon_2))}{\sum_u (\Psi(\sum_z \Upsilon_1) - \Psi(\sum_z \Upsilon_2))}.$$

Following the same derivation, we have the update rule for $\beta_{t,v}$:

$$\hat{\beta}_{t,v} \leftarrow \frac{\beta_{t,v} \sum_z (\Psi(\Upsilon_a) - \Psi(\Upsilon_b))}{\sum_z (\Psi(\sum_v \Upsilon_a) - \Psi(\sum_v \Upsilon_b))}.$$

ACKNOWLEDGMENTS

We are grateful to our reviewers for providing very detailed and helpful feedback on earlier versions of this article.

REFERENCES

[1] Amr Ahmed and Eric P. Xing. 2012. Timeline: A dynamic hierarchical Dirichlet process model for recovering birth/death and evolution of topics in text stream. In *UAI*. AUAI, 20–29.

[2] Faiyaz Al Zamal, Wendy Liu, and Derek Ruths. 2012. Homophily and latent attribute inference: Inferring latent attributes of Twitter users from neighbors. In *ICWSM*. AAAI, 387–390.

[3] Enrique Amigó, Julio Gonzalo, and Felisa Verdejo. 2013. A general evaluation measure for document organization tasks. In *SIGIR*. ACM, 643–652.

[4] Arthur Asuncion, Max Welling, Padhraic Smyth, and Yee Whye Teh. 2009. On smoothing and inference for topic models. In *UAI*. AUAI, 27–34.

[5] Krisztian Balog and Maarten de Rijke. 2007. Finding similar experts. In *SIGIR*. ACM, 821–822.

[6] David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. 2010. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *J. ACM* 57, 2 (2010), 7:1–7:30.

[7] David M. Blei and John D. Lafferty. 2006. Dynamic topic models. In *ICML*. 113–120.

[8] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3, 4–5 (2003), 993–1022.

[9] Ilaria Bordino, Carlos Castillo, Debora Donato, and Aristides Gionis. 2010. Query similarity by projecting the query-flow graph. In *SIGIR*. ACM, 515–522.

[10] Georg Buscher, Ryen W. White, Susan Dumais, and Jeff Huang. 2012. Large-scale analysis of individual and task differences in search result page examination strategies. In *WSDM*. ACM, 373–382.

[11] Meeyoung Cha, Hamed Haddadi, Fabricio Benevenuto, and P. Krishna Gummadi. 2010. Measuring user influence in Twitter: The million follower fallacy. In *ICWSM*. AAAI, 30.

[12] Charalampos Chelmis and Viktor K. Prasanna. 2013. Social link prediction in online social tagging systems. *ACM Transactions on Information Systems* 31, 4 (2013), Article 20.

[13] Weizheng Chen, Jinpeng Wang, Yan Zhang, Hongfei Yan, and Xiaoming Li. 2015. User based aggregation for biterm topic model. In *ACL*. ACL, 489–494.

[14] Zhiyuan Chen and Bing Liu. 2014. Mining topics in documents: Standing on the shoulders of big data. In *KDD*. ACM, 1116–1125.

[15] Xueqi Cheng, Xiaohui Yan, Yanyan Lan, and Jiafeng Guo. 2014. BTM: Topic modeling over short texts. *IEEE Transactions on Knowledge and Data Engineering* 26 (2014), 2928–2941.

[16] Qiming Diao and Jing Jiang. 2014. Recurrent Chinese restaurant process with a duration-based discount for event identification from Twitter. In *SDM*. SIAM, 388–397.

[17] Qiming Diao, Jing Jiang, Feida Zhu, and Ee-Peng Lim. 2012. Finding bursty topics from microblogs. In *ACL*. ACL, 536–544.

[18] Charles Elkan. 2006. Clustering documents with an exponential-family approximation of the Dirichlet compound multinomial distribution. In *ICML*. ACM, 289–296.

[19] Brendan J. Frey and Delbert Dueck. 2007. Clustering by passing messages between data points. *Science* 315, 5814 (2007), 972–976.

[20] Wei Gao and Fabrizio Sebastiani. 2016. From classification to quantification in tweet sentiment analysis. *Social Network Analysis and Mining* 6, 1 (2016), 1–22.

[21] Sean Gerrish and David M. Blei. 2010. A language-based approach to measuring scholarly impact. In *ICML*. ACM, 375–382.

[22] Jennifer Gillenwater, Alex Kulesza, and Ben Taskar. 2012. Discovering diverse and salient threads in document collections. In *EMNLP-CoNLL*. ACL, 710–720.

[23] Spence Green, Nicholas Andrews, Matthew R. Gormley, Mark Dredze, and Christopher D. Manning. 2012. Entity clustering across languages. In *NAACL-HLT*. ACL, 60–69.

[24] Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. *PNAS* 101, suppl 1 (2004), 5228–5235.

[25] Xianpei Han and Le Sun. 2012. An entity-topic model for entity linking. In *EMNLP*. ACL, 105–115.

[26] Xiangnan He, Min-Yen Kan, Peichu Xie, and Xiao Chen. 2014. Comment-based multi-view clustering of web 2.0 items. In *Proceedings of the 23rd International Conference on World Wide Web*. ACM, 771–782.

[27] Katja Hofmann, Krisztian Balog, Toine Bogers, and Maarten de Rijke. 2010. Contextual factors for finding similar experts. *Journal of the Association for Information Science and Technology* 61, 5 (2010), 994–1014.

[28] Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *SIGIR*. ACM, 50–57.

[29] Ruizhang Huang, Guan Yu, Zhaojun Wang, Jun Zhang, and Liangxing Shi. 2013. Dirichlet process mixture model for document clustering with feature partition. *IEEE Trans. Knowl. Data Eng.* 8, 25 (2013), 1748–1759.

[30] Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification* 1, 2 (1985), 193–218.

[31] Tomoharu Iwata, Shinji Watanabe, Takeshi Yamada, and Naonori Ueda. 2009. Topic tracking model for analyzing consumer purchase behavior. In *IJCAI*. AAAI, 1427–1432.

[32] Anil K. Jain. 2010. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters* 31, 8 (2010), 651–666.

[33] Ou Jin, Nathan N. Liu, Kai Zhao, Yong Yu, and Qiang Yang. 2011. Transferring topical knowledge from auxiliary long texts for short text clustering. In *CIKM*. ACM, 775–784.

[34] Alex Kulesza and Ben Taskar. 2010. Structured determinantal point processes. In *NIPS*. 1171–1179.

[35] Alex Kulesza and Ben Taskar. 2012. Determinantal point processes for machine learning. *Foundation & Trends in Machine Learning* 5, 2–3 (2012), 123–286.

[36] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. 2010. What is Twitter, a social network or a news media? In *WWW*. ACM, 591–600.

[37] Jure Leskovec, Kevin J. Lang, Anirban Dasgupta, and Michael W. Mahoney. 2009. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics* 6, 1 (2009), 29–123.

[38] Fangtao Li, Chao Han, Minlie Huang, Xiaoyan Zhu, Ying-Ju Xia, Shu Zhang, and Hao Yu. 2010. Structure-aware review mining and summarization. In *COLING*. ACL, 653–661.

[39] Ian Li, Yi Tian, Qiang Yang, and Ke Wang. 2001. Classification pruning for web-request prediction. In *WWW*. ACM.

[40] Shangsong Liang, Fei Cai, Zhaochun Ren, and Maarten de Rijke. 2016. Efficient structured learning for personalized diversification. *IEEE Transactions on Knowledge and Data Engineering* 28, 11 (2016), 2958–2973.

[41] Shangsong Liang and Maarten de Rijke. 2015. Burst-aware data fusion for microblog search. *Information Processing & Management* 51, 2 (2015), 83–113.

[42] Shangsong Liang, Zhaochun Ren, and Maarten de Rijke. 2014a. Fusion helps diversification. In *SIGIR*. ACM, 303–312.

[43] Shangsong Liang, Zhaochun Ren, and Maarten de Rijke. 2014b. Personalized search result diversification via structured learning. In *KDD*. ACM, 751–760.

[44] Shangsong Liang, Zhaochun Ren, Wouter Weerkamp, Edgar Meij, and Maarten de Rijke. 2014c. Time-aware rank aggregation for microblog search. In *CIKM*. ACM, 989–998.

[45] Shangsong Liang, Emine Yilmaz, and Evangelos Kanoulas. 2016. Dynamic clustering of streaming short documents. In *KDD*. ACM, 995–1004.

[46] James B. MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. 281–297.

[47] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

[48] Julian J. McAuley and Jure Leskovec. 2012. Learning to discover social circles in ego networks. In *NIPS*. 548–56.

[49] Edgar Meij, Wouter Weerkamp, and Maarten de Rijke. 2012. Adding semantics to microblog posts. In *WSDM*. ACM, 563–572.

[50] Thomas P. Minka. 2000. *Estimating a Dirichlet Distribution*. Technical Report. Microsoft Research.

[51] Alan Mislove, Bimal Viswanath, Krishna P. Gummadi, and Peter Druschel. 2010. You are who you know: Inferring user profiles in online social networks. In *WSDM*. ACM, 251–260.

[52] Bamshad Mobasher, Robert Cooley, and Jaideep Srivastava. 1999. Creating adaptive web sites through usage-based clustering of URLs. In *IEEE KDEX Workshop*. IEEE.

[53] Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning* 2–3, 39 (2000), 103–134.

[54] Michael J. Paul, ChengXiang Zhai, and Roxana Girju. 2010. Summarizing contrastive viewpoints in opinionated text. In *EMNLP*. ACL, 66–76.

[55] Marco Pennacchiotti and Ana-Maria Popescu. 2011. A machine learning approach to Twitter user classification. In *ICWSM*. AAAI, 281–288.

[56] Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text and web with hidden topics from large-scale data collections. In *WWW*. ACM, 91–100.

[57] Altaf Rahman and Vincent Ng. 2011. Narrowing the modeling gap: A cluster-ranking approach to coreference resolution. *Journal of Artificial Intelligence Research* 40 (2011), 469–521.

[58] Aniket Rangrej, Sayali Kulkarni, and Ashish V. Tendulkar. 2011. Comparative study of clustering techniques for short text documents. In *WWW*. ACM, 111–112.

[59] Zhaochun Ren and Maarten de Rijke. 2015. Summarizing contrastive themes via hierarchical non-parametric processes. In *SIGIR*. ACM, 93–102.

[60] Zhaochun Ren, Oana Inel, Lora Aroyo, and Maarten de Rijke. 2016. Time-aware multi-viewpoint summarization of multilingual social text streams. In *CIKM*. ACM, 387–396.

[61] Zhaochun Ren, Shangsong Liang, Edgar Meij, and Maarten de Rijke. 2013. Personalized time-aware tweets summarization. In *SIGIR*. ACM, 513–522.

[62] Zhaochun Ren, Maria-Hendrike Peetz, Shangsong Liang, Willemijn van Dolen, and Maarten de Rijke. 2014. Hierarchical multi-label classification of social text streams. In *SIGIR*. ACM, 213–222.

[63] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. 2004. The author-topic model for authors and documents. In *UAI*. AUAI, 487–494.

[64] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *WWW*. ACM, 851–860.

[65] Milad Shokouhi. 2013. Learning to personalize query auto-completion. In *SIGIR*. ACM, 103–112.

[66] Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. 2000. Web usage mining: Discovery and applications of usage patterns from web data. In *SIGKDD Explorations Newsletter*. ACM, 12–23.

[67] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. 2012. Hierarchical Dirichlet processes. *J. Amer. Statist. Assoc.* (2012), 1566–1581.

[68] Oren Tsur, Adi Littman, and Ari Rappoport. 2013. Efficient clustering of short messages into general domains. In *ICWSM*. AAAI, 621–630.

[69] Ibrahim Uysal and W. Bruce Croft. 2011. User oriented tweet ranking: A filtering approach to microblogs. In *CIKM*. ACM, 2261–2264.

[70] Christophe Van Gysel, Maarten de Rijke, and Evangelos Kanoulas. 2016a. Learning latent vector spaces for product search. In *CIKM*. ACM, 165–174.

[71] Christophe Van Gysel, Maarten de Rijke, and Marcel Worring. 2016b. Unsupervised, efficient and semantic expertise retrieval. In *WWW*. ACM, 1069–1079.

[72] Jan Vosecky, Kenneth Wai-Ting Leung, and Wilfred Ng. 2014. Collaborative personalized Twitter search with topic-language models. In *SIGIR*. ACM, 53–62.

[73] Hanna M. Wallach, David M. Mimno, and Andrew McCallum. 2009. Rethinking LDA: Why priors matter. In *NIPS*. 1973–1981.

[74] Xuerui Wang and Andrew McCallum. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In *KDD*. ACM, 424–433.

[75] Xing Wei, Jimeng Sun, and Xuerui Wang. 2007. Dynamic mixture models for multiple time-series. In *IJCAI*. AAAI, 2909–2914.

[76] Shuo Xu, Qingwei Shi, Xiaodong Qiao, Lijun Zhu, Han Zhang, Hanmin Jung, Seungwoo Lee, and Sung-Pil Choi. 2014. A dynamic users' interest discovery model with distributed inference algorithm. *International Journal of Distributed Sensor Networks* 10, 4 (2014), 1–11.

[77] Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. 2013. A biterm topic model for short texts. In *WWW*. ACM, 1445–1456.

[78] Jie Yin. 2013. Clustering microtext streams for event identification. In *IJCNLP*. ACL, 719–725.

[79] Jianhua Yin and Jianyong Wang. 2014. A Dirichlet multinomial mixture model-based approach for short text clustering. In *KDD*. ACM, 233–242.

[80] Guan Yu, Ruizhang Huang, and Zhaojun Wang. 2010. Document clustering via Dirichlet process mixture model with feature selection. In *KDD*. ACM, 763–772.

[81] Ke Zhai and Jordan Boyd-Graber. 2013. Online latent Dirichlet allocation with infinite vocabulary. In *ICML*. ACM, 561–569.

[82] Wayne Xin Zhao, Jing Jiang, Jing He, Yang Song, Palakorn Achananuparp, Ee-Peng Lim, and Xiaoming Li. 2011. Topical keyphrase extraction from Twitter. In *ACL*. ACL, 379–388.

[83] Yukun Zhao, Shangsong Liang, Zhaochun Ren, Jun Ma, Emine Yilmaz, and Maarten de Rijke. 2016. Explainable user clustering in short text streams. In *SIGIR*. ACM, 155–164.

Received September 2016; revised January 2017; accepted March 2017