Explainable User Clustering in Short Text Streams

Yukun Zhao[†*] yukunzhao.sdu@gmail.com
Shangsong Liang[‡*] shangsong.liang@ucl.ac.uk
Zhaochun Ren[§*] z.ren@uva.nl
Jun Ma[†] majun@sdu.edu.cn
Emine Yilmaz[‡] emine.yilmaz@ucl.ac.uk
Maarten de Rijke[§] derijke@uva.nl

[†]Shandong University, Jinan, China
[‡]University College London, London, United Kingdom
[§]University of Amsterdam, Amsterdam, The Netherlands

ABSTRACT

User clustering has been studied from different angles: behavior-based, to identify similar browsing or search patterns, and content-based, to identify shared interests. Once user clusters have been found, they can be used for recommendation and personalization. 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 short text streams. 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 (or UCT for short). UCT adaptively tracks changes of each user's time-varying topic distribution based both on the short texts the user posts during a given time period and on the previously estimated distribution. To infer changes, we propose a Gibbs sampling algorithm where a set of word-pairs from each user is constructed for sampling. 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 clustering model compared to state-of-the-art baselines.

Keywords

User clustering; Short text processing; User topic modeling

1. INTRODUCTION

With the rising popularity of social media, hundreds of millions of active users are sharing short texts on microblogging platforms such as Twitter[1] and Sina Weibo.[2] A good understanding of users' dynamic preferences is important for the design of applications that cater for users of such platforms, such as personalized microblog search, twitter summarization, and computational advertising. In this paper, we study the problem of *user clustering in the context of short text streams*, where users are taken 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 [5, 18, 24] 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. [24] constructs vector matrices for URLs and users and then utilizes K-means [16] to cluster users based on browsing vectors. These 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 for short, to tackle the problem of dynamic user clustering in short text streams. Traditional topic models such as probabilistic latent semantic indexing (PLSI) [12], latent Dirichlet allocation (LDA) [3], author topic models [31, 37] or the user interest topic model [21], have been widely used to uncover topics of documents and users. These topic models do not work well in the context of short text streams due to the problem of sparsity. How to utilize topic distributions for user clustering is still an open problem.

Inspired by previous work [5–7, 19, 20, 22, 28, 29], 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 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

[*]These three authors contributed equally to the paper.

SIGIR '16, July 17 - 21, 2016, Pisa, Italy

ISBN 978-1-4503-4069-4/16/07...$15.00

DOI: http://dx.doi.org/10.1145/2911451.2911522

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

time, although their interests may differ at previous times. To the best of knowledge, we are the first to perform dynamic user clustering in short text streams 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, 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 our UCT clustering model.

The main contributions of our work are:

(1) We propose the task of dynamically clustering users in the context of short text streams.
(2) We propose a dynamic multinomial Dirichlet mixture user clustering topic model, UCT, to address the user clustering task, where users' time-varying topic distributions can be captured.
(3) We propose a collapsed Gibbs sampling algorithm for the inference of dynamic users' topic distributions in the context of short text streams, where we tackle the problem of word co-occurrence sparsity.
(4) Our proposed clustering model can effectively cluster previously seen users as well as users who newly arrive in the stream.
(5) 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.

2. RELATED WORK

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

2.1 User clustering and text clustering

State-of-the-art research on user clustering mainly focuses on web user clustering [4, 18, 24, 32]. These papers study users' access information from logged server data including query and click data and then uncovers clusters of these users that exhibit similar information needs. For instance, Buscher et al. [4] 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, has grouped users by expertise [1, 11]; recent advances in distributed representation learning have given rise to new types of joint topic and entity representations [34] but, so far, these have not been used for user clustering yet. 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 short text streams such as Twitter or Weibo. In this paper, we aim at dynamically clustering users in short text streams.

Another relevant line of work is text clustering. Yu et al. [40] and Huang et al. [13] propose a Dirichlet process mixture with feature selection model (DPMFS) and a Dirichlet process mixture with feature partition model (DPMFP) for normal document clustering, respectively. They compare DPMFP with four other clustering models: EM text classification (EM-TC) [25], K-means [16], LDA [3] and exponential-family approximation of the Dirichlet compound multinomial distribution (EDCM) [8]; they find that DPMFP performs best. In the context of short text documents, Rangrej et al. [27] compare three clustering algorithms including K-means, Singular Value Decomposition and Affinity Propagation [9] 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. Tsur et al. [33], Yin [38], and Yu et al. [40] focus on the problem of online clustering of a stream of tweets. They all 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 [39] 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. All of these algorithms aim at clustering short documents—the problem of dynamically clustering users in the context of short text streams has so far been ignored, however.

2.2 Topic modeling

Probabilistic topic models, such as PLSI [12] and LDA [3], aim to analyze latent topics of documents. Various LDA-type topic models have been proposed. The author topic model [31] 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 short text streams: model users as distributions over topics inferred from their tweets and then cluster users based on their topic distributions.

Various dynamic topic models have been proposed to track changes of topics in streams. The dynamic topic model (DTM) [2] 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 dynamic mixture model (DMM) [36] considers a single dynamic sequence of documents, which corresponds to a single topic over time. The topic tracking model (TTM) [15] focuses on tracking time-varying consumer behavior, in which consumers' interests change over time. The topic over time model (ToT) [35] 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. 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 short text streams.

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. [26] train latent topics from large external resources. Jin et al. [17] learn hidden topics on short texts via transfer learning from auxiliary long text data. Ren et al. [30] apply a document expansion method that consists of entity linking and sentence extraction. Chen and Liu [6] retain the results learned in the past and using them to help future learning. Cheng et al. [7] extract bi-terms in each tweet to capture word co-occurrence explicitly for enhancing the performance of short text topic modeling. Again, unlike UCT, these algorithms aim at working with a static collection of documents only.

We work with short text streams and propose a dynamic Dirichlet multinomial mixture user clustering topic model, 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 clusters of users in the context of short text streams.

3. PRELIMINARIES

In this section, we introduce the main notations and task to be addressed in this paper.

Table 1 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, a year etc. Also, z is a topic and K is the number of topics we infer for our UCT model; w is a word in a tweet and b represents a word-pair extracted from a tweet.

Table 1: Main notation used in UCT.

Notation	Gloss	Notation	Gloss
u	user	t	time slice
z	topic	K	number of topics
w	word	b	word-pair
$\omega_{t,u}$	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		
α_t	the parameter of user Dirichlet prior		
β_t	the parameter of token 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		

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 model training process, 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 times of w is assigned to topic z, respectively. $\omega_{t,u}$ is a cluster to which user u belongs at time t, and the cluster $\omega_{t,u}$ can be changed over time as we assume the user's interest $\theta_{t,u}$ is time-varying.

The task we address is to dynamically track clusters of users over time in the context of short text streams such that users in the same cluster share similar interests. Specifically, for each time period t, given a set of users $\mathbf{U}_t = \{u_1, u_2, \ldots, u_m\}$ at time 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 $\omega_{t,u}$ being the cluster to which user u belongs at t.

4. METHOD

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

4.1 Overview

We use Twitter as our default setting of short text streams 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 $t \in \{\ldots, T-1, T\}$, 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 short text streams 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.

Algorithm 1: Overview of the algorithm for user clustering in short text streams.

Input : A set of users $\mathbf{U}_t$ along with their published tweets $\mathbf{D}_t$
Output: cluster of each user $\omega_{t,u}$
1 Extract a set of word-pairs $\mathbf{B}_{t,u}$ for each user u, see §4.2
2 Use UCT model to track each user's interests as $\theta_{t,u}$, see §4.3
3 Cluster users based on their interests distribution $\theta_{t,u}$, see §4.4

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

Based on the multinomial distribution $\theta_{t,u}$, we explore the cluster of users using K-means clustering [16] (see step 3 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 [3, 15, 35] 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 paper, we do not use the words in tweets (our documents) to directly infer topics for users due to the limited length of tweets.

In order to tackle the lack of context in modeling users' interests, we explicitly consider word correlations via co-occurring words in a word-pair instead of a whole tweet, where a word-pair is a set of two order-exchangeable words being assigned to the same topic. Specifically, after removing stop words and apply 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 [6, 7], we regard each tweet as an individual context unit, in which word-pairs in a tweet 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 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 subsection shows how to use the word-pair set $\mathbf{B}_{t,u}$ to model the user's interests.

4.3 Dynamic user clustering topic model

In this subsection, we detail our dynamic multinomial Dirichlet mixture user clustering topic model (UCT). UCT captures a user's interests at time t as $\theta_{t,u}$, that is, as a 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, week, month, quarter or even year.

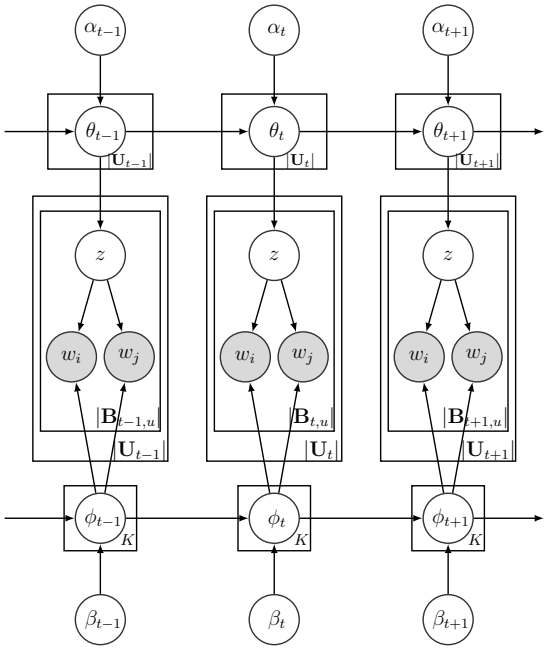


Figure 1: Graphical representation of our dynamic user clustering topic model, UCT. Shaded nodes represent observed variables.

Fig. 1 shows a graphical representation of our UCT model, where shaded and unshaded nodes indicate observed and latent variables, respectively. At 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 ϕ_t, and infer current users' interests θ_t. From Fig. 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 model a user's interests by combining their topic distribution obtained during the previous time slice $t-1$ as prior knowledge and newly arriving observed data for inferring current model for the user's interests. 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. The K-dimensional variable $\theta_{t,u} = \{\theta_{t,u,z}\}_{z=1}^{K}$ has the following probability density:

$$P(\theta_{t,u} \mid \theta_{t-1,u}, \alpha_t) = \frac{\Gamma(\sum_z \theta_{t-1,u,z} + \alpha_{t,z})}{\prod_z \Gamma(\theta_{t-1,u,z} + \alpha_{t,z})} \cdot \prod_z \theta_{t,u,z}^{\theta_{t-1,u,z}+\alpha_{t,z}-1}, \quad (3)$$

where $\Gamma(x)$ is a Gamma function. In contrast with static topic models [3, 31], the Dirichlet prior changes from α to $\theta_{t-1,u}+\alpha_t$, 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 current time period t by using the following Dirichlet distribution:

$$P(\phi_{t,z} \mid \phi_{t-1,z}, \beta_t) = \frac{\Gamma(\sum_v \phi_{t-1,z,v} + \beta_{t,v})}{\prod_v \Gamma(\phi_{t-1,z,v} + \beta_{t,v})} \cdot \prod_v \phi_{t,z,v}^{\phi_{t-1,z,v}+\beta_{t,v}-1}, \quad (4)$$

The topic-words distributions are considered to be inferred through priors $\phi_{t-1,z} + \beta_t$. We estimate α_t and β_t for each time period instead of using simple 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 Fig. 1 we know that topic z is related with a distribution of words with the multinomial distribution $\phi_{t,z} = \{\phi_{t,z,w} \mid w \in \mathbf{V}_t\}$. 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 θ_{t-1}, ϕ_{t-1} at the previous time period and the priors β_t, α_t are utilized for inferring the current distribution θ_t and ϕ_t. The generative process is as follows:

i. Draw K multinomials $\phi_{t,z}$ from Dirichlet priors β_t and ϕ_{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 $\phi_{t,z_{t,u,b},w_i}$;
   (c) Draw another word $w_j \in b$ from multinomial $\phi_{t,z_{t,u,b},w_j}$;

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:

$$P(b \mid t, z) = P(w_i \mid t, z)P(w_j \mid t, z), \quad (5)$$

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

$$P(b \mid t) = \sum P(z \mid t)P(w_i \mid t, z)P(w_j \mid t, z). \quad (6)$$

Inference is intractable in this model. Following [3, 10, 15, 36, 39], 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 ϕ_t and θ_t, analytically capturing the uncertainty associated with them. In this way we facilitate the sampling, i.e., we need not sample ϕ_t and θ_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 1). The input of our Gibbs sampling algorithm is $\mathbf{B}_t$ (all users' word-pairs sets at time slice t) and the output consists of all users' interests distributions over topics at current time t. For the initialization of our Gibbs sampling, we randomly assign a topic $z = z_{t,u,b}$ 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 the joint probability of the 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. Using the chain rule, we obtain the conditional probability conveniently as follows:

$$\begin{aligned} P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi_{t-1}}, \mathbf{\Theta_{t-1}}, \alpha_t, \beta_t) \\ \propto \frac{(n_{t,z,w_i} + \phi_{t-1,z,w_i} + \beta_{t,w_i} - 1)}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v}) - 1)} \\ \times \frac{(n_{t,z,w_j} + \phi_{t-1,z,w_j} + \beta_{t,w_j} - 1)}{(\sum_{v=1}^{V_t}(n_{t,z,v} + \phi_{t-1,z,v} + \beta_{t,v}) - 1)} \end{aligned} \quad (7)$$

Algorithm 2: Gibbs Sampling for UCT

Input : $K, N, t, \mathbf{B}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_{t-1}, \beta_{t-1}$
Output: multinomial parameter θ_t and ϕ_t

Initialize $z_{t,u,b}, m_{t,u,z}, n_{t,z,w}$ 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 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 $iter = 1, \ldots, N$ **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_b from $P(z_{t,u,b} = z \mid \mathbf{B}_t, \mathbf{Z}_{t,-(u,b)}, \mathbf{U}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$; see (7)
      update $z_{t,u,b}, m_{t,u,z}, n_{t,z,w_i}, n_{t,z,w_j}$
compute the parameters θ_t and ϕ_t using equation (8).

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

where $m_{t,u,z}$ represents the number of word-pairs published by user u and assigned to topic z, and $n_{t,z,w}$ represents the number of times word w is assigned to topic z. 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 updating rules in fixed-point iterations for obtaining these two Dirichlet priors,

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

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

where $\Psi(x) = \frac{\partial \log \Gamma(x)}{x}$ is a Digamma function, and

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

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 $\mathbf{\Phi}_t$, respectively:

$$\theta_{t,u,z} = \frac{m_{t,u,z} + \theta_{t-1,u,z+}\alpha_{t,z} - 1}{\sum_{z=1}^{K}(m_{t,u,z} + \theta_{t-1,u,z} + \alpha_{t,z}) - 1},$$
$$\phi_{t,z,w} = \frac{n_{t,z,w} + \phi_{t-1,z,w} + \beta_{t,w} - 1}{\sum_{w=1}^{V_t}(n_{t,z,w} + \phi_{t-1,z,w} + \beta_{t,w}) - 1}. \quad (8)$$

4.4 Clustering users

Clustering previously seen users. After we have determined each user's $u \in \mathbf{U}_t$ dynamic topic distribution at time t, $\theta_{t,u}$ (obtained by (8)), we use K-means [16] 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 $u_{\text{new}} \notin \mathbf{U}_t$. 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 §4.2. We compute the probability of the user u_{new} being interested in topic z at t, i.e., $\theta_{t,u_{\text{new}},z}$, as:

$$P(z \mid t, u_{\text{new}}) = \prod_{b \in \mathbf{B}_{t,u_{\text{new}}}} P(z \mid t, b) P(b \mid t, u_{\text{new}}), \quad (9)$$

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

$$\begin{aligned} P(z \mid t, b) &= \frac{P(w_i \mid t, z) P(w_j \mid t, z) P(z \mid t)}{P(b \mid t)} \\ &= \frac{P(w_i \mid t, z) P(w_j \mid t, z) P(z \mid t)}{\sum_z P(z \mid t) P(w_i \mid t, z) P(w_j \mid t, z)} \\ &= \frac{P(z \mid t) \phi_{t,z,w_i} \phi_{t,z,w_j}}{\sum_z P(z \mid t) \phi_{t,z,w_i} \phi_{t,z,w_j}}, \end{aligned} \quad (10)$$

where $P(w \mid t, z)$ is the probability of word w associated with topic z at t, i.e., $\phi_{t,z,w}$, and $P(z \mid t)$ is the probability of topic z at t. We obtain $P(z \mid t)$ for (10) as:

$$P(z \mid t) = \frac{n_t(z, w)}{n_t(w)}, \quad (11)$$

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 (9) as:

$$P(b \mid t, u_{\text{new}}) = \frac{n_{t,u_{\text{new}}}(b)}{\sum_b n_{t,u_{\text{new}}}(b)}, \quad (12)$$

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

Finally, after applying (10), (11) and (12) to (9), we obtain the new user's interests $\theta_{t,u_{\text{new}}}$. We then group this user into a cluster $\omega_{t,u}$ where they share most interests with other users in the cluster:

$$\omega_{t,u_{\text{new}}} = \arg\max_{\omega_{t,u}} \sum_{u \in \omega_{t,u}} \frac{\cos(\theta_{t,u}, \theta_{t,u_{\text{new}}})}{|\omega_{t,u}|}, \quad (13)$$

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; §5.1 lists our research questions; §5.2 describes our dataset; §5.3 and §5.4 list the baselines and metrics for evaluation, respectively;

5.1 Research questions

We list the research questions that guide the remainder of the paper:

RQ1 Does our dynamic user clustering method UCT outperform state-of-the-art baseline methods? (See §6.1)
RQ2 What is the impact of the different time slices in our dynamic user clustering method? (See §6.2)
RQ3 What is the quality of the topical representation inferred by our user clustering topic model? (See §6.3)

RQ4 Can we capture the dynamics of users' interests and make the clustering results produced by our proposed dynamic user topic model explainable? (See 6.4)

5.2 Dataset

In order to answer our research questions, we work with a dataset collected from Twitter. The data set contains 1,375 active users 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 short text streams and manually judge the clusters of the 1,375 users based on the content of their published tweets. We obtain ground truth clusters for 5 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 times 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 pre-processing, we remove stop words and apply Porter stemming using the Lemur toolkit.[3]

5.3 Baselines

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

K-means. This is a traditional clustering algorithm [16]. It represents users by TF-IDF vectors and categorizes them into different clusters based on their TF-IDF vector similarities.

GSDMM. This is a Dirichlet multinomial mixture model-based approach for short text clustering [39]. It represents each short document through a single topic and groups each user into a cluster that contains most of his tweets.

Latent Dirichlet Allocation (LDA). This model infers topic distributions specific to each document via the latent dirichlet allocation (LDA) model.

Author Topic Model (AuthorT). This model [31] 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.

Dynamic topic model (DTM). This model [2] utilizes a Gaussian distribution for inferring topic distribution of long text documents in streams.

Topic over time model (ToT). This model [35] normalizes timestamps of long documents in a collection and then infers topics distribution for each document.

Topic tracking model (TTM). This model [15] 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 topic models, i.e., 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 equal to the number of topics.

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

5.4 Evaluation metrics

Given the number of clusters P and the number of output clusters Q in the ground truth, 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 to evaluate our experimental results, all of which are widely used in the literature [7, 39].

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 2-combinations from a given set $\omega_i \cap c_j$ and ω_i, respectively. Obviously, a higher precision indicates better user clustering performance.

Normalized Mutual Information (NMI) [23]. NMI is a measure that allows us to make the trade-off 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 $I(\Omega; \mathcal{C})$ denotes 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 $E(\mathcal{C})$:

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

where n is the total number of users. Note that when $\mathcal{C}$ is equal to Ω, NMI reaches 1, its maximum value. Larger NMI value indicate better clustering performance.

Adjusted Rand Index (ARI) [14, 23]. 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 [14]. The maximum value is 1 for exact match; larger values mean better performance for clustering. ARI$(\mathcal{C}, \Omega)$ is computed as follows, where n is the total number of users.

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

The three metrics introduced so far are for evaluating the performance of user clustering, whereas the following metric is for evaluating the quality of topic representations of users in clusters.

H-score [7]. 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 [7] metric, which is computed as:

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

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

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

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

where $\text{dis}(u_i,u_j)$ is the symmetric Kullback-Leibler distance of topic distributions of user u_i and user u_j. 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.

We report the Precison, NMI and ARI scores of all eight methods listed above to evaluate clustering performance. Importantly, to evaluate the quality of topical representations, we report H-scores of all baseline methods except GSDMM. We cannot compute H-scores for GSDMM as it assumes each document to be 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.

6. RESULTS AND ANALYSIS

In the following subsections 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 short text streams, and compare UCT with a traditional clustering method, K-means, GSDMM, which integrates a state-of-the-art clustering topic model for short documents in a static set, and three dynamic user clustering models, DTM, ToT and TTM (see §5.3). 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, NMI and ARI values of the eight methods by averaging the performance across the two parts.

Fig. 2 shows the performance of UCT and the baselines in terms of Precision, ARI and NMI. First, we see that UCT performs significantly better than K-means, GSDMM and the five topic models on all the metrics, which demonstrates the effectiveness of our model for user clustering. UCT and the other 5 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, 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 help 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.

Table 2: The average number of tweets a user published during a week, a month, a quarter, half a year, a year. Plus the average number of word-pairs extracted for each user.

	a week	a month	a quarter	half a year	a year
#tweets	16	111	220	418	744
#word-pairs	1012	3586	9199	14348	29810

UCT significantly outperforms all baselines, and this finding confirms that the way UCT infers dynamic topic distributions of short documents in streams improve the performance of user clustering.

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. Fig. 3 shows the evaluation results in terms of Precision, 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.

UCT always outperforms LDA, AuthorT, DTM, TTM, ToT, 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 distribution 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 three 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.

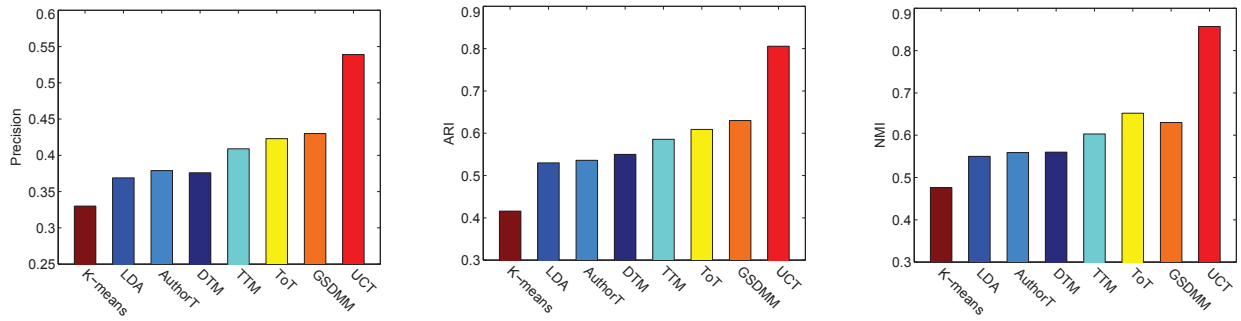
To further understand why UCT and the baseline methods increase their performance when the length of the time period use for evaluation increases, we provide an analysis of word co-occurrence patterns in different time periods. Statistics of the number of tweets users published in different time periods are shown in Table 2. 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 1012, which is the number of word-pairs. A larger number of word-pairs help to better infer topic distributions in Gibbs sampling. The longer the time period, the more word-pairs can be utilized in our Gibbs sampling for the topic inference.

6.3 Quality of topical representations

We now address research question **RQ3**. To assess the quality of topics extracted by UCT, we compare UCT and the baseline methods. Fig. 4 shows the comparison of the performance of UCT and the baselines in terms of H-score. When compute the H-scores for evaluating the quality of topical representation in UCT and the baselines, we use the quarterly ground-truth user clustering results.

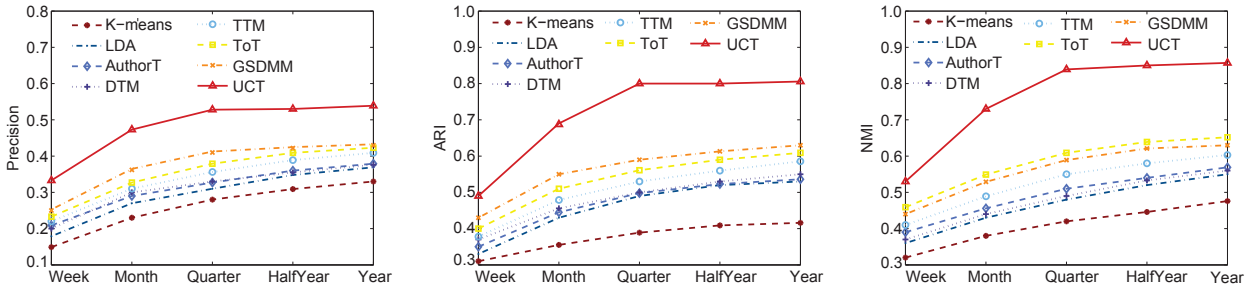
It is clear from Fig. 4 that UCT obtains a significantly smaller H-score compared to all other six models,[4] which indicates that the average inter-cluster distance is small compared to the aver-

[4]Recall that we cannot calculate the H-score for GSDMM as it assumes each document to be in only a single topic



(a) Precision (b) ARI (c) NMI

Figure 2: Clustering performance of DCT and the baselines using a time period of a quarter. The performance is evaluated using Precision, ARI and NMI, respectively.



(a) Precision (b) ARI (c) NMI

Figure 3: 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, ARI and NMI, respectively.

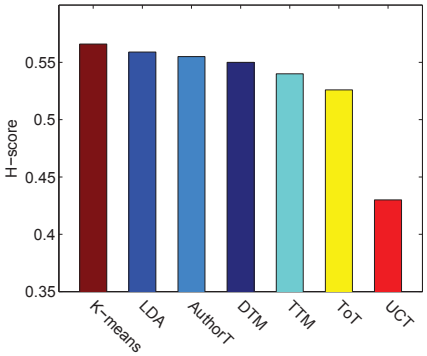


Figure 4: Evaluation results for the quality of topic representations for UCT and the baselines, using the H-score metric and time periods of a quarter.

age intra-cluster distance. A smaller H-score means that the topical representation of users is more similar to that labeled manually (each 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 $P(w \mid t, z) \cdot P(z \mid t, u)$, i.e., $\phi_{t,z,w} \cdot \theta_{t,u,z}$, associated with the user; 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 associated with the cluster, which is computed by $\frac{1}{|c|}\sum_{u \in c} P(w \mid t, z)P(z \mid t, u)$, i.e., $\frac{1}{|c|}\sum_{u \in c} \phi_{t,z,w} \cdot \theta_{t,u,z}$. Then, the top-$N$ words that obtain the highest probabilities are selected to represent the cluster. Table 3 shows the top 15 words extracted from a cluster and two users in this cluster. 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 see that the two users in the same cluster share more similar interests like "kids, immigration, community, education." 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.

6.4 Dynamic topic representation of users

Finally, we address research question **RQ4**. As UCT captures each user's dynamic topic distribution, we investigate the content of the users' interests. We conduct 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 15 words in Table 4, where the words are selected from the 10 most probable topics of the user and then the 20 most probable words for each topic.

As seen in Table 4, 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.", "prototype, android, web, internet, etc." in the following two quarters, respectively. As time moves on in 2015, their interests change to "Russia, government, designer, app, problem and etc." and "proto-

Table 3: Top 15 words representing a cluster and two users extracted by UCT and ToT, respectively. 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.

UCT	ToT
kid color community robbery spiritual immigration alert kicker education child violence star chocolate girl gay	media kid toast immigration dog campaign alert Mexico fan advertisement bet slide John image people
kid unit robbery ride arrest education star police civilian flat immigration internal rule city community	kid education internal photo fan media ad community game score rio toast process http process
immigration kid child kicker process community soccer color police officer rule city game violence gay	kid process soccer child media people dog Mexico reason score song education robbery fan star

type, 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" that are about business, politics and some sports. Then they talk more about college football and feminism and equality with words like "TXST, star, game, campus, feminism, equality and 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 of dynamic clustering is explainable and understandable in the context of short text streams.

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 user clustering topic model, UCT, 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.

For evaluation purposes, we have compared the performance of UCT to that of a traditional clustering algorithm, K-means, non-dynamic topic models, LDA, the author topic model, and state-of-the-art dynamic topic models, viz. DTM, ToT and TTM. Our experimental results demonstrate the clustering effectiveness of our model for user clustering in the context of short document streams. 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 previously captured topic distributions of users for inferring a user's current interests, and to improve efficiency of the Gibbs sampling algorithm as the process is time-consuming.

Acknowledgements. We thank Juan Echeverría Guzman at UCL for collecting the dataset for us. This work was supported by the National Natural Science Foundation of China under Grant No. 61272240 and 61103151, the Big Data Institute, University College London, Ahold, Amsterdam Data Science, the Bloomberg Research Grant program, the Dutch national program COMMIT, Elsevier, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr 312827 (VOX-Pol), the ESF Research Network Program ELIAS, the Royal Dutch Academy of Sciences (KNAW) under the Elite Network Shifts project, the Microsoft Research Ph.D. program, the Netherlands eScience Center under project number 027.012.105, the Netherlands Institute for Sound and Vision, the Netherlands Organisation for Scientific Research (NWO) under project nrs 727.011.005, 612.001.116, HOR-11-10, 640.006.013, 612.066.930, CI-14-25, SH-322-15, 652.002.001, 612.001.551, the Yahoo Faculty Research and Engagement Program, and Yandex. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

8. REFERENCES

[1] K. Balog and M. de Rijke. Finding similar experts. In *SIGIR*, pages 821–822. ACM, 2007.

[2] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML*, pages 113–120, 2006.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Machine Learning research*, 3(4–5):993–1022, 2003.

[4] G. Buscher, R. W. White, S. Dumais, and J. Huang. Large-scale analysis of individual and task differences in search result page examination strategies. In *WSDM*, pages 373–382. ACM, 2012.

[5] W. Chen, J. Wang, Y. Zhang, H. Yan, and X. Li. User based aggregation for biterm topic model. In *ACL*, pages 489–494, 2015.

[6] Z. Chen and B. Liu. Mining topics in documents: standing on the shoulders of big data. In *KDD*, pages 1116–1125. ACM, 2014.

[7] X. Cheng, X. Yan, Y. Lan, and J. Guo. A biterm topic model for short texts. In *WWW*, pages 1445–1456. ACM, 2013.

[8] C. Elkan. Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution. In *ICML*, pages 289–296, 2006.

[9] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.

[10] T. L. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101(suppl 1):5228–5235, 2004.

[11] K. Hofmann, K. Balog, T. Bogers, and M. de Rijke. Contextual factors for finding similar experts. *J. Am. Soc. Inf. Sci. Techn.*, 61(5):994–1014, May 2010.

[12] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, pages 50–57. ACM, 1999.

Table 4: Top 15 words representing two users' interests over time, covering five quarters from April 2014 to May 2015. The first row shows the top 15 words per quarter to represent a user whose interests center on the design of apps. The second row shows the top 15 words per quarter to represent another user's 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.

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 prototyping tweet feature ios team coding perspective lane motorway car	app kid store dog dialog browser design scenario book mobile element email programming night inspiration	prototype android web internet design house breakfast ios film tweet media social people Russia license	Russia government email designer app photo problem engineer mobile strategy smartphone product team hardware ios	prototype apple music ios mac video entrepreneur correlation point interaction task screen years amp reason
center partner WalMart TXST mall David belt offer game player improvement blue enforcement county Oklahoma	TXST star guy fan game night campus sports basketball member grace feminism equality score post	ESPN TXST starSports community StarOpinion StarNews student semester bowl game university star tonight traffic conference	TXST state respect feminism community Texas sexism opinion game campus podcast America nation government student	violence TXST victim responsibility police official opinion state campus respond columnist follow season lecture Texas

[13] R. Huang, G. Yu, Z. Wang, J. Zhang, and L. Shi. Dirichlet process mixture model for document clustering with feature partition. *IEEE Trans. Knowl. Data Eng.*, 8(25):1748–1759, 2013.

[14] L. Hubert and P. Arabie. Comparing partitions. *J. Classification*, 1(2):193–218, 1985.

[15] T. Iwata, S. Watanabe, T. Yamada, and N. Ueda. Topic tracking model for analyzing consumer purchase behavior. In *IJCAI*, pages 1427–1432, 2009.

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

[17] O. Jin, N. N. Liu, K. Zhao, Y. Yu, and Q. Yang. Transferring topical knowledge from auxiliary long texts for short text clustering. In *CIKM*, pages 775–784. ACM, 2011.

[18] I. Li, Y. Tian, Q. Yang, and K. Wang. Classification pruning for web-request prediction. In *WWW*. ACM, 2001.

[19] S. Liang and M. de Rijke. Burst-aware data fusion for microblog search. *Inf. Proc. Man.*, 51(2):83–113, 2015.

[20] S. Liang, Z. Ren, and M. de Rijke. Fusion helps diversification. In *SIGIR*, pages 303–312, 2014.

[21] S. Liang, Z. Ren, and M. de Rijke. Personalized search result diversification via structured learning. In *KDD*, pages 751–760. ACM, 2014.

[22] S. Liang, Z. Ren, W. Weerkamp, E. Meij, and M. de Rijke. Time-aware rank aggregation for microblog search. In *CIKM*, pages 989–998. ACM, 2014.

[23] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge university press, 2008.

[24] B. Mobasher, R. Cooley, and J. Srivastava. Creating adaptive web sites through usage-based clustering of urls. In *IEEE KDEX workshop*. IEEE, 1999.

[25] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 2–3(39):103–134, 2000.

[26] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi. Learning to classify short and sparse text and web with hidden topics from large-scale data collections. In *WWW*, pages 91–100. ACM, 2008.

[27] A. Rangrej, S. Kulkarni, and A. V. Tendulkar. Comparative study of clustering techniques for short text documents. In *WWW Companion*, pages 111–112. ACM, 2011.

[28] Z. Ren and M. de Rijke. Summarizing contrastive themes via hierarchical non-parametric processes. In *SIGIR*, pages 93–102, 2015.

[29] Z. Ren, S. Liang, and M. de Rijke. Personalized time-aware tweets summarization. In *SIGIR*, pages 513–522, 2013.

[30] Z. Ren, M.-H. Peetz, S. Liang, W. van Dolen, and M. de Rijke. Hierarchical multi-label classification of social text streams. In *SIGIR*, pages 213–222. ACM, 2014.

[31] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *UAI*, pages 487–494, 2004.

[32] J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan. Web usage mining: discovery and applications of usage patterns from web data. In *SIGKDD Explorations*, pages 12–23. ACM, 2000.

[33] O. Tsur, A. Littman, and A. Rappoport. Efficient clustering of short messages into general domains. In *ICWSM*, pages 621–630, 2013.

[34] C. Van Gysel, M. de Rijke, and M. Worring. Unsupervised, efficient and semantic expertise retrieval. In *WWW*, pages 1069–1079. ACM, 2016.

[35] X. Wang and A. McCallum. Topics over time: a non-markov continuous-time model of topical trends. In *KDD*, pages 424–433. ACM, 2006.

[36] X. Wei, J. Sun, and X. Wang. Dynamic mixture models for multiple time-series. In *IJCAI*, pages 2909–2914, 2007.

[37] S. Xu, Q. Shi, X. Qiao, et al. A dynamic users' interest discovery model with distributed inference algorithm. *IJDSN*, 2015:Article ID 280892, 2014.

[38] J. Yin. Clustering microtext streams for event identification. In *IJCNLP*, pages 719–725, 2013.

[39] J. Yin and J. Wang. A dirichlet multinomial mixture model-based approach for short text clustering. In *KDD*, pages 233–242. ACM, 2014.

[40] G. Yu, R. Huang, and Z. Wang. Document clustering via dirichlet process mixture model with feature selection. In *KDD*, pages 763–772. ACM, 2010.