Dynamic Clustering of Streaming Short Documents

Liang, S.; Yilmaz, E.; Kanoulas, E.

DOI
10.1145/2939672.2939748

Publication date
2016

Document Version
Final published version

Published in
KDD'16

License
Article 25fa Dutch Copyright Act

Link to publication

Citation for published version (APA):
Liang, S., Yilmaz, E., & Kanoulas, E. (2016). Dynamic Clustering of Streaming Short Documents. In *KDD'16: proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining : August 13-17, 2016, San Francisco, CA, USA* (pp. 995-1004). Association for Computing Machinery. https://doi.org/10.1145/2939672.2939748

Dynamic Clustering of Streaming Short Documents

Shangsong Liang[†]
shangsong.liang@ucl.ac.uk

Emine Yilmaz[†]
emine.yilmaz@ucl.ac.uk

Evangelos Kanoulas[‡]
e.kanoulas@uva.nl

[†]University College London, London, United Kingdom
[‡]University of Amsterdam, Amsterdam, The Netherlands

ABSTRACT

Clustering technology has found numerous applications in mining textual data. It was shown to enhance the performance of retrieval systems in various different ways, such as identifying different query aspects in search result diversification, improving smoothing in the context of language modeling, matching queries with documents in a latent topic space in ad-hoc retrieval, summarizing documents etc. The vast majority of clustering methods have been developed under the assumption of a static corpus of long (and hence textually rich) documents. Little attention has been given to streaming corpora of short text, which is the predominant type of data in Web 2.0 applications, such as social media, forums, and blogs. In this paper, we consider the problem of dynamically clustering a streaming corpus of short documents. The short length of documents makes the inference of the latent topic distribution challenging, while the temporal dynamics of streams allow topic distributions to change over time. To tackle these two challenges we propose a new dynamic clustering topic model - DCT - that enables tracking the time-varying distributions of topics over documents and words over topics. DCT models temporal dynamics by a short-term or long-term dependency model over sequential data, and overcomes the difficulty of handling short text by assigning a single topic to each short document and using the distributions inferred at a certain point in time as priors for the next inference, allowing the aggregation of information. At the same time, taking a Bayesian approach allows evidence obtained from new streaming documents to change the topic distribution. Our experimental results demonstrate that the proposed clustering algorithm outperforms state-of-the-art dynamic and non-dynamic clustering topic models in terms of perplexity and when integrated in a cluster-based query likelihood model it also outperforms state-of-the-art models in terms of retrieval quality.

CCS Concepts

•**Information systems** → **Clustering;**

Keywords

Clustering; Topic Models; Streaming Text; Cluster-based Retrieval

KDD '16, August 13 - 17, 2016, San Francisco, CA, USA

ISBN 978-1-4503-4232-2/16/08. . . $15.00

DOI: http://dx.doi.org/10.1145/2939672.2939748

1. INTRODUCTION

New media applications and the increasing prevalence of mobile devices have facilitated the collection and rapid dissemination of news and information by anyone connected to the Internet. Massive amounts of user generated content, often in the form of short text (e.g. microblog posts), typically clustered around real-life events [6, 20], are streaming in, and being consumed by interconnected users. Organizing large text collections around concise topics (or clusters) allows effective summarization and retrieval of information [5, 7, 13]. In the case of a rapidly streaming short text, however, traditional clustering algorithms are either not applicable, or do not tackle the temporal and sparse nature of the text corpus [7, 25]. In this paper we propose a dynamic clustering topic model method – DCT – for short-length streaming text and we demonstrate that it can effectively model both the temporal nature of topics in streaming text and the sparsity problem of short text, improving the performance of clustering and ad-hoc search.

One of the key challenges in clustering streaming data is the dynamic nature of topics (or clusters): topic distributions change with time, with previously salient topics "fading-off" and vice versa [1, 6, 7, 10, 22]. Therefore, techniques developed ought to allow for changes in the topic distribution along time. For example, Twitter posts about Apple Inc. on September 9, 2015, when Apple introduced *iPhone 6s plus*, are expected to be clustered around *Apple iPhone 6s plus*, while this may not be the case on December 3, 2015, when *Apple swift* was announced. The problem of clustering documents in streams has been widely investigated in the past [1, 6, 10, 22]. However, most of the previous work makes the assumption that the content of documents is rich enough to infer a per-document multinomial distribution of topics. The second key challenge in clustering streaming data is that this assumption does not hold for short text, as the number of words in each document is limited, which prohibits the accurate inference of a topic distribution over the document. Our method tackles the two challenges by introducing a collapsed Gibbs sampling algorithm that (a) assigns a single topic to all the words of a short document, and (b) uses the inferred topic distribution of past documents as a prior of the topic distribution of the current documents, while at the same time allowing new evidence (newly streamed documents) to change the posterior distribution of topics. Based on the exact definition of the prior the proposed model enables both short-term and long-term dependencies between the previously and currently inferred distributions.

In this paper, we take a special interest in the application of topic models in the area of information retrieval. Our goal is to infer the relevance of each cluster to a user query by calculating the dynamic topic distribution over short documents and incorporating that in a query likelihood model for ad-hoc retrieval. We evaluate our proposed clustering model on a publicly available Twitter dataset by comparing the retrieval performance achieved with state-of-the-art methods and demonstrate the superiority of our algorithm.

The contribution of this work is threefold:

(1) We propose a Dynamic Dirichlet Multinomial Mixture Model that captures short and long term temporal dependencies, tracking dynamic topic distributions over short document streams.

(2) We propose a collapsed Gibbs sampling algorithm for our Dynamic Dirichlet Multinomial Mixture Model to infer the changes in topic and document probability distributions.

(3) We analyze the effectiveness of the proposed clustering models by using the produced clusters to improve the performance of short text ad-hoc retrieval against a Twitter dataset, and demonstrate that our method significantly outperforms state-of-the-art methods.

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

2. RELATED WORK

There are two lines of work related to our work, topic modeling and clustering, with a rich literature available on both topics. In the following sections we only discuss the most related models and algorithms.

2.1 Topic Modeling

Topic modeling provides a suite of algorithms to discover hidden thematic structure in a collection of documents. A topic model takes a collection of documents as input, and discovers a set of "latent topics"—recurring themes that are discussed in the collection—and the degree to which each document exhibits those topics [3]. Latent Dirichlet Allocation (LDA) [3] is one of the simplest topic models, and it decomposes a collection of documents into topics—biased probability distributions over terms—and represents each document with a subset of these topics.

Many models that extend LDA have been proposed, such as topic over time model [20], dynamic mixture model [22], topic tracking model [10], online multi-scale dynamic topic model [11] and more recently, (static) Dirichlet multinomial mixture model [25], Dirichlet-hawkes topic model [6] and user-aware sentiment topic model [24]. These models can either infer topics in static collections of short text, e.g. the (static) Dirichlet multinomial mixture model [25], or infer dynamic topics in long documents, e.g. the dynamic mixture model [22], the topic tracking model [10], and the online multi-scale dynamic topic model [11]. Instead, we propose two dynamic Dirichlet multinomial mixture topic models for short text streams: one for short term dependency of the current inference of the topics and another for long term dependency. Based on these topic models we can infer the dynamic changes of the multinomial distribution of the documents in a stream, and the document probabilities to the topics, and we use the inferred topics to cluster the documents. Hence, our model can both infer topics in short text streams and track the dynamic changes in clusters.

2.2 Clustering

Clustering is one of the main technologies that has been applied to tackle many challenges in data mining, text mining, and information retrieval [5]. For instance, [21] proposed a cluster-based document retrieval model where the clusters are generated by LDA.

Table 1: Main notation used in dynamic clustering topic model.

Notation	Gloss
d	document
z	topic
t	time
v	word
V	total number of words
Z	total number of latent topics
$\mathbf{d}'_t$	set of documents arriving at time t
$\mathbf{d}_t$	document stream up to time t, shorten for $\mathbf{d}_{\le t}$
α_t	parameter of topic Dirichlet prior at time t
β_t	parameter of word Dirichlet prior at time t
$\mathbf{\Theta}_t$	dynamic topic distribution at time t
$\mathbf{\Phi}_t$	dynamic word distribution at time t

[13] presented a burst-aware approach to fusing document lists retrieved in response to a query via integrating information used by fusion methods with that induced from time-sensitive clusters of documents. Efron et al [7] found that relevant documents tend to cluster together in time and utilizing some existing clustering algorithms can boost the performance of tweet search. In terms of data mining, Botezatu et al. [4] proposed a multi-view incident ticket clustering algorithm for optimal ticket dispatching.

To this date, a large number of clustering algorithms have been proposed with KNN (K-Nearest Neighbours) and K-Means as some of the most famous ones. Among those, given that we want to tackle the problem of clustering short documents in streams, we focus on Dirichlet multinomial mixture clustering model [25], that performs well on short text, and which is based on topic modeling. This model acknowledges that as the number of words in short documents is limited, and thus each word in the same document can be assigned to one topic. Then documents assigned to the same topic are in the same cluster. Their experimental results validated the effectiveness of this assumption. However, this model and more recent short document clustering model based on convolutional neural networks [23] can only cluster a static collection of short documents. Other clustering technologies based on topic modeling include dynamic mixture model [22], topic over time model [20] and topic tracking model [10]. However, until now all of the dynamic topic models make a strong assumption that documents arriving in a data stream are long and provide rich context for the inference. To the best of our knowledge, our proposed clustering algorithm is the first attempt to cluster streams of short text documents.

3. TASK DESCRIPTION

The task we address in this work is the clustering of short text streaming documents, with clusters changing dynamically, as new documents stream in. The dynamic clustering algorithm is essentially a function f that satisfies:

$$\mathbf{d}_{\le t} = \{\ldots, \mathbf{d}'_{t-2}, \mathbf{d}'_{t-1}, \mathbf{d}'_t\} \xrightarrow{f} \mathbf{c}_{\le t} = \{\mathbf{c}'_1, \mathbf{c}'_2, \ldots, \mathbf{c}'_Z\},$$

where $\mathbf{d}_{\le t}$ represents the *stream* of documents with $\mathbf{d}'_t$ being the most recent *set* of short documents, arrived at time t, and $\mathbf{c}_{\le t}$ is the resulting set of clusters of documents up to time t, with $\mathbf{c}'_z$ being the z-th cluster in $\mathbf{c}_t$ and Z the total number of clusters. $\mathbf{d}'_t$ comprises a set of short text documents, with each document d being represented by a sequence of words appearing in d, coming from a vocabulary $\mathbf{V} = \{v_1, v_2, \ldots, v_V\}$. We assume that the length of d is no more than a predefined small length (for instance, 140 characters in the case of Twitter). For brievity, in the remainder of the paper, we denote $\mathbf{d}_{\le t}$ and $\mathbf{c}_{\le t}$ with $\mathbf{d}_t$ and $\mathbf{c}_t$, respectively.

Table 1 summarizes the main notation used in our dynamic clustering topic model.

4. DYNAMIC CLUSTERING MODEL

In this section, we describe our proposed dynamic clustering topic model, **DCT**, aiming at the effective clustering of short document streams.

4.1 Preliminaries

The goal of the dynamic clustering topic model is to infer the dynamically changing topic distribution and document distribution over topics at any given time t. That is, we want to infer the temporal word probability for a topic, $P(v|t,z)$, and the temporal topic probability over a document, $P(z|t,d)$. Previous work [25] has demonstrated that algorithms that assign a single topic to all the words in a short document outperform those that assign different topics to different words in terms of clustering quality. The intuition behind this observation is that the number of words in short documents is limited and a short document is likely to be associated with one topic. Following [25], our proposed Gibbs sampling – as it will be described in the following sections – assigns a single topic to all the words in each short document.

Following the notation of past topic modeling work [2, 3, 10, 20], we let $\boldsymbol{\Theta}_t = \{\theta_{t,z}\}_{z=1}^{Z}$be the topic distribution at time t with $\theta_{t,z} = P(z|t) > 0$, and $\sum_{z=1}^{Z} \theta_{t,z} = 1$. We also let $\boldsymbol{\Phi}_t = \{\phi_{t,z}\}_{z=1}^{Z}$be the word distribution over topics at time t. $\phi_{t,z} = \{\phi_{t,z,v}\}_{v=1}^{V}$ is the (multinomial) distribution of words for topic z at time t, while the probability of a word v belonging to z at t, $\phi_{t,z,v} = P(v|t,z) > 0$, and $\sum_{v=1}^{V} \phi_{t,z,v} = 1$; V is the size of the vocabulary $\mathbf{V}$. In fully bayesian non-dynamic topic models (such as LDA [3]), there is an underlying assumption that the per-document topic distribution is independent of the past distributions, and have a Dirichlet prior with a static set of parameters $\kappa = \{\kappa_z\}_{z=1}^{Z}$, with $\kappa_z > 0$,

$$P(\boldsymbol{\Theta}_t|\kappa) \propto \prod_{z=1}^{Z} \theta_{t,z}^{\kappa_z - 1}, \tag{1}$$

Similarly, the per-topic word distribution $\phi_{t,z}$ also has a Dirichlet prior with a static set of parameters $\gamma = \{\gamma_v\}_{v=1}^{V}$, with $\gamma_v > 0$,

$$P(\phi_{t,z}|\gamma) \propto \prod_{v=1}^{V} \phi_{t,z,v}^{\gamma_v - 1}, \tag{2}$$

The assumptions made in (1) and (2) are not realistic when it comes to a data stream setting, where the distributions at time t are dependent on past distributions. In the following subsections, we infer $\boldsymbol{\Theta}_t$ and $\boldsymbol{\Phi}_t$ by modeling short-term dependency (Section 4.2) and long-term dependency (Section 4.3).

4.2 Short-term Dependency DCT

Modeling short-term dependency. To model the temporal dependencies of the topics in a document stream, and by following the work of past dynamic topic models [10, 11, 22], we propose a *short-term-dependency* DCT model. According to this model, the topic distribution at time t remains the same as the one at time $t-1$ if no new documents are observed, while it is updated on the basis of new evidence when a new set of documents is observed at time t. To achieve that we factorize the parameter κ in (1) into the mean of the distribution at the previous time-step, $\theta_{t-1,z}$, and a set of precision values $\alpha_t = \{\alpha_{t,z}\}_{z=1}^{Z}$. Hence, $\kappa = \alpha_t \boldsymbol{\Theta}_{t-1}$, which allows the mean of the current distribution $\boldsymbol{\Theta}_t$ to depend on the mean of the previous distribution $\boldsymbol{\Theta}_{t-1}$,

$$P(\boldsymbol{\Theta}_t|\boldsymbol{\Theta}_{t-1}, \alpha_t) \propto \prod_{z=1}^{Z} \theta_{t,z}^{(\alpha_{t,z}\theta_{t-1,z})-1}, \tag{3}$$

where the precision value $\alpha_{t,z}$ represents the topic persistency, that is how salience is topic z at time t compared to that at time $t-1$. The distribution is a conjugate prior of the Multinomial distribution, hence the inference can be performed by Gibbs sampling [17].

In a similar way, to model the dynamic changes of the multinomial distribution of words specific to topic z, we assume a Dirichlet prior, in which the mean of the current distribution $\boldsymbol{\Phi}_t$ evolves from the mean of the previous distribution $\boldsymbol{\Phi}_{t-1}$ with the precision being β_t,

$$P(\phi_{t,z}|\phi_{t-1,z}, \beta_{t,z}) \propto \prod_{v=1}^{V} \phi_{t,z,v}^{(\beta_{t,z,v}\phi_{t-1,z,v})-1}, \tag{4}$$

where as before, the Dirichlet prior parameter γ in (2) is factorized into the mean and precision, $\gamma = \beta_{t,z}\phi_{t-1,z}$, with $\beta_t = \{\beta_{t,z}\}_{z=1}^{Z}$ being the set of precision values at time t for the topics. Here $\beta_{t,z} = \{\beta_{t,z,v}\}_{v=1}^{V}$, with $\beta_{t,z,v}$ representing the persistency of word v in topic z at time t, a measure of how consistently word v belongs to topic z at time t compared to that at the previous time $t-1$. We describe the inference for $\boldsymbol{\Theta}_t$, $\boldsymbol{\Phi}_t$, α_t and β_t in later part of this section.

Assuming that we know the topic distribution at time $t-1$, $\boldsymbol{\Theta}_{t-1}$, and the word distribution over topics at time $t-1$, $\boldsymbol{\Phi}_{t-1}$, the proposed Dynamic Dirichlet Multinomial Mixture Model is a generative topic model that depends on $\boldsymbol{\Theta}_{t-1}$ and $\boldsymbol{\Phi}_{t-1}$. We can initialize (at time $t = 0$) the means of the two distributions to $\theta_{0,z} = 1/Z$ and $\phi_{0,z,v} = 1/V$. The generative process (used by the Gibbs sampler for parameter estimation) of our model for documents in stream $\mathbf{d}_t$ at time t, is as follows,

i. Draw a multinomial distribution $\boldsymbol{\Theta}_t$ from a Dirichlet prior distribution $\alpha_t \boldsymbol{\Theta}_{t-1}$;

ii. Draw Z, one for each topic z, multinomial distributions $\phi_{t,z}$ from a Dirichlet prior distribution $\beta_{t,z}\phi_{t-1,z}$;

iii. For each document $d \in \mathbf{d}_t$, draw a topic z_d from the multinomial distribution $\boldsymbol{\Theta}_t$ and for each word v_d in the document d:

    (a) Draw a word v_d from multinomial ϕ_{t,z_d};

Fig. 1 illustrates the graphical representation of our Dynamic Dirichlet Multinomial Mixture Model; given that documents are short, and following [25], all words in the same document d are drawn from the Multinomial distribution associated with the same topic z_d. The parameterization of the proposed dynamic topic model is as follows:

$$\begin{aligned} \boldsymbol{\Theta}_t &\sim \text{Dirichlet}(\alpha_t \boldsymbol{\Theta}_{t-1}) \\ \phi_{t,z}|\beta_{t,z}\phi_{t-1,z} &\sim \text{Dirichlet}(\beta_{t,z}\phi_{t-1,z}) \\ z_d &\sim \text{Multinomial}(\boldsymbol{\Theta}_t) \\ v_d|\phi_{t,z_d} &\sim \text{Multinomial}(\phi_{t,z_d}) \end{aligned}$$

Note that in the generative process described above, there is a fixed number of latent topics Z. A non-parametric Bayes version of our dynamic topic model that automatically integrates over the number of topics is possible, but we leave this as future work.

Inference for the short-term dependency DCT. The inference of the distribution parameters of the model is intractable. Following [14, 15, 19, 20] we employ a collapsed Gibbs sampler [9] for an approximate inference. We adopt a conjugate prior (Dirichlet) for the multinomial distributions, and thus we can easily integrate out the uncertainty associated with $\phi_{t,z}$ and $\boldsymbol{\Theta}_t$. In this way we enable sampling since we do not need to sample $\phi_{t,z}$ or $\boldsymbol{\Theta}_t$.

In the Gibbs sampling procedure we need to calculate the conditional distribution $P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t)$, at time t,

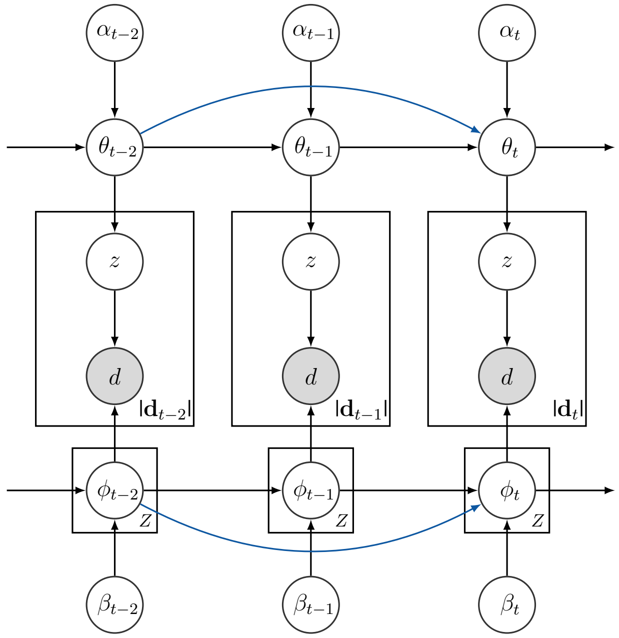


Figure 1: Graphical representation of our dynamic Dirichlet multinomial mixture clustering topic model, DCT. Note that short term dependence DCT model excludes the two blue curved lines; while long term dependence DCT model does include these two lines. The figure is best viewed in color.

where $\mathbf{z}_{t,-d}$ represents the topic assignments for all documents in $\mathbf{d}_t$ except document d. We begin with the joint probability of the current document set $\mathbf{d}_t$, $P(\mathbf{d}_t, \mathbf{z}_t | \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$ (see Appendix A for the detail of the join probability), and using the chain rule, we can obtain the following conditional probability,

$$P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t) \propto \frac{m_{t,z} + \alpha_{t,z}\theta_{t-1,z} - 1}{\sum_{z=1}^{Z}(m_{t,z} + \alpha_{t,z}\theta_{t-1,z}) - 1} \times \frac{\prod_{v \in d} \prod_{j=1}^{N_{d,v}} (n_{t,z,v,-d} + \beta_{t,z,v}\phi_{t-1,z,v} + j - 1)}{\prod_{i=1}^{N_d} (n_{t,z,-d} + i - 1 + \sum_{v=1}^{V} \beta_{t,z,v}\phi_{t-1,z,v})}, \tag{5}$$

where $m_{t,z}$ is the total number of documents in $\mathbf{d}_t$ assigned to topic z, $N_{d,v}$ is the number of word v in the document d, $n_{t,z,v,-d}$ is the total number of the word v assigned to topic z except that in d, and $n_{t,z,-d}$ is the total number of documents assigned to z except d. Detailed derivation of Gibbs sampling for our proposed DCT model is provided in Appendix A. During sampling, at each iteration, the precision parameters α_t and β_t can be estimated by maximizing the joint distribution $P(\mathbf{d}_t, \mathbf{z}_t | \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$. We apply fixed-point iteration to get the optimal α_t and β_t at time t. The following update rule of α_t for maximizing the joint distribution in our fixed-point iteration is derived by applying two bounds in [18],

$$\alpha_{t,z} \leftarrow \frac{\alpha_{t,z}\left(\Psi(m_{t,z} + \alpha_{t,z}\theta_{t-1,z}) - \Psi(\alpha_{t,z}\theta_{t-1,z})\right)}{\Psi(\sum_{z=1}^{Z} m_{t,z} + \alpha_{t,z}\theta_{t-1,z}) - \Psi(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})},$$

where $\Psi(\cdot)$ defined by $\Psi(x) = \frac{\partial \log \Gamma(x)}{\partial x}$ is the digamma function; whereas the following update rule of β_t is,

$$\beta_{t,z,v} \leftarrow \frac{\sum_{z=1}^{Z} \beta_{t,z,v}\phi_{t-1,z,v}A_{t,z,v}}{\sum_{z=1}^{Z} \phi_{t-1,z,v}B_{t,z,v}},$$

where $A_{t,z,v} = \Psi(n_{t,z,v} + \beta_{t,z,v}\phi_{t-1,z,v}) - \Psi(\beta_{t,z,v}\phi_{t-1,z,v})$, $B_{t,z,v} = \Psi(\sum_{v=1}^{V} n_{t,z,v} + \beta_{t,z,v}\phi_{t-1,z,v}) - \Psi(\sum_{v=1}^{V} \beta_{t,z,v}\phi_{t-1,z,v})$ and $n_{t,z,v}$ is the number of word v assigned to topic z in stream $\mathbf{d}_t$.

Algorithm 1: Inference for the Dynamic Dirichlet Multinomial Mixture Model at time t.

```
Input  : Previous topic distribution $\mathbf{\Theta}_{t-1}$
         Previous word distribution specific to topics $\mathbf{\Phi}_{t-1}$
         A set of short documents $\mathbf{d}_t$ at time $t$
         Initialized $\alpha_t$ and $\beta_t$
         Number of iterations $N_{iter}$
Output: Current topic distribution $\mathbf{\Theta}_t$
         Current word distribution specific to topics $\mathbf{\Phi}_t$
         Documents' probabilities to each topic at time $t$,
           $P(z|t,d)$
1 Initialize topic assignments randomly for all documents in $\mathbf{d}_t$
2 for iteration = 1 to $N_{iter}$ do
3     for $d$ = 1 to $|\mathbf{d}_t|$ do
4         draw $z_d$ from $P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \mathbf{\Phi}_{t-1}, \mathbf{\Theta}_{t-1}, \alpha_t, \beta_t)$
5         update $m_{t,z_d}$ and $n_{t,z_d,v}$
6     update $\alpha_t$ and $\beta_t$
7 Compute the posterior estimates $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$
8 Compute $P(z|t,d)$
```

Our derivation of the update rules for α_t and β_t, and the two bounds used in deviating the update rules are detailed in Appendix B. An overview of our proposed collapsed Gibbs sampling algorithm, including the input and output, is shown in Algorithm 1.

4.3 Long-term Dependency DCT

Modeling long-term dependency. So far the distributions $\mathbf{\Theta}_t$ and $\mathbf{\Phi}_t$ depend on the previous time-step distributions. Research has shown that topic distributions - or the interests of a user on a topic when searching for information - may depend on a longer time-step history. We model such a *long-term (L-steps) dependency* DCT model on the basis of the distribution priors as follows:

$$P(\mathbf{\Theta}_t|\{\mathbf{\Theta}_{t-l}, \alpha_{t,l}\}_{l=1}^{L}) \propto \prod_{z=1}^{Z} \theta_{t,z}^{\left(\sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z}\right)-1}. \tag{6}$$

The mean in this case is proportional to the weighted sum of the past L "topic trends" in the documents, and $\alpha_{t,l} = \{\alpha_{t,z,l}\}_{z=1}^{Z}$ represents how the topics at time t are related to the l-previous topics. For a comparison with the short-term dependency model refer to Eq. (3) and Eq. (6). Further, long-term dependency reduces the information loss and the bias of the inference due to the multiple estimates.

Similarly, the Dirichlet prior of the topic trends $\phi_{t,z}$ at t can be modified 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. Thus, we have:

$$P(\phi_{t,z}|\{\phi_{t-l,z}, \beta_{t,z,l}\}_{l=1}^{L}) \propto \prod_{v=1}^{V} \phi_{t,z,v}^{\left(\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v}\right)-1}, \tag{7}$$

where $\beta_{t,z,l} = \{\beta_{t,z,v,l}\}_{v=1}^{V}$ represents how the word distribution over topics at time t are related to the l-previous one.

Inference for the long-term dependency DCT. The parameters $\mathbf{\Theta}_t$ and $\phi_{t,z}$ in Eq. (6) and Eq. (7) can be integrated in the exact same way as before (since priors are still Dirichlet distributed) and $\mathbf{\Theta}_t$ and $\phi_{t,z}$ at time t are inferred using the proposed Gibbs sampling in Algorithm 1. The only difference lies in the way we sample the latent topic for each document (step 4 in Algorithm 1) and the update rules for the priors (step 6 in Algorithm 1). Similar to Eq.(5), we sample a latent topic for a document d by:

$$P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \{\mathbf{\Phi}_{t-l}, \mathbf{\Theta}_{t-l}, \alpha_{t,l}, \beta_{t,l}\}_{l=1}^{L}) \propto \frac{m_{t,z} + \sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z} - 1}{\sum_{z=1}^{Z}(m_{t,z} + \sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z}) - 1} \times \frac{\prod_{v\in d}\prod_{j=1}^{N_{d,v}}(n_{t,z,v,-d} + \sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v} + j - 1)}{\prod_{i=1}^{N_d}(n_{t,z,-d} + i - 1 + \sum_{v=1}^{V}\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v})}. \quad (8)$$

The derivation of Eq. (8) is similar to that of Eq. (5) (see Appendix A). Again, we update $\alpha_{t,z,l}$ in Eq. (8) using the two bounds in [18] with fixed-point iteration such that:

$$\alpha_{t,z,l} \leftarrow \frac{\alpha_{t,z,l}C_{t,z}}{D_{t,z} - D'_{t,z}},$$

where $C_{t,z} = \Psi(m_{t,z}+\sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z})-\Psi(\sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z})$, and $D_{t,z} = \Psi(\sum_{z=1}^{Z} m_{t,z} + \sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z})$ and $D'_{t,z} = \Psi(\sum_{z=1}^{Z}\sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z})$. Similarly, we update $\beta_{t,z,v,l}$ in (8) with fixed-point iteration by:

$$\beta_{t,z,v,l} \leftarrow \frac{\sum_{z=1}^{Z} \beta_{t,z,v,l}\phi_{t-l,z,v}A'_{t,z,v}}{\sum_{z=1}^{Z} \phi_{t-l,z,v}B'_{t,z,v}},$$

where $A'_{t,z,v} = \Psi(n_{t,z,v}+\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v})-\Psi(\sum_{l=1}^{L} \beta_{t,z,v,l}$ $\phi_{t-l,z,v})$, and $B'_{t,z,v} = \Psi(\sum_{v=1}^{V} n_{t,z,v}+\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v})-$ $\Psi(\sum_{v=1}^{V}\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v})$. Given the space limitation, we do not show the derivations of the update rules for $\alpha_{t,z,l}$ and $\beta_{t,z,v,l}$, as they are similar to those for $\alpha_{t,z}$ and $\beta_{t,z,v}$ in our short term dependency DCT model (see Appendix B).

4.4 Clustering

Now, we can infer the dynamic topic distribution at time t, $\mathbf{\Theta}_t$ in our short-term dependency DCT model as,

$$\theta_{t,z} = \frac{m_{t,z} + \alpha_{t,z}\theta_{t-1,z}}{\sum_{z=1}^{Z} m_{t,z} + \alpha_{t,z}\theta_{t-1,z}} = \frac{m_{t,z} + \alpha_{t,z}\theta_{t-1,z}}{m_t + \sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z}},$$

where m_t is the total number of documents in $\mathbf{d}_t$, and infer a multinomial distribution over words for topic z at time t as,

$$\phi_{t,z,v} = \frac{n_{t,z,v} + \beta_{t,z,v}\phi_{t-1,z,v}}{n_{t,z} + \sum_{v=1}^{V} \beta_{t,z,v}\phi_{t-1,z,v}}, \quad (9)$$

where $n_{t,z}$ is the number of words assigned to topic z at time t. Similarly, we can infer the dynamic topic distribution at time t in our long-term dependency DCT model as,

$$\theta_{t,z} = \frac{m_{t,z} + \sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z}}{m_t + \sum_{z=1}^{Z}\sum_{l=1}^{L} \alpha_{t,z,l}\theta_{t-l,z}},$$

and the multinomial distribution over words for topic z at time t,

$$\phi_{t,z,v} = \frac{n_{t,z,v} + \sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v}}{n_{t,z} + \sum_{v=1}^{V}\sum_{l=1}^{L} \beta_{t,z,v,l}\phi_{t-l,z,v}}. \quad (10)$$

As one can observe in all equations for the two models, the short-term model is just a special case of the long-term one for $L = 1$.

Having computed $\theta_{t,z}$ and $\phi_{t,z,v}$, we can compute the probability that a document d is relevant to topic z_d at time t in the stream $\mathbf{d}_t$, $P(z_d|t, d)$ as:

$$P(z_d|t,d) = \frac{P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \{\mathbf{\Phi}_{t-l}, \mathbf{\Theta}_{t-l}, \alpha_{t,l}, \beta_{t,l}\}_{l=1}^{L})}{\sum_{z'_d=1}^{Z} P(z'_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \{\mathbf{\Phi}_{t-l}, \mathbf{\Theta}_{t-l}, \alpha_{t,l}, \beta_{t,l}\}_{l=1}^{L})}, \quad (11)$$

where $P(z_d|\mathbf{z}_{t,-d}, \mathbf{d}_t, \{\mathbf{\Phi}_{t-l}, \mathbf{\Theta}_{t-l}, \alpha_{t,l}, \beta_{t,l}\}_{l=1}^{L})$ can be obtained by Eq. (5) and Eq. (8) for the short- and long-term dependency DCT model, respectively. Finally, the document d in stream $\mathbf{d}_t$ at t is clustered to cluster $\mathbf{c}'_z$, i.e., the topic $z = \arg\max_{z_d} P(z_d|t, d)$.

5. EXPERIMENTAL SETUP

Ideally, we would like to evaluate the performance of our dynamic clustering model by directly comparing the clustering result with ground truth labels in a streaming short text corpus. However, to the best of our knowledge, there is no such collection available to this date; obtaining cluster labels for all documents in a stream and all points in time is rather expensive. Instead, we perform an extrinsic evaluation of the proposed model: (a) we incorporate the clustering algorithm derived by the DCT model into a cluster-based query likelihood model for ad-hoc retrieval [5, 21], and test the clustering quality on the basis of retrieval performance, and (b) we test the ability of the DCT generative model to predict the observed data on the basis of perplexity [2, 3]. We compare the performance of our model with other state-of-the-art clustering models.

The cluster-based ad-hoc retrieval model [21] used in our experimental setup is the following:

$$P(q|t,d) = \prod_{v\in q} P(v|t,d)^{n(v,q)}, \quad (12)$$

where $n(v, q)$ is the term frequency of term v in query q, and $P(v|t, d)$ is the probability of document $d \in \mathbf{d}_t$ being relevant to the query term v, which is computed by using a Dirichlet smoothing language model [5] as,

$$P(v|t,d) = \lambda P_{\text{Cluster}}(v|t,d) + (1-\lambda)\Big(\frac{N_d}{N_d+\mu}P_{\text{ML}}(v|t,d) + \big(1 - \frac{N_d}{N_d+\mu}\big)P_{\text{ML}}(v|\mathbf{d}_t)\Big) \quad (13)$$

where λ is a free parameter, μ is a Dirichlet prior in language model [5], and $P_{\text{ML}}(v|t, d)$, $P_{\text{ML}}(v|\mathbf{d}_t)$ and $P_{\text{Cluster}}(v|t, d)$ are the maximum likelihood estimates of word v in the document d, in the current short document stream $\mathbf{d}_t$ and in the document d in terms of clusters at time t, respectively. According to the cluster-based retrieval model proposed in [21], $P_{\text{Cluster}}(w|t, d)$ is computed by,

$$P_{\text{Cluster}}(v|t,d) = \sum_{z=1}^{Z} P(v|t,d,z)P(z|t,d),$$

where $P(v|t, d, z)$ is the probability of word v being relevant to topic z at time t, and $P(z|t, d)$ the probability of document d being assigned to topic z. When applying the proposed DCT clustering model, for instance, we set $P(v|t, d, z) = \phi_{t,z,v}$, where $\phi_{t,z,v}$ is defined in Eq. (9) and Eq. (10) for the short and long term dependence DCT models, respectively, while $P(z|t, d)$ is defined in Eq. (11), for the two models, respectively.

5.1 Research Questions

The research questions we investigate in experimental section of the paper are:

On the basis of ranking performance:

RQ1: How does the proposed DCT clustering model perform compared to state-of-the-art clustering algorithms in searching a short-text document stream?

RQ2: Is the performance consistent across different user queries?

RQ3: How does the performance of the long-term dependence DCT model compares to that of the short-term dependence DCT model?

RQ4: What is the impact of the free parameter λ in Eq. (13) when applying the DCT model on cluster-based ad-hoc retrieval?

RQ5: Is the performance of cluster-based ad-hoc retrieval sensitive to the number of clusters used in the DCT model?

On the basis of the generative model:

RQ6: What is the performance of the generative DCT model compared to other baseline topic models in terms of the likelihood of generating the top-k documents (measured by perplexity [3])?

5.2 Data Set

One of the key criteria for a suitable test collection for our ad-hoc retrieval task is the dynamic nature of the intent of a users' query. That is we make the assumption that for the same query, e.g. *Egypt*, the intent may change over time (something that we hope to be reflected in the identified dynamic topic distribution). Publicly available labeled corpora, such as Tweets2011 and Tweets2013 used for ad hoc retrieval in TREC 2011–2015 Microblog track [16], have been constructed however by judging documents against a static query intent; furthermore the time-span of the collection is relatively small (16 and 59 days, respectively).

To allow for a dynamic query intent we construct a new test collection based on a publicly available corpus of Twitter posts (an 1% sample of all tweets). [1] The corpus has been collected between February 1, 2015 and April 30, 2015, covering a period of 90 days. Most of the tweets are written in English; we remove non-English tweets and retweets to end up with 369 million tweets. We follow Fisher et al. [8] and generated queries and relevance labels as follows: (a) Manual selected hashtags on topics of general interest, such as "#Apple" and "#Egypt" are transformed into keyword queries. (b) Given a query at time t, we label the top-k documents retrieved by a time-sensitive language model (see Section 5), resulting in the query-document ground truth used in our experiments. Assessors are university students employed remotely, while no specific intent for a query was provided to them. Therefore, it was up to their own judgment to decide what constitutes relevant and what not. To enable the possibility of query intent drifting relevance judgments were not obtained retrospectively (i.e. at the end of the 90 days period) but we simulated a streaming scenario and obtained labels at 20-day intervals[2]. This resulted in 5 sets of ground truth data: on February 9th, March 1st, March 21st, April 10th, and April 30th of 2015. Our test collection includes 107 queries, and 5 sets of (disjoint) ground truth labels for each one of them.

5.3 Baselines, Evaluation Metrics, and Setting

We compare the DCT [3] model with a number of baselines and state-of-the-art algorithms:

Language Model (LM) [5]: Directly ranks documents by their relevance scores computed by a multinomial query likelihood model - Eq. (12) and (13) after removing $P_{\text{Cluster}}(v|t,d)$.

Time-aware Microblog Search (TMS) [7]: Based on the temporal cluster hypothesis that relevant documents tend to cluster together in time,first adopts a feedback framework where temporal features are extracted from an initial ranked list of documents and then reranks this list to produce a final ranking.

Laten Dirichlet Allocation (LDA) [21]: Clusters documents based on LDA and ranks them by a cluster-based document retrieval model (Eq. (12) and (13)), in the same way DCT ranks documents.

Dirichlet Multinomial Mixture Model (DMM) [25]: Clusters documents based on a vanilla Dirichlet multinomial mixture model (without the temporal dependencies introduced by this paper) and ranks them by Eq. (12) and (13).

Topic Tracking Model (TTM) [10]: Clusters documents based on a dynamic topic tracking model that captures temporal dependencies between long text streams, and ranks them by Eq. (12) and (13).

LDA, DMM, TTM and our proposed DCT use the same retrieval model, i.e., Eq.(13), to compute the relevant scores for the documents; they only differ in the way they perform clustering. For all methods including the vanilla LM, we define the probability of a term given a document and a point in time as $P_{\text{ML}}(v|t,d) = P_{\text{ML}}(v|d) \cdot b^{-(t-t_d)}$, where b is a base parameter that determines the rate of the recency decay and t_d is the creation time of document d. In the remainder of this paper we refer to the cluster-based retrieval models with the name of the clustering method they employ, that is, LDA, DMM, TTM and DCT.

The evaluation metrics used to assess the performance of the ranking algorithms are the ones widely used in TREC 2011—2015 Microblog tracks [16]: NDCG [12], MAP [5], Recall, R-prec, and P@k (Precision at k) [5]. R-Prec is the precision after R documents have been retrieved, where R is the total number of relevant document for the query. We set k to 30 to align with the cut-off used in the TREC Microblog tracks [16]. The statistical significance of the observed differences between the performance of two ranking algorithms across the 107 queries is tested using a two-tailed paired t-test and is denoted using $^{\blacktriangle}$ (or $^{\blacktriangledown}$) for $\alpha = .01$, and $^{\vartriangle}$ (and $^{\triangledown}$) for $\alpha = .05$.

Experiments are run as follows: First, we obtain the top-k documents, $\mathbf{d}_t$, in response to a user's query using a vanilla query likelihood model at time t. In our experiments we used $k = 500$, but we experimented with other values for k; for any $k > 100$ the results of our experiments remained stable. For cluster-based retrieval, topics are then inferred over the documents in $\mathbf{d}_t$, and (a) documents are re-ranked based on the cluster-based query likelihood model, and rankings are evaluated on the basis of different information retrieval metrics, and (b) we calculate the likelihood of observing these documents in the collection on the basis of the underlying generative model. We use a 60/30/10 split of our collection for training, validation and testing, respectively. We train the vanilla LM, LDA, DMM, TTM, and DCT for different values of the parameters λ, and μ in Eq. (13); λ varies from 0 to 1.0 and μ from 0 to 1000. The optimal λ and μ values are decided based on the validation set, and evaluated on the test set. The training/validation/test splits are permuted until all 107 queries have been chosen once for the test set. We repeat the experiments 10 times and report the average evaluation measures.

6. RESULTS

We start by comparing the retrieval performance of DCT with the rest of the methods in Section 5.3 **(RQ1)**, and the persistence of the performance across queries **(RQ2)**. We then analyse the effect of

[1] https://archive.org/details/twitterstream

[2] Applying shorter intervals requires more efforts of manual labeling and we found that it yielded not significantly different results in many cases.

[3] The code of the DCT topic model is available at https://bitbucket.org/sliang1/dct/get/DCT.zip

Table 2: Mean performance over the five test cutoff days. The best performance per metric is in boldface. Statistically significant differences between DCT and the best baseline, TTM, are marked in the upper right-hand corner of DCT's performance scores.

	NDCG	MAP	Recall	R-prec	P@30
LM	.4446	.2241	.3092	.3534	.2772
LDA	.5018	.2749	.3439	.3976	.3171
TMS	.5286	.2991	.3604	.4152	.3353
DMM	.5715	.3416	.3854	.4460	.3685
TTM	.5936	.3638	.4019	.4648	.3928
DCT	**.6421**$^{\blacktriangle}$	**.4132**$^{\blacktriangle}$	**.4298**$^{\blacktriangle}$	**.5021**$^{\blacktriangle}$	**.4277**$^{\blacktriangle}$

various parameters in our model: the dependency length **(RQ3)**, the mixture parameter λ **(RQ4)**, and the predefined number of topics **(RQ5)**. Last, we test the generalisability of the proposed generative model in terms of perplexity **(RQ6)**.

6.1 The Ranking Performance of DCT

RQ1: We compare the ranking performance of the short term DCT cluster-based retrieval model with the rest of the methods in Section 5.3.

Table 2 reports the performance averaged across all five testing time cutoffs. The ranking of models with respect to the retrieval performance is consistent across the different evaluation measures, and in particular the following order is observed: DCT $>$ TTM $>$ DMM $>$ TMS $\geq$ LDA $>$ LM. Here $>$ denotes statistically significantly better performance at a significance level of 99%, and $\geq$ denotes statistically significantly better performance at a significance level of 95%. To get a better insight on the persistence of the results across the five testing time cutoffs, we compare the performance of the six algorithms on a per cutoff basis. We visualise the results in Fig. 2 in terms of five heat maps, one per metric, so that the relative performance per model and per time cutoff can be observed, by examining the intense of the color (dark blue translates to high measure values, and light blue to low measure values). The five heat maps lead to the exact same findings per time cutoff to those when the average values were considered: in most cases, DCT statistically significantly outperforms TTM, which is followed by DMM, TMS, LDA, and LM.

The finding DCT $>$ TTM in both Table 2 and Fig. 2 illustrates that the way we track the changes of topics specific to a query in DCT works better than the way it is done by TTM which focuses on long documents. The finding DCT $>$ DMM illustrates that DCT integrates time information better in the inference of topics distribution at time t compared to DMM, which ignores time information. An interesting observation in Fig. 2 is that as time progresses, the performance of both DCT and all other baselines slightly decreases due to the fact that more and new intents underlying a given query appear and make the retrieval task more challenging. Instead the performance of DCT remains stable across all the test time cutoffs.

6.2 Query-level Analysis

RQ2: We take an in-depth view of the improvements of DCT performance over the best baseline (TTM) on a per query basis.

Fig. 3 shows the per query performance differences in terms of all the metrics, averaged across all the test days. The number of queries on which DCT outperforms TTM is larger than the number of queries on which TTM outperforms DCT, for every metric. Further, the positive differences of DCT against TTM are larger than the negative differences in most case. Both of these findings further support the conclusion that DCT can effectively capture the topic distribution at a given time and query for clustering short documents in streams compared to state-of-the-art dynamic or non-dynamic clustering topic models for long or short text documents. There are only very few cases in which DCT performs worse than TTM.

6.3 Impact of Dependency Length

RQ3: We compare the short-term DCT with the long-term DCT (DCT-L with L being the length of dependency under consideration). We vary the length of dependency from 1 to 8 time-steps.

Fig. 4 shows the performance on the metrics, averaged across the five test cutoff days. It is clear from the figure that the longer the dependences captured by the model the better the performance of the ranking algorithm. This is especially true for $L = 1, \ldots, 4$, while after that the performance reaches a plato. This illustrates that our DCT model can enhance the performance of clustering when past distribution information is integrated in the model.

In the remaining of the analysis we will focus on the short term DCT model so that we can study the performance of our dynamic topic model independently of the length of the dependency. The performance of DCT-L is at least as good as the performance of the short-term DCT.

6.4 Contribution of Clustering Ingredient

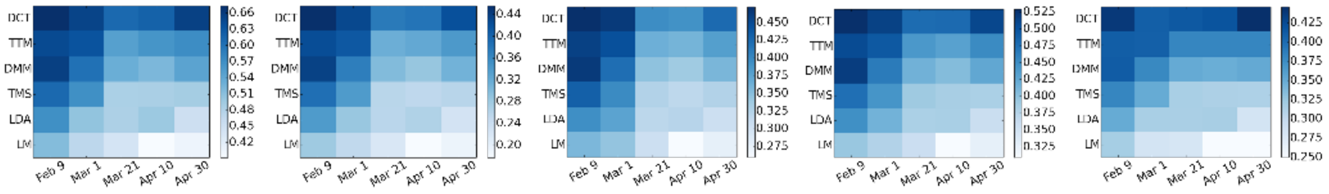
RQ4: We vary λ in Eq. (13) and measure the average performance of our model to analyze the contribution of clustering ingredient in the cluster-based retrieval model.

Fig. 5 depicts the performance on all metrics. For $\lambda = 0$, the performance of DCT and the rest of the methods is identical with the time-sensitive language model (LM) performance, as expected. As λ increases from 0 to 0.6, giving more weight to the cluster terms, the performance of all cluster-based methods improves, with the DCT clusters providing more relevant to the query terms. This leads to a faster improvement of DCT compared to the rest of the methods (TTM, DMM, and LDA), which demonstrates the homogeneity of clusters on the query topic. The performance of all algorithms drops as expected when larger weights are given to the cluster terms. However, even when the query is completely ignored ($\lambda = 1$) the DCT clusters continue to provide good on-topic terms outperforming the language model method in the task of reranking. Again, these findings strengthen our conclusion that integrating high quality clustering information, as provided by our dynamic clustering model, can enhance the performance of ad-hoc retrieval in short document streams.

6.5 Effect of the Number of Topics

RQ5: We examine the effect of the number of latent topics passed as an input parameter to DCT and the rest of the clustering models on the overall retrieval performance. We vary the number of latent topics from 2 to 16 for each query, and compare the performance in terms of all the metrics, averaged across all five test cutoff days.

As illustrated by Fig. 6 when only two latent topics are modeled, the four clustering models yields almost the same performance; if the number of available topics to be inferred is small DCT does not offer any improvements compared to other methods. With the number of latent topics increasing to 4 and 8, the positive performance differences between DCT and baseline methods also increases. When the number of latent topics further increases (e.g. between 8 to 16), the performance of all the clustering models reaches a plato. This also demonstrates the merit of the proposed DCT model: it is robust and insensitive to the number of latent topics and once enough latent topics are used it is able to improve the



(a) NDCG (b) MAP (c) Recall (d) R-prec (e) P@30

Figure 2: Heat maps of retrieval performance (one per metric); columns represent testing cutoff days (February 9, March 1, March 21, April 10, and April 30, respectively); rows represent methods (DCT, TTM, DMM, TMS, LDA, and LM, going from top to bottom).

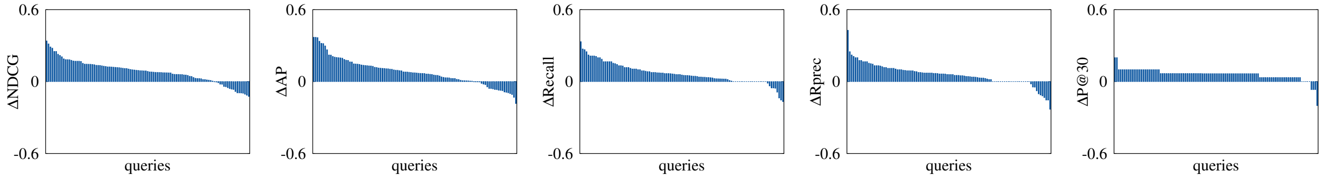


Figure 3: Per-query retrieval performance differences between DCT and TTM, averaged across all test days. A bar above the line y=0 indicates that DCT outperforms TTM, while the opposite is true for bars below y=0.

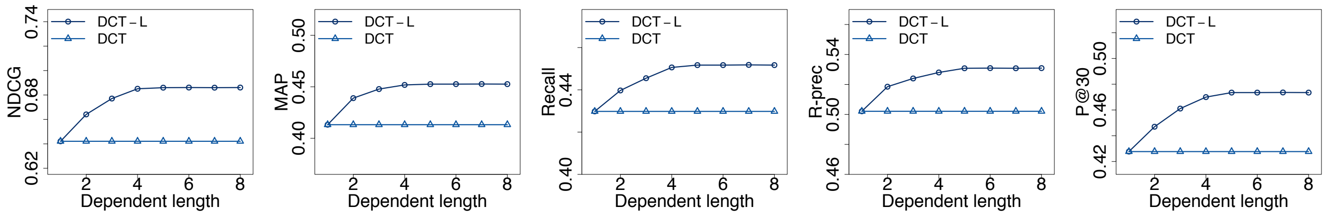


Figure 4: Mean retrieval performance of the short-term and the long-term DCT with the dependency length L varying between 2 and 8.

performance of the cluster-based retrieval model and work better than the state-of-the-art dynamic and non-dynamic clustering models in short document streams.

6.6 Perplexity

RQ6: Last, we evaluate the performance of DCT and the baseline models in terms of perplexity, which is widely used as an evaluation metric in previous topic modeling work [2, 3]. The perplexity used in language modeling, is monotonically decreasing with the likelihood of the documents, and is algebraically equivalent to the inverse of the geometric mean per-word likelihood. The perplexity [3] that is widely used to evaluate the generalization performance of many topic models is computed as $\text{Perplexity}(\mathbf{d}_t) = \exp\Big(-\frac{\sum_{d=1}^{|\mathbf{d}_t|}\sum_{v\in d}\log p(v|t,d)}{\sum_{d=1}^{|\mathbf{d}_t|} N_d}\Big)$, where N_d is the number of words in document d, and $p(v|t,d) = \sum_z p(v|t,d,z)p(z|t,d)$. A lower perplexity score indicates better generalization performance. Fig. 7 shows the mean perplexity performance of DCT and the baseline models, over the five test cutoff days with the number of latent topics ranging between 2 and 16 for each query. As it can be observed, DCT consistently performs better than the rest of the models, with the performance flattening out when the number of topics is equal or more than 8.

7. CONCLUSION

Clustering technologies have been widely used in a number of text related applications including information retrieval, and summarization. In this work we studied the problem of clustering short document streams, and proposed a new dynamic Dirichlet multinomial mixture clustering topic model, DCT, to effectively handle both the textual sparsity of short documents, and the dynamic nature of topics across time. The proposed clustering model can capture short-term and long-term trends in topics. We evaluated the performance of the proposed model in terms of retrieval effectiveness. We conducted experiments over a Twitter streaming dataset, which was manually labeled. To allow possible drifts in the query intent across time we did not provide any static query intent description to the assessors. We compared the performance of the proposed model with a state-of-the-art dynamic topic model that infers clusters in the context of long documents, a static topic model that infers clusters in static short document sets, a state-of-the-art time-aware microblog search model, an LDA topic model, and a time-sensitive language model. Our experimental results demonstrate the effectiveness of the proposed dynamic clustering model.

As future work we intent to automatically estimate the (dynamic) number of topics in our clustering model in the context of short document streams, and use the proposed model to improve the performance of other text-related applications such as tweet summarization, sentiment analysis, and query suggestion in the context of short document streams.

Acknowledgments. We thank Prof. W. Bruce Croft and Stephen M. Harding at the University of Massachusetts, Amherst for helpful discussion and preprocessing the dataset, respectively. This research was partially supported by the UCL Big Data Institute and Elsevier.

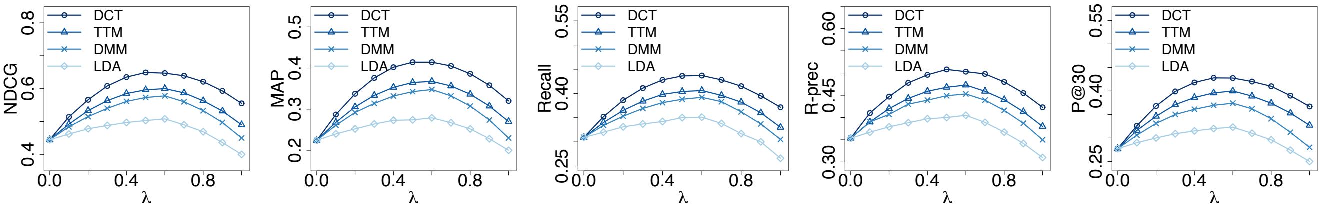


Figure 5: Mean retrieval performance of DCT and the state-of-the-art topic models when varying the smoothing parameter λ.

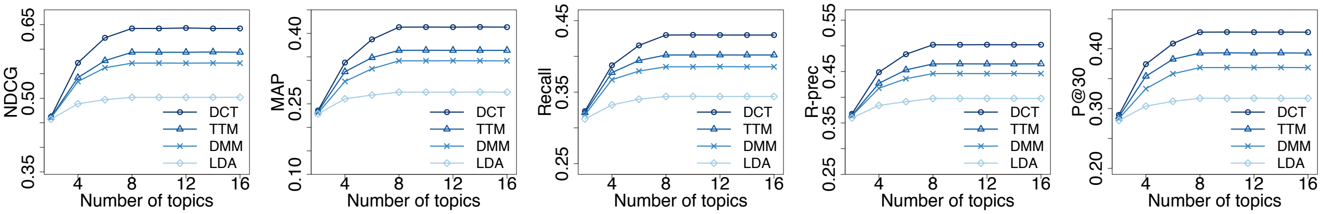


Figure 6: Mean retrieval performance of DCT and the state-of-the-art topic models with the number of latent topics ranging from 2 to 16.

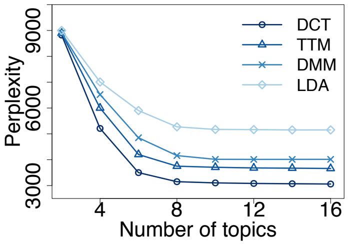


Figure 7: Mean perplexity of DCT and state-of-the-art topic models with varying numbers of latent topics.

8. REFERENCES

[1] N. Begum, L. Ulanova, J. Wang, and E. Keogh. Accelerating dynamic time warping clustering with a novel admissible pruning strategy. In *KDD*, pages 49–58, 2013.

[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. Mach. Learn. Res.*, 3:993–1022, 2003.

[4] M. Botezatu, J. Bogojeska, I. Giurgiu, H. Voelzer, and D. Wiesmann. Multi-view incident ticket clustering for optimal ticket dispatching. In *KDD*, pages 1711–1720, 2015.

[5] W. B. Croft, D. Metzler, and T. Strohman. *Search engines: Information retrieval in practice*. Addison-Wesley Reading, 2015.

[6] N. Du, M. Farajtabar, A. Ahmed, A. J. Smola, and L. Song. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *KDD*, pages 347–362, 2015.

[7] M. Efron, J. Lin, J. He, and A. de Vries. Temporal feedback for tweet search with non-parametric density estimation. In *SIGIR*, pages 33–42, 2014.

[8] D. Fisher, A. Jain, M. Keikha, W. B. Croft, and N. Lipka. Evaluating ranking diversity and summarization in microblogs using hashtags. Technical report, University of Massachusetts, 2015.

[9] T. L. Griffiths and M. Steyvers. Finding scientific topics. *PNAS*, 101: 5228–5235, 2004.

[10] T. Iwata, S. Watanabe, T. Yamada, and N. Ueda. Topic tracking model for analyzing consumer purchase behavior. In *IJCAI*, volume 9, pages 1427–1432, 2009.

[11] T. Iwata, T. Yamada, Y. Sakurai, and N. Ueda. Online multiscale dynamic topic models. In *KDD*, pages 663–672. ACM, 2010.

[12] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.

[13] S. Liang and M. de Rijke. Burst-aware data fusion for microblog search. *Information Processing & Management*, pages 89–113, 2015.

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

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

[16] J. Lin, M. Efron, Y. Wang, and G. Sherman. Overview of the TREC 2014 Microblog track. In *TREC 2015*. NIST, 2015.

[17] J. S. Liu. The collapsed Gibbs sampler in Bayesian computations with applications to a gene regulation problem. *J. Am. Stat. Assoc.*, 89(427):958–966, 1994.

[18] T. Minka. Estimating a dirichlet distribution, 2000.

[19] X. Quan, Q. Wang, Y. Zhang, L. Si, and L. Wenyin. Latent discriminative models for social emotion detection with emotional dependency. *ACM Trans. Inf. Syst.*, 34(1):2:1–2:19, 2015.

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

[21] X. Wei and W. B. Croft. LDA-based document models for ad-hoc retrieval. In *SIGIR*, pages 178–185, 2006.

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

[23] J. Xu, P. Wang, G. Tian, B. Xu, and J. Zhao. Short text clustering via convolutional neural networks. In *NAACL-HLT*, pages 62–69, 2015.

[24] Z. Yang, A. Kotov, A. Mohan, and S. Lu. Parametric and non-parametric user-aware sentiment topic models. In *SIGIR*, pages 413–422, 2015.

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

APPENDIX

A. GIBBS SAMPLING DERIVATION FOR DYNAMIC CLUSTERING TOPIC MODEL

We begin with the joint distribution $P(\mathbf{d}_t, \mathbf{z}_t|\boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t)$. We can take advantage of conjugate priors to simplify the integrals. We use the abbreviation $\overline{\phi} = \phi_{t-1,z,v}$ in the following. All other symbols are defined in Section 3 and 4. Due to space limitation, we only provide the derivation for the short term dependence DCT model, and the derivation for the long term DCT model is actually similar.

$$P(\mathbf{d}_t, \mathbf{z}_t|\boldsymbol{\Phi}_{t-1}, \boldsymbol{\Theta}_{t-1}, \alpha_t, \beta_t) = P(\mathbf{d}_t|\mathbf{z}_t, \boldsymbol{\Phi}_{t-1}, \beta_t)P(\mathbf{z}_t|\boldsymbol{\Theta}_{t-1}, \alpha_t)$$

$$= \int P(\mathbf{d}_t|\mathbf{z}_t, \boldsymbol{\Phi}_t)P(\boldsymbol{\Phi}_t|\boldsymbol{\Phi}_{t-1}, \beta_t)d\boldsymbol{\Phi}_t \int P(\mathbf{z}_t|\boldsymbol{\Theta}_t)P(\boldsymbol{\Theta}_t|\boldsymbol{\Theta}_{t-1}, \alpha_t)d\boldsymbol{\Theta}_t$$

$$
\begin{aligned}
&= \int \prod_{d=1}^{|\mathbf{d}_t|} \prod_{i=1}^{N_d} P(v_{t,di}|\phi_{t,z_{di}}) \prod_{z=1}^{Z} P(\phi_{t,z}|\phi_{t-1,z},\beta_t) d\Phi_t \\
&\quad \times \int \prod_{d=1}^{|\mathbf{d}_t|} P(z_{t,d}|\theta_t) P(\theta_t|\theta_{t-1},\alpha_t) d\Theta_t \\
&= \int \prod_{z=1}^{Z} \prod_{v=1}^{V} \phi_{t,z,v}^{n_{t,z,v}} \prod_{z=1}^{Z} P(\phi_{t,z}|\phi_{t-1,z},\beta_t) d\Phi_t \\
&\quad \times \int \prod_{d=1}^{|\mathbf{d}_t|} P(z_{t,d}|\theta_t) P(\theta_t|\theta_{t-1},\alpha_t) d\Theta_t \\
&= \int \prod_{z=1}^{Z} \prod_{v=1}^{V} \phi_{t,z,v}^{n_{t,z,v}} \prod_{z=1}^{Z} \left(\frac{\Gamma(\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}{\prod_{v=1}^{V} \Gamma(\beta_{t,z,v}\overline{\phi})} \prod_{v=1}^{V} \phi_{t,z,v}^{\beta_{t,z,v}\overline{\phi}-1} \right) d\Phi_t \\
&\quad \times \int \prod_{z=1}^{Z} \theta_{t,z}^{m_{t,z}} \left(\frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,z}\theta_{t-1,z})} \right) \prod_{z=1}^{Z} \theta_{t,z}^{\alpha_{t,z}\theta_{t-1,z}-1} d\Theta_t \\
&= \prod_{z=1}^{Z} \frac{\Gamma(\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}{\prod_{v=1}^{V} \Gamma(\beta_{t,z,v}\overline{\phi})} \prod_{z=1}^{Z} \int \prod_{v=1}^{V} \phi_{t,z,v}^{n_{t,z,v}+\beta_{t,z,v}\overline{\phi}-1} d\Phi_t \\
&\quad \times \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,z}\theta_{t-1,z})} \int \prod_{z=1}^{Z} \theta_{t,z}^{m_{t,z}+\alpha_{t,z}\theta_{t-1,z}-1} d\Theta_t \\
&= \prod_{z=1}^{Z} \frac{\Gamma(\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}{\prod_{v=1}^{V} \Gamma(\beta_{t,z,v}\overline{\phi})} \prod_{z=1}^{Z} \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi})} \\
&\quad \times \frac{\Gamma(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})}{\prod_{z=1}^{Z} \Gamma(\alpha_{t,z}\theta_{t-1,z})} \frac{\prod_{z=1}^{Z} \Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}.
\end{aligned}
$$

Applying the chain rule, we can obtain the conditional probability:

$$
\begin{aligned}
&P(z_d = z|\mathbf{z}_{t,-d},\mathbf{d}_t,\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t) = \tfrac{P(\mathbf{z}_t,\mathbf{d}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t)}{P(\mathbf{z}_{t,-d},\mathbf{d}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t)} \\
&\propto \frac{P(\mathbf{z}_t,\mathbf{d}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t)}{P(\mathbf{z}_{t,-d},\mathbf{d}_{t,-d}|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t)} \\
&= \prod_{z=1}^{Z} \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi})} \times \frac{\prod_{z=1}^{Z} \Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})} \Big/ \\
&\quad \prod_{z=1}^{Z} \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})} \times \frac{\prod_{z=1}^{Z} \Gamma(m_{t,z,-d}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,z,-d}+\alpha_{t,z}\theta_{t-1,z})}.
\end{aligned}
$$

Because document d is associated with its own topic z, it becomes

$$
\begin{aligned}
&= \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi})} \times \frac{\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})} \Big/ \\
&\quad \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})} \times \frac{\Gamma(m_{t,z,-d}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(\sum_{z=1}^{Z} m_{t,z,-d}+\alpha_{t,z}\theta_{t-1,z})} \\
&= \frac{\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}-1)} \frac{\Gamma(\sum_{z=1}^{Z}(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})-1)}{\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})} \\
&\quad \times \frac{\prod_{v=1}^{V} \Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi})}{\prod_{v=1}^{V} \Gamma(n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})} \frac{\Gamma(\sum_{v=1}^{V} n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})}{\Gamma(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi})} \\
&= \frac{\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})}{\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}-1)} \frac{\Gamma(\sum_{z=1}^{Z}(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})-1)}{\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})} \\
&\quad \times \frac{\prod_{v\in d} \Gamma(n_{t,z,v}+\beta_{t,z,v})}{\prod_{v\in d} \Gamma(n_{t,z,v,-d}+\beta_{t,z,v})} \frac{\Gamma(n_{t,z,-d}+\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}{\Gamma(n_{t,z,-d}+N_d+\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}.
\end{aligned}
$$

Applying $\Gamma(x) = (x-1)\Gamma(x-1)$ and $\Gamma(x+m) = \prod_{i=1}^{m}(x+i-1)\Gamma(x)$, the above becomes

$$
\begin{aligned}
&= \frac{m_{t,z}+\alpha_{t,z}\theta_{t-1,z}-1}{\sum_{z=1}^{Z}(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})-1} \frac{\frac{\prod_{v\in d} \Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi})}{\prod_{v\in d} \Gamma(n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi})}}{\prod_{i=1}^{N_d}(n_{t,z,-d}+i-1+\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})} \\
&= \frac{m_{t,z}+\alpha_{t,z}\theta_{t-1,z}-1}{\sum_{z=1}^{Z}(m_{t,z}+\alpha_{t,z}\theta_{t-1,z})-1} \\
&\quad \times \frac{\prod_{v\in d} \prod_{j=1}^{N_{d,v}} (n_{t,z,v,-d}+\beta_{t,z,v}\overline{\phi}+j-1)}{\prod_{i=1}^{N_d}(n_{t,z,-d}+i-1+\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})}
\end{aligned}
$$

B. DERIVATION OF THE UPDATE RULES

We apply a fixed-point iteration for estimating the parameters α_t and β_t by maximizing the joint distribution $P(\mathbf{d}_t,\mathbf{z}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t)$. Here we only show the derivation for short term dependence DCT model, while that for long term dependence DCT model is similar. Instead of maximizing the joint distribution directly, we try to maximize the following:

$$
\begin{aligned}
&\log P(\mathbf{d}_t,\mathbf{z}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\alpha_t,\beta_t) \\
&= \sum_{z=1}^{Z} \log\Gamma(\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi}) - \sum_{z=1}^{Z} \log\Gamma(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi}) \\
&\quad + \sum_{z=1}^{Z}\sum_{v=1}^{V} \log\Gamma(n_{t,z,v}+\beta_{t,z,v}\overline{\phi}) - \sum_{z=1}^{Z}\sum_{v=1}^{V} \log\Gamma(\beta_{t,z,v}\overline{\phi}) \\
&\quad + \log\Gamma(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z,v}) - \log\Gamma(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) \\
&\quad + \sum_{z=1}^{Z} \log\Gamma(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \sum_{z=1}^{Z} \log\Gamma(\alpha_{t,z}\theta_{t-1,z})
\end{aligned}
$$

Using the bounds [18]: for any $x^* \in \mathbb{R}^+$, $n \in \mathbb{Z}^+$ and x^*'s estimation x:

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

and

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

supposing $\alpha^*_{t,z}$ is the optimal parameter in the next fixed-point iteration, it follows that

$$
\begin{aligned}
&\log P(\mathbf{d}_t,\mathbf{z}_t|\mathbf{\Phi}_{t-1},\mathbf{\Theta}_{t-1},\{\alpha_{t,1},\ldots\alpha^*_{t,z},\ldots,\alpha_{t,Z}\},\beta_t) \geq B(\alpha^*_{t,z}) \\
&= \alpha_{t,z}\theta_{t-1,z}(\Psi(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \Psi(\alpha_{t,z}\theta_{t-1,z})) \log\alpha^*_{t,z}\theta_{t-1,z} \\
&\quad - \alpha^*_{t,z}\theta_{t-1,z}\left(\Psi(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z})\right) + C,
\end{aligned}
$$

where C is function not containing the term $\alpha^*_{t,z}$ and thus will be integrated out by taking $\frac{\partial(\cdot)}{\partial\alpha^*_{t,z}}$ to $\alpha^*_{t,z}$. Then, we let

$$
\begin{aligned}
\frac{\partial B(\alpha^*_{t,z})}{\partial\alpha^*_{t,z}} &= \frac{\alpha_{t,z}\theta_{t-1,z}(\Psi(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \Psi(\alpha_{t,z}\theta_{t-1,z}))}{\alpha^*_{t,z}} \\
&\quad - \theta_{t-1,z}\left(\Psi(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \Psi(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})\right) \\
&= 0,
\end{aligned}
$$

which results in

$$
\alpha^*_{t,z} = \frac{\alpha_{t,z}(\Psi(m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \Psi(\alpha_{t,z}\theta_{t-1,z}))}{\Psi(\sum_{z=1}^{Z} m_{t,z}+\alpha_{t,z}\theta_{t-1,z}) - \Psi(\sum_{z=1}^{Z} \alpha_{t,z}\theta_{t-1,z})},
$$

where $\Psi(\cdot)$ is the digamma function defined by $\Psi(x) = \frac{\partial\log\Gamma(x)}{\partial x}$.

Following the same derivation, again supposed $\beta^*_{t,z,v}$ is the optimal parameter in the next fixed-point iteration, we have

$$
\beta^*_{t,z,v} = \frac{\sum_{z=1}^{Z} \beta_{t,z,v}\overline{\phi}\left(\Psi(n_{t,z,v}+\beta_{t,z,v}\overline{\phi}) - \Psi(\beta_{t,z,v}\overline{\phi})\right)}{\sum_{z=1}^{Z} \overline{\phi}\left(\Psi(\sum_{v=1}^{V} n_{t,z,v}+\beta_{t,z,v}\overline{\phi}) - \Psi(\sum_{v=1}^{V} \beta_{t,z,v}\overline{\phi})\right)}.
$$