Sequential Modeling of Topic Dynamics with Multiple Timescales

Article *in* ACM Transactions on Knowledge Discovery from Data · February 2012

DOI: 10.1145/2086737.2086739 · Source: DBLP

Tomoharu Iwata
Nippon Telegraph and Telephone
38 PUBLICATIONS **645** CITATIONS
SEE PROFILE

Takeshi Yamada
NTT Communication Science Laboratories
60 PUBLICATIONS **2,582** CITATIONS
SEE PROFILE

Yasushi Sakurai
Kumamoto University
69 PUBLICATIONS **1,630** CITATIONS
SEE PROFILE

Naonori Ueda
Nippon Telegraph and Telephone
188 PUBLICATIONS **6,141** CITATIONS
SEE PROFILE

Sequential Modeling of Topic Dynamics with Multiple Timescales

TOMOHARU IWATA, NTT Communication Science Laboratories
TAKESHI YAMADA, NTT Science and Core Technology Laboratory Group
YASUSHI SAKURAI and NAONORI UEDA, NTT Communication Science Laboratories

We propose an online topic model for sequentially analyzing the time evolution of topics in document collections. Topics naturally evolve with multiple timescales. For example, some words may be used consistently over one hundred years, while other words emerge and disappear over periods of a few days. Thus, in the proposed model, current topic-specific distributions over words are assumed to be generated based on the multiscale word distributions of the previous epoch. Considering both the long- and short-timescale dependency yields a more robust model. We derive efficient online inference procedures based on a stochastic EM algorithm, in which the model is sequentially updated using newly obtained data; this means that past data are not required to make the inference. We demonstrate the effectiveness of the proposed method in terms of predictive performance and computational efficiency by examining collections of real documents with timestamps.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*; I.2.6 [**Artificial Intelligence**]: Learning; I.5.1 [**Pattern Recognition**]: Models—*Statistical*

General Terms: Algorithms

Additional Key Words and Phrases: Topic model, time-series analysis, online learning, multiscale

ACM Reference Format:
Iwata, T., Yamada, T., Sakurai, Y., and Ueda, N. 2011. Sequential modeling of topic dynamics with multiple timescales. ACM Trans. Knowl. Discov. Data 5, 4, Article 19 (February 2012), 27 pages.
DOI = 10.1145/2086737.2086739 http://doi.acm.org/10.1145/2086737.2086739

1. INTRODUCTION

Great interest is being shown in developing topic models that can analyze and summarize the dynamics of document collections, such as scientific papers, news articles, and blogs [Ahmed and Xing 2010; AlSumait et al. 2008; Blei and Lafferty 2006; Canini et al. 2009; Iwata et al. 2009; Nallapati et al. 2007; Wang et al. 2008; Wang and McCallum 2006; Watanabe et al. 2011; Wei et al. 2007; Zhang et al. 2010]. A topic model is a hierarchical probabilistic model, in which a document is modeled as a mixture of topics, and a topic is modeled as a probability distribution over words. Topic models are successfully used in a wide variety of applications including information retrieval [Blei et al. 2003], collaborative filtering [Hofmann 2003], and visualization [Iwata et al. 2008] as well as the analysis of dynamics.

In this article, we propose a topic model that permits the sequential analysis of the dynamics of topics with multiple timescales, which we call the *Multiscale Dynamic Topic Model* (MDTM), and its efficient online inference procedures. Topics naturally

Authors' addresses: T. Iwata, Y. Sakurai, and N. Ueda, NTT Communication Science Laboratories; email: iwata.tomoharu@lab.ntt.co,jp; T. Yamada, NTT Science and Core Technology Laboratory Group.

DOI 10.1145/2086737.2086739 http://doi.acm.org/10.1145/2086737.2086739

evolve with multiple timescales. Let us consider the topic “politics” in a news article collection as an example. There are some words that appear frequently over many years, such as “constitution,” “congress,” and “president.” On the other hand, some words, such as the names of congress members, may appear frequently over periods of tens of years, and other words, such as the names of bills under discussion, may appear for only a few days. Thus, in MDTM, current topic-specific distributions over words are assumed to be generated based on estimates of multiple timescale word distributions of the previous epoch. Using these multiscale priors improves the predictive performance of the model because the information loss is reduced by considering both the long- and short-timescale dependency. Furthermore, by using multiple estimates, the variance of the inference is reduced when compared with models that have a single timescale.

The online inference and parameter estimation processes can be achieved efficiently based on a stochastic expectation-maximization (EM) algorithm, in which the model is sequentially updated using newly obtained data; past data does not need to be stored and processed to make new inferences. Some topics may exhibit strong long-timescale dependence, and others may exhibit strong short-timescale dependence. Furthermore, the dependence may change over time. Therefore, we infer these dependencies for each timescale, for each topic, and for each epoch. By inferring the dependencies from the observed data, MDTM can flexibly adapt to topic dynamics. A disadvantage of online inference is that it can be more unstable than batch inference. With MDTM, the stability can be improved by smoothing using multiple estimates with different timescales.

The remainder of this article is organized as follows. In Section 2, we briefly review related work. In Section 3, we formulate a topic model for multiscale dynamics, and describe its online inference procedures. In Section 4, we demonstrate the effectiveness of the proposed method by analyzing the dynamics of real document collections. Finally, we present concluding remarks and a discussion of future work in Section 5.

2. RELATED WORK

2.1 Topic Modeling

A number of methods for analyzing the evolution of topics in document collections have been proposed, such as the dynamic topic model [Blei and Lafferty 2006], topic over time [Wang and McCallum 2006], online latent Dirichlet allocation [AlSumait et al. 2008], and topic tracking model [Iwata et al. 2009]. However, none of these methods take account of multiscale dynamics. For example, the dynamic topic model (DTM) [Blei and Lafferty 2006] depends only on the previous epoch distribution. On the other hand, MDTM depends on multiple distributions with different timescales. Therefore, with MDTM, we can model multiple timescale dependence, and so infer the current model more robustly. Moreover, while DTM uses a Gaussian distribution to account for the dynamics, the proposed model uses conjugate priors. Therefore, inference in MDTM is relatively simple compared with that in DTM.

There are two approaches for incorporating dynamics in topic models. The first approach models dynamics by defining evolution on the hidden variables [Blei and Lafferty 2006; Nallapati et al. 2007]. The second approach uses the topics in the past and current epochs to define a prior for future epochs [Ahmed and Xing 2008, 2010; Blei and Frazier 2010; Gerrish and Blei 2010]. The proposed model uses the second approach; however it uses the topics for modeling word distributions with multiple scales.

Recently, online inference algorithms for topic models have been proposed [Canini et al. 2009; Hoffman et al. 2010; Sato et al. 2010], which can process one document at a time. These algorithms are standard topic models but adapted to handle massive document collections, including those arriving in a stream, and they do not model temporal information or dynamics, which is our goal. Another difference is that the proposed inference algorithm is online in the sense that it learns one epoch at a time, instead of one document at a time. However, it is straightforward to handle the latter case by incorporating the algorithm described in Hoffman et al. [2010] and Sato et al. [2010].

2.2 Multiscale Dynamics

The multiscale topic tomography model (MTTM) [Nallapati et al. 2007] can analyze the evolution of topics at various resolutions of timescales by assuming nonhomogeneous Poisson processes. In contrast, MDTM models the topic evolution within the Dirichlet-multinomial framework in the same way as most topic models including latent Dirichlet allocation [Blei et al. 2003]. Another advantage of MDTM over MTTM is that it can make inferences in an online fashion. Therefore, MDTM can greatly reduce both the computational cost and the memory requirement because past data need not be stored. Online inference is essential for modeling the dynamics of document collections in which large numbers of documents continue to accumulate at any given moment, such as news articles and blogs. This is because it is necessary to adapt to the new data immediately for topic tracking, and it is impractical to prepare sufficient memory capacity to store all past data.

The multiscale analysis of time-series data such as wavelets is remotely related. AWSOM [Papadimitriou et al. 2003] is one of the first streaming methods for forecasting and is designed to discover arbitrary periodicities in single time sequences. Sakurai et al. [2005] proposed BRAID, which builds a multilevel window structure and efficiently detects lag correlations between data streams. Singular value decomposition (SVD) is used for analyzing multiscale patterns in streaming data [Papadimitriou et al. 2005] as well as topic models. However, since SVD assumes Gaussian noise, it is inappropriate for discrete data such as document collections [Hofmann 1999].

3. PROPOSED METHOD

3.1 Preliminaries

In the proposed model, documents are assumed to be generated sequentially at each epoch. Suppose we have a set of D_t documents at the current epoch, t, and each document is represented by $\boldsymbol{w}_{t,d} = \{w_{t,d,n}\}_{n=1}^{N_{t,d}}$, i.e. the set of words in the document, where $1 \leq d \leq D_t$. Our notation is summarized in Table I. We assume that epoch t is a discrete variable, and we can set the time period for an epoch arbitrarily at, for example, one day or one year.

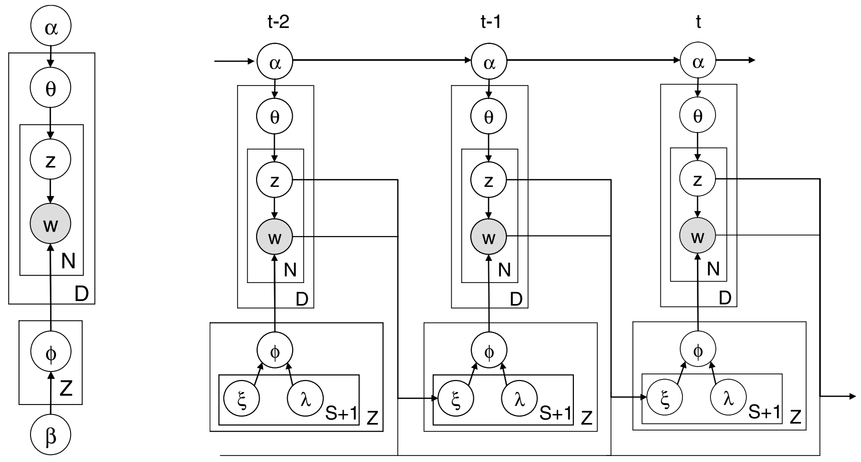
Before introducing the proposed model, we review latent Dirichlet allocation (LDA) [Blei et al. 2003; Griffiths and Steyvers 2004], which forms the basis of the proposed model. LDA assumes the following generative process of words in a document. Each document has topic proportions $\boldsymbol{\theta}_{t,d}$. For each of the $N_{t,d}$ words in the document, topic $z_{t,d,n}$ is chosen from the topic proportions, and then word $w_{t,d,n}$ is generated from a topic-specific multinomial distribution over words $\boldsymbol{\phi}_{z_{t,d,n}}$. Topic proportions $\boldsymbol{\theta}_{t,d}$ and word distributions $\boldsymbol{\phi}_z$, are assumed to be generated according to symmetric Dirichlet distributions. Figure 1 (a) shows a graphical model representation of LDA, where the shaded and unshaded nodes indicate observed and latent variables, respectively.

3.2 Model

We consider a set of multiple timescale distributions over words for each topic to incorporate multiple timescale properties. To account for the influence of the past at

Table I. Notation

Symbol	Description
D_t	number of documents at epoch t
$N_{t,d}$	number of words in the dth document at epoch t
W	number of unique words
$w_{t,d,n}$	nth word in the dth document at epoch t, $w_{t,d,n} \in \{1, \cdots, W\}$
Z	number of topics
$z_{t,d,n}$	topic of the nth word in the dth document at epoch t, $z_{t,d,n} \in \{1, \cdots, Z\}$
S	number of scales
L_s	number of past-periods to consider in scale s
$\boldsymbol{\theta}_{t,d}$	multinomial distribution over topics for the dth document at epoch t, $\boldsymbol{\theta}_{t,d} = \{\theta_{t,d,z}\}_{z=1}^{Z}$, $\theta_{t,d,z} \geq 0$, $\sum_z \theta_{t,d,z} = 1$
$\boldsymbol{\phi}_{t,z}$	multinomial distribution over words for the zth topic at epoch t, $\boldsymbol{\phi}_{t,z} = \{\phi_{t,z,w}\}_{w=1}^{W}$, $\phi_{t,z,w} \geq 0$, $\sum_w \phi_{t,z,w} = 1$
$\boldsymbol{\xi}_{t,z}^{(s)}$	multinomial distribution over words for the zth topic with scale s at epoch t, $\boldsymbol{\xi}_{t,z}^{(s)} = \{\xi_{t,z,w}^{(s)}\}_{w=1}^{W}$, $\xi_{t,z,w}^{(s)} \geq 0$, $\sum_w \xi_{t,z,w}^{(s)} = 1$



(a) LDA (b) MDTM

Fig. 1. Graphical models of, (a) latent Dirichlet allocation, and (b) the multiscale dynamic topic model.

different timescales on the current epoch, we assume that current topic-specific word distributions $\boldsymbol{\phi}_{t,z}$ are generated according to the multiscale word distributions at the previous epoch $\{\boldsymbol{\xi}_{t-1,z}^{(s)}\}_{s=1}^{S}$. The multiscale word distribution $\boldsymbol{\xi}_{t-1,z}^{(s)} = \{\xi_{t-1,z,w}^{(s)}\}_{w=1}^{W}$ represents a distribution over words of topic z with scale s at epoch $t-1$, and is defined as follows.

$$\xi_{t,z,w}^{(s)} = \frac{\sum_{t'=t-L_s+1}^{t} N_{t',z,w}}{\sum_w \sum_{t'=t-L_s+1}^{t} N_{t',z,w}}, \tag{1}$$

where L_s is the length of scale s and $N_{t,z,w}$ is the number of times word w was assigned to topic z at epoch t. In particular, we use the following asymmetric Dirichlet distribution for the prior of the current word distribution $\boldsymbol{\phi}_{t,z}$, in which the Dirichlet

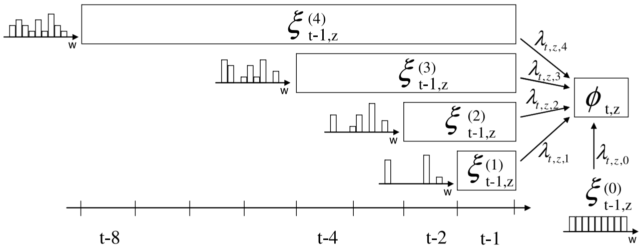


Fig. 2. Multiscale word distributions at epoch t with $S = 4$. Each histogram shows $\boldsymbol{\xi}^{(s)}_{t-1,z}$, which is a multinomial distribution over words with timescale s.

parameter is defined such that the mean of $\boldsymbol{\phi}_{t,z}$ becomes proportional to the weighted sum of multiscale word distributions at the previous epoch,

$$\boldsymbol{\phi}_{t,z} \sim \text{Dirichlet}\left(\sum_{s=0}^{S} \lambda_{t,z,s}\boldsymbol{\xi}^{(s)}_{t-1,z}\right), \tag{2}$$

where $\lambda_{t,z,s}$ is a weight for scale s in topic z at epoch t, and $\lambda_{t,z,s} > 0$. By estimating weights $\{\lambda_{t,z,s}\}_{s=0}^{S}$ for each epoch, for each topic, and for each timescale using the current data as described in Section 3.3, MDTM can flexibly respond to the influence of the previous short- and long-timescale distributions on the current distribution. In this way, the estimated multiscale word distributions $\{\boldsymbol{\xi}^{(s)}_{t-1,z}\}_{s=1}^{S}$ at the previous epoch constitute the hyperparameters in the current epoch. Their estimation will be described in Section 3.4.

There are many different ways of setting the scales, but for generality and simplicity, we set them so that $\boldsymbol{\xi}^{(s)}_{t,z}$ indicates the word distribution from $t - 2^{s-1} + 1$ to t, where a larger s represents longer timescale, and $\boldsymbol{\xi}^{(s=1)}_{t,z}$ is equivalent to the estimate of a unit-time word distribution $\boldsymbol{\phi}_{t,z}$. We use a uniform word distribution $\xi^{(s=0)}_{t,z,w} = W^{-1}$ for scale $s = 0$. This uniform distribution is used to avoid the zero probability problem. Figure 2 shows multiscale word distributions in this setting. Multiscale word distributions $\boldsymbol{\xi}^{(s)}_{t,z}$ are likely to become smoother as the timescale increases and exhibit more peaks as the timescale decreases. By using the information presented in these various timescales as the prior for the current distribution with weights, we can infer the current distribution more robustly. In Figure 2, we use timescales of a geometric progression: $1, 2, 4, \ldots, 2^{S-1}$. Instead of using 2^{s-1} epochs for scale s, we could use any number of epochs. For example, if we know that the given data exhibit a periodicity e.g., of one week and one month, we can use the scale of one week for $s = 1$ and one month for $s = 2$. In such a case, we can still estimate the parameters in a similar way to that used with the algorithm described in Section 3.4. Typically, we do not know the periodicity of given data in advance, we therefore consider a simple scale setting in this article.

In LDA, topic proportions $\boldsymbol{\theta}_{t,d}$ are sampled from a Dirichlet distribution. To capture the dynamics of topic proportions with MDTM, we assume that the Dirichlet parameters $\boldsymbol{\alpha}_t = \{\alpha_{t,z}\}_{z=1}^{Z}$ depend on the previous parameters. In particular, we use the following Gamma prior for a Dirichlet parameter of topic z at epoch t,

$$\alpha_{t,z} \sim \text{Gamma}(\gamma\alpha_{t-1,z}, \gamma), \tag{3}$$

where the mean is $\alpha_{t-1,z}$, and the variance is $\alpha_{t-1,z}/\gamma$. By using this prior, the mean is the same as that at the previous epoch unless otherwise indicated by the new data.

Parameter γ controls the temporal consistency of the topic proportion prior $\alpha_{t,z}$. When γ is high, $\alpha_{t,z}$ is likely to be close to $\alpha_{t-1,z}$.

Assuming that we have already calculated the multiscale parameters at epoch $t-1$, $\Xi_{t-1} = \{\{\boldsymbol{\xi}^{(s)}_{t-1,z}\}_{s=0}^{S}\}_{z=1}^{Z}$ and $\boldsymbol{\alpha}_{t-1} = \{\alpha_{t-1,z}\}_{z=1}^{Z}$, and given parameters, γ and $\boldsymbol{\Lambda}_t = \{\{\lambda_{t,z,s}\}_{s=0}^{S}\}_{z=1}^{Z}$, MDTM is characterized by the following process for a set of documents $\boldsymbol{W}_t = \{\boldsymbol{w}_{t,d}\}_{d=1}^{D_t}$ at epoch t.

(1) For each topic $z = 1, \cdots, Z$:
  (a) Draw topic proportion prior
    $\alpha_{t,z} \sim \text{Gamma}(\gamma\alpha_{t-1,z}, \gamma)$,
  (b) Draw word distribution
    $\boldsymbol{\phi}_{t,z} \sim \text{Dirichlet}(\sum_s \lambda_{t,z,s}\boldsymbol{\xi}^{(s)}_{t-1,z})$,
(2) For each document $d = 1, \cdots, D_t$:
  (a) Draw topic proportions
    $\boldsymbol{\theta}_{t,d} \sim \text{Dirichlet}(\boldsymbol{\alpha}_t)$,
  (b) For each word $n = 1, \cdots, N_{t,d}$:
    i.Draw topic
      $z_{t,d,n} \sim \text{Multinomial}(\boldsymbol{\theta}_{t,d})$,
    ii.Draw word
      $w_{t,d,n} \sim \text{Multinomial}(\boldsymbol{\phi}_{t,z_{t,d,n}})$.

Figure 1(b) shows a graphical model representation of MDTM.

3.3 Online Inference

We present an online inference algorithm for MDTM, that sequentially updates the model at each epoch using the newly obtained document set and the multiscale model of the previous epoch. The information in the data up to, and including, the previous epoch is aggregated into the previous multiscale model. The online inference and parameter estimation can be efficiently achieved by a stochastic EM algorithm [Andrieu et al. 2003], in which the collapsed Gibbs sampling of latent topics, $\boldsymbol{Z}_t$, and the maximum a posteriori (MAP) estimation of hyperparameters, $\boldsymbol{\alpha}_t$ and $\boldsymbol{\Lambda}_t$, are alternately performed.

We assume a set of documents $\boldsymbol{W}_t$ at current epoch t, and estimates of parameters from the previous epoch $\boldsymbol{\alpha}_{t-1}$, Ξ_{t-1}, γ and $\boldsymbol{\Lambda}_t$ are given. The joint distribution on the set of documents, the set of topics, and the topic proportion priors given the parameters are defined as follows.

$$P(\boldsymbol{W}_t, \boldsymbol{Z}_t, \boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1}, \gamma, \Xi_{t-1}, \boldsymbol{\Lambda}_t) = P(\boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1}, \gamma)P(\boldsymbol{Z}_t|\boldsymbol{\alpha}_t)P(\boldsymbol{W}_t|\boldsymbol{Z}_t, \Xi_{t-1}, \boldsymbol{\Lambda}_t), \quad (4)$$

where $\boldsymbol{Z}_t = \{\{z_{t,d,n}\}_{n=1}^{N_{t,d}}\}_{d=1}^{D_t}$ represents a set of topics. The first term on the right hand side of (4) is as follows using (3).

$$P(\boldsymbol{\alpha}_t|\boldsymbol{\alpha}_{t-1}, \gamma) = \prod_z \frac{\gamma^{\gamma\alpha_{t-1,z}}\alpha_{t,z}^{\gamma\alpha_{t-1,z}-1}\exp(-\gamma\alpha_{t,z})}{\Gamma(\gamma\alpha_{t-1,z})}, \quad (5)$$

where $\Gamma(\cdot)$ is the gamma function. We can integrate out the multinomial distribution parameters in MDTM, $\{\boldsymbol{\theta}_{t,d}\}_{d=1}^{D_t}$ and $\{\boldsymbol{\phi}_{t,z}\}_{z=1}^{Z}$, by taking advantage of Dirichlet-multinomial conjugacy. The second term is calculated by

$$P(\boldsymbol{Z}_t|\boldsymbol{\alpha}_t) = \prod_{d=1}^{D_t}\int\prod_{n=1}^{N_{t,d}} P(z_{t,d,n}|\boldsymbol{\theta}_{t,d})P(\boldsymbol{\theta}_{t,d}|\boldsymbol{\alpha}_t)d\boldsymbol{\theta}_{t,d}, \quad (6)$$

and we have the following equation by integrating out $\{\boldsymbol{\theta}_{t,d}\}_{d=1}^{D_t}$.

$$P(\boldsymbol{Z}_t|\boldsymbol{\alpha}_t) = \left(\frac{\Gamma\left(\sum_z \alpha_{t,z}\right)}{\prod_z \Gamma\left(\alpha_{t,z}\right)} \right)^{D_t} \prod_d \frac{\prod_z \Gamma\left(N_{t,d,z} + \alpha_{t,z}\right)}{\Gamma\left(N_{t,d} + \sum_z \alpha_{t,z}\right)}, \tag{7}$$

where $N_{t,d,z}$ is the number of words in the dth document assigned to topic z at epoch t, and $N_{t,d} = \sum_z N_{t,d,z}$. Similarly, by integrating out $\{\boldsymbol{\phi}_{t,z}\}_{z=1}^{Z}$, the third term is given as follows.

$$P(\boldsymbol{W}_t|\boldsymbol{Z}_t, \Xi_{t-1}, \boldsymbol{\Lambda_t}) = \prod_z \frac{\Gamma\left(\sum_s \lambda_{t,z,s}\right)}{\prod_w \Gamma\left(\sum_s \lambda_{t,z,s}\xi^{(s)}_{t-1,z,w}\right)} \frac{\prod_w \Gamma\left(N_{t,z,w} + \sum_s \lambda_{t,z,s}\xi^{(s)}_{t-1,z,w}\right)}{\Gamma\left(N_{t,z} + \sum_s \lambda_{t,z,s}\right)}, \tag{8}$$

where $N_{t,z,w}$ is the number of times word w was assigned to topic z at epoch t, and $N_{t,z} = \sum_w N_{t,z,w}$.

The inference of the latent topics $\boldsymbol{Z}_t$ can be efficiently computed by using collapsed Gibbs sampling. Let $j = (t, d, n)$ for notational convenience, and z_j be the assignment of a latent topic to the nth word in the dth document at epoch t. Then, given the current state of all but one variable z_j, a new value for z_j is sampled from the following probability.

$$P(z_j = k|\boldsymbol{W}_t, \boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \propto \frac{N_{t,d,k\backslash j} + \alpha_{t,k}}{N_{t,d\backslash j} + \sum_z \alpha_{t,z}} \frac{N_{t,k,w_j\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w_j}}{N_{t,k\backslash j} + \sum_s \lambda_{t,k,s}}, \tag{9}$$

where $\backslash j$ represents the count yielded by excluding the nth word in the dth document. See Appendix A for the derivation.

The parameters $\boldsymbol{\alpha}_t$ and $\boldsymbol{\Lambda}_t$ are estimated by maximizing the joint distribution (4). The fixed-point iteration method described in Minka [2000] can be used for maximizing the joint distribution as follows.

$$\alpha_{t,z} \leftarrow \frac{\gamma\alpha_{t-1,z} - 1 + \alpha^{\text{old}}_{t,z} \sum_d \left[\Psi\left(N_{t,d,z} + \alpha^{\text{old}}_{t,z}\right) - \Psi\left(\alpha^{\text{old}}_{t,z}\right)\right]}{\gamma + \sum_d \left[\Psi\left(N_{t,d} + \sum_{z'} \alpha^{\text{old}}_{t,z'}\right) - \Psi\left(\sum_{z'} \alpha^{\text{old}}_{t,z'}\right)\right]}, \tag{10}$$

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

$$\lambda_{t,z,s} \leftarrow \lambda^{\text{old}}_{t,z,s} \frac{\sum_w \xi^{(s)}_{t-1,z,w} \left[\Psi\left(N_{t,z,w} + \sum_{s'} \lambda^{\text{old}}_{t,z,s'}\xi^{(s')}_{t-1,z,w}\right) - \Psi\left(\sum_{s'} \lambda^{\text{old}}_{t,z,s'}\xi^{(s')}_{t-1,z,w}\right)\right]}{\Psi\left(N_{t,z} + \sum_{s'} \lambda^{\text{old}}_{t,z,s'}\right) - \Psi\left(\sum_{s'} \lambda^{\text{old}}_{t,z,s'}\right)}. \tag{11}$$

See Appendix B and Appendix C for the derivation. By iterating Gibbs sampling with (9) and maximum likelihood estimation with (10) and (11), we can infer latent topics while optimizing the parameters. Since MDTM uses the past distributions as the current prior, the label switching problem [Stephens 2000] is not likely to occur when the estimated $\lambda_{t,z,s}$ is high, which implies that current topics strongly depend on the previous distributions. Label switching can occur when the estimated $\lambda_{t,z,s}$ is low. By allowing low $\lambda_{t,z,s}$, which is estimated from the given data at each epoch and each topic, MDTM can adapt flexibly to changes even if existing topics disappear and new topics appear in midstream.

The time complexity of one iteration of our Gibbs sampling is $O(D_t N_t Z)$, where N_t is the average number of words in documents at epoch t, and it does not depend on the number of scales. The time complexity of one iteration of hyperparameter estimation increases linearly with the number of scales.

3.4 Efficient Estimation of Multiscale Word Distributions

By using the topic assignments obtained after iterating the stochastic EM algorithm, we can estimate multiscale word distributions. Since $\xi^{(s)}_{t,z,w}$ represents the probability of word w in topic z from $t - 2^{s-1} + 1$ to t, the estimation is as follows.

$$\xi^{(s)}_{t,z,w} = \frac{\hat{N}^{(s)}_{t,z,w}}{\sum_w \hat{N}^{(s)}_{t,z,w}} = \frac{\sum_{t'=t-2^{s-1}+1}^{t} \hat{N}_{t',z,w}}{\sum_w \sum_{t'=t-2^{s-1}+1}^{t} \hat{N}_{t',z,w}}, \tag{12}$$

where $\hat{N}^{(s)}_{t,z,w}$ is the expected number of times word w was assigned to topic z from epochs $t - 2^s + 1$ to t, and $\hat{N}_{t,z,w}$ is the expected number of times word w is assigned to topic z at epoch t. The expected number is calculated by $\hat{N}_{t,z,w} = N_{t,z}\hat{\phi}_{t,z,w}$, where $\hat{\phi}_{t,z,w}$ is a point estimate of the probability of word w in topic z at epoch t. Although we integrate out $\phi_{t,z,w}$, we can recover its point estimate as follows.

$$\hat{\phi}_{t,z,w} = \frac{N_{t,z,w} + \sum_s \lambda_{t,z,s}\xi^{(s)}_{t-1,z,w}}{N_{t,z} + \sum_s \lambda_{t,z,s}}. \tag{13}$$

While we can use the actual number of times $N_{t,z,w}$ in (12), we use the expected number of times $\hat{N}_{t,z,w}$ to constrain the estimate of $\xi^{(s=1)}_{t,z,w}$ to be the estimate of $\phi_{t,z,w}$ as follows.

$$\xi^{(s=1)}_{t,z,w} = \frac{\hat{N}_{t,z,w}}{\sum_w \hat{N}_{t,z,w}} = \hat{\phi}_{t,z,w}. \tag{14}$$

Note that the value $\hat{N}^{(s)}_{t,z,w}$ can be updated sequentially from the previous value $\hat{N}^{(s)}_{t-1,z,w}$ as follows.

$$\hat{N}^{(s)}_{t,z,w} \leftarrow \hat{N}^{(s)}_{t-1,z,w} + \hat{N}_{t,z,w} - \hat{N}_{t-2^{s-1},z,w}. \tag{15}$$

Therefore, $\hat{N}^{(s)}_{t,z,w}$ can be efficiently updated through just two additions rather than 2^{s-1} additions. We refer to it as the naive updating method.

LEMMA 3.1. *The naive updating method requires $O(2^S Z W)$ memory for updating multiscale word distributions.*

PROOF. The naive updating method subtracts $\hat{N}_{t-2^{S-1},z,w}$ for updating $\hat{N}^{(s)}_{t,z,w}$. Therefore, it has to maintain $O(2^S)$ values of $\hat{N}_{t-2^{S-1},z,w}$ from $t - 2^{S-1}$ to $t - 1$. We need the $O(2^S)$ values for each of Z topics and each of W words. Thus, it would take $O(2^S Z W)$ space. □

Since the memory requirement increases exponentially with the number of scales in the naive updating method, this requirement prevents us from modeling long-timescale dynamics. Thus, we consider approximating the update by decreasing the update frequency for long-timescale distributions as in Algorithm 1, which is linear with respect to the number of scales.

LEMMA 3.2. *The proposed updating method requires $O(SZ W)$ memory for updating multiscale word distributions.*

PROOF. The proposed updating method can update $\hat{N}^{(s)}_{t,z,w}$ using the previous value at scale $s - 1$, $\hat{N}^{(s-1)}_{t-1,z,w}$, and the current value at scale $s - 1$, $\hat{N}^{(s-1)}_{t,z,w}$. When we update

ALGORITHM 1: Algorithm for the approximate update of $\hat{N}^{(s)}_{t,z,w}$.

```
$\hat{N}^{(1)}_{t,z,w} \leftarrow \hat{N}_{t,z,w}$
for $s = 2, \cdots, S$ do
    if $t \bmod 2^{s-1} = 0$ then
        $\hat{N}^{(s)}_{t,z,w} \leftarrow \hat{N}^{(s-1)}_{t,z,w} + \hat{N}^{(s-1)}_{t-1,z,w}$
    else
        $\hat{N}^{(s)}_{t,z,w} \leftarrow \hat{N}^{(s)}_{t-1,z,w}$
    end
end
```

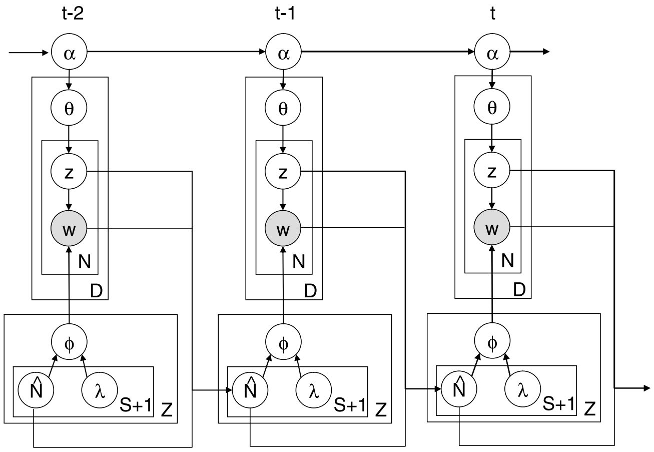


Fig. 3. Graphical models of the online multiscale dynamic topic model.

from $s = 1$ to $s = S$, we must store only $O(S)$ values of $\hat{N}^{(s-1)}_{t-1,z,w}$ at the previous epoch. We need the $O(S)$ values for each of Z topics and each of W words. Thus, it would take $O(SZW)$ space. □

Figure 4 shows approximate updating of $\hat{N}^{(s)}_{t,z,w}$ with $S = 3$ from $t = 4$ to $t = 8$. Each box represents $\hat{N}_{t',z,w}$, where the number represents epoch t'. The boxes in the bottom row indicate that the $\hat{N}_{t,z,w}$ value is newly calculated at epoch t. Each row at each epoch represents $\hat{N}^{(s)}_{t,z,w}$ as sum of the corresponding boxes, and the shaded boxes indicate that the values are updated from the previous values. $\hat{N}^{(s)}_{t,z,w}$ is updated every 2^{s-1} epoch. For example, value $\hat{N}^{(s)}_{t,z,w}$ at scale $s = 1$ is updated every epoch, the value at scale $s = 2$ is updated every two epochs, and the value at scale $s = 3$ is updated every four epochs. The value at scale $s = 1$ can be updated by replacing it with the newly calculated value. The value at scale $s = 2$ can be updated by adding the newly calculated value to the previous value at scale $s = 1$; see epochs $t = 6$ or $t = 8$ in Figure 4. The value at scale $s = 3$ can be updated by summing the newly calculated value, the previous value at scale $s = 1$ and the previous value at scale $s = 2$; see epochs $t = 8$ in Figure 4. As seen in the preceding, we can calculate the value by using values at the previous epoch with this approximate updating; we do not need to store the past values before the previous epoch.

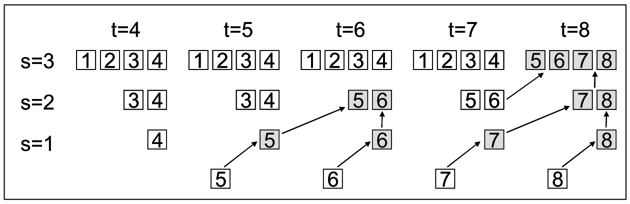


Fig. 4. Approximate updating of $\hat{N}^{(s)}_{t,z,w}$ from $t = 4$ to $t = 8$ with $S = 3$.

ALGORITHM 2: Algorithm for the approximate updating of $\hat{N}^{(s)}_{t,z,w}$ using a buffer. $\hat{N}^{\text{buffer}}_{t,z,w}$ represents the buffer value.

```
$\hat{N}^{\text{buffer}}_{t,z,w} \leftarrow \hat{N}^{\text{buffer}}_{t-1,z,w} + \hat{N}_{t,z,w}$
**if** $(t-1) \bmod L_s = 0$ **then**
    $\hat{N}^{(s)}_{t,z,w} \leftarrow \hat{N}^{\text{buffer}}_{t,z,w}$
    $\hat{N}^{\text{buffer}}_{t,z,w} \leftarrow \hat{N}_{t,z,w}$
**end**
```

Since the dynamics of a word distribution for a long-timescale are considered to be slower than that for a short-timescale, this approximation—decreasing the update frequency for long-timescale distributions—is reasonable. Figure 3 shows a graphical model representation of online inference in MDTM.

For the Dirichlet prior parameter of the word distribution, we use the weighted sum of the multiscale word distributions, as in (2). The parameter can be rewritten as the weighted sum of the word distributions for each epoch as follows.

$$\sum_{s=1}^{S} \lambda_{t,z,s}\xi^{(s)}_{t-1,z,w} = \sum_{t'=t-2^{S-1}}^{t-1} \lambda'_{t,z,t'}\hat{\phi}_{t',z,w}, \tag{16}$$

where

$$\hat{\phi}_{t,z,w} = \frac{\hat{N}_{t,z,w}}{\sum_{w'}\hat{N}_{t,z,w'}}, \tag{17}$$

is the expected probability that word w appears in topic z at epoch t, and

$$\lambda'_{t,z,t'} = \sum_{s=\lceil \log_2(t-t')+1\rceil}^{S} \frac{\lambda_{t,z,s}\sum_{w}\hat{N}_{t',z,w}}{\sum_{w}\sum_{t''=t-2^{s-1}}^{t-1}\hat{N}_{t'',z,w}}, \tag{18}$$

is its weight. See Appendix D for the derivation. Therefore, the multiscale dynamic topic model can be seen as an approximation of a model that depends on the word distributions for each of the previous epochs. By considering multiscale word distributions, the number of weight parameters $\mathbf{\Lambda}_t$ can be reduced from $O(2^S Z)$ to $O(SZ)$, and this leads to more robust inference. Furthermore, the use of multiscaling also reduces the memory requirement from $O(2^S Z W)$ to $O(SZW)$ as previously described.

3.5 Enhanced Methods for Approximate Updating

Approximate updating method for arbitrary scale. In the preceding discussion, we set scales so that $\boldsymbol{\xi}^{(s)}_{t,z}$ indicates the word distribution from $t - 2^{s-1} + 1$ to t. However, we can set scales arbitrarily. If we know that the given data exhibit periodicity, e.g., of one

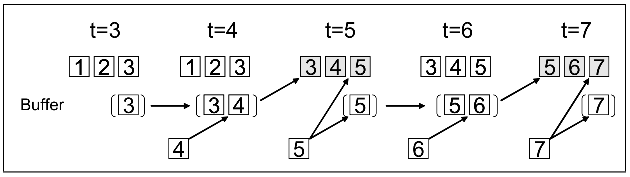


Fig. 5. Approximate updating of $\hat{N}^{(s)}_{t,z,w}$ with scale size $L_s = 3$.

ALGORITHM 3: Algorithm for the approximate updating of $\hat{N}^{(s)}_{t,z,w}$ using B buffers. $\hat{N}^{b}_{t,z,w}$ represents the bth buffer value, t_b represents the timing with which the bth buffer is updated, and $0 \leq t_b < L_s$.

```
for $b = 1, \cdots, B$ do
    $\hat{N}^{b}_{t,z,w} \leftarrow \hat{N}^{b}_{t-1,z,w} + \hat{N}_{t,z,w}$
    if $(t - t_b) \bmod L_s = 0$ then
        $\hat{N}^{(s)}_{t,z,w} \leftarrow \hat{N}^{b}_{t,z,w}$
        $\hat{N}^{b}_{t,z,w} \leftarrow \hat{N}_{t,z,w}$
    end
end
```

week and one month, we can use the scale of one week for $s = 1$ and one month for $s = 2$. In these cases, we can also efficiently update values by using buffers for storing intermediate values. Algorithm 2 shows the updating algorithm with scale size L_s using a buffer.

LEMMA 3.3. *The proposed updating method (arbitrary scale) requires $O(SZW)$ memory for updating multiscale word distributions.*

PROOF. The proposed method for an arbitrary scale can update $\hat{N}^{(s)}_{t,z,w}$ using a single buffer. We need a buffer for each of S scales, each of Z topics, and each of W words. Thus, it would take $O(SZW)$ space. □

While this approximation needs a buffer, the memory requirement of this method is still $O(SZW)$, which is also linear with respect to the number of scales. Figure 5 shows the approximate updating of $\hat{N}^{(s)}_{t,z,w}$ with scale size $L_s = 3$. At epoch $t = 3$, the newly calculated value $\hat{N}_{t=3,z,w}$ is stored in the buffer. At epoch $t = 4$, the buffer value is updated by adding the newly calculated value $\hat{N}_{t=4,z,w}$. At epoch $t = 5$, value $\hat{N}^{(s)}_{t,z,w}$ is updated by using the newly calculated value and the buffer value at the previous epoch. The buffer is cleared and the newly calculated value $\hat{N}_{t=5,z,w}$ is stored in the buffer. In this way, $\hat{N}^{(s)}_{t,z,w}$ is updated every $L_s - 1$ epochs by using the value at the previous epoch.

Improving approximation by using multiple buffers. We can improve the approximation by increasing the update frequency. By using B buffers, the value is updated B times every $L_s - 1$ epochs, and the rate of unapproximated values is $\frac{B}{L_s - 1}$. Algorithm 3 shows the updating algorithm with scale size L_s using B buffers.

LEMMA 3.4. *The proposed updating method (arbitrary frequency) requires $O(BSZW)$ memory for updating multiscale word distributions.*

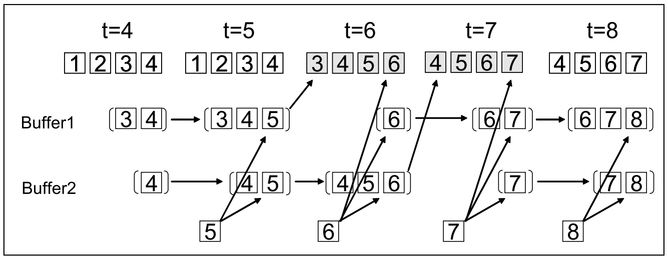


Fig. 6. Approximate updating of $\hat{N}^{(s)}_{t,z,w}$ with scale size $L_s = 4$ using two buffers.

Table II.

Average perplexities over epochs. The value in the parenthesis represents the standard deviation over datasets.

	MDTM	DTM	LDAall	LDAone	LDAonline
NIPS	**1754.9** (41.3)	1771.6 (37.2)	1802.4 (36.4)	1822.0 (44.0)	1769.8 (41.5)
PNAS	**2964.3** (122.0)	3105.7 (146.8)	3262.9 (159.7)	5221.5 (268.7)	3401.7 (149.1)
Digg	**3388.9** (37.7)	3594.2 (46.4)	3652.6 (27.1)	5162.9 (43.4)	3500.0 (43.6)
Addresses	**1968.8** (56.5)	2105.2 (49.7)	2217.2 (75.3)	3033.5 (70.9)	2251.6 (62.0)

PROOF. The proposed method for an arbitrary frequency can update $\hat{N}^{(s)}_{t,z,w}$ using the values in B buffers. We need B buffers for each of S scales, each of Z topics, and each of W words. Thus, it would take $O(BSZW)$ space. □

By using more buffers, we can approximate the values more closely. This method allows users to choose the update frequency. Figure 6 shows approximate updating with two buffers and scale size $L_s = 4$. The value at each buffer is updated by adding the newly calculated value. When the size of the buffer becomes the scale size $L_s - 1$, the value is used for calculating $\hat{N}^{(s)}_{t,z,w}$, and the buffer is updated with the value $\hat{N}_{t,z,w}$. For example, at epoch $t = 5$, the first buffer is updated by adding $\hat{N}_{t=5,z,w}$. At epoch $t = 6$, it is used for calculating $\hat{N}^{(s)}_{t=6,z,w}$ by adding it to $\hat{N}_{t=6,z,w}$, and the buffer is updated with $\hat{N}_{t=6,z,w}$.

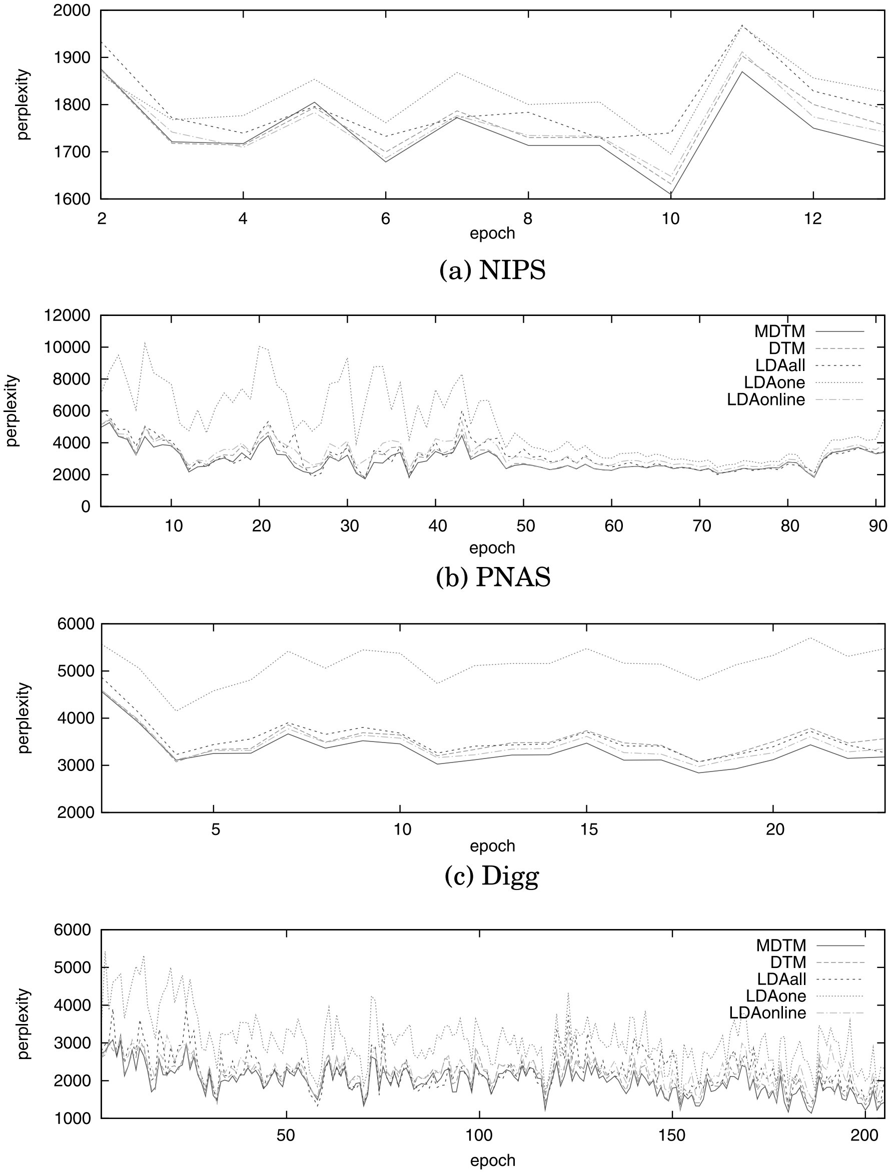
4. EXPERIMENTS

4.1 Setting

We evaluated the multiscale dynamic topic model with online inference (MDTM) using four real document collections with timestamps: NIPS, PNAS, Digg, and Addresses.

The NIPS data consist of papers from Neural Information Processing Systems (NIPS) conferences from 1987 to 1999. There were 1740 documents, and the vocabulary size was 14,036. The unit epoch was set at one year, so there were 13 epochs. The PNAS data consist of the titles of papers that have appeared in the Proceedings of the National Academy of Sciences from 1915 to 2005. There were 79,477 documents, and the vocabulary size was 20,534. The unit epoch was set at one year, so there were 91 epochs. The Digg data consist of blog posts that have appeared on the social news Web site Digg[1] from January 29th to February 20th 2009. There were 108,356 documents, and the vocabulary size was 23,494. The unit epoch was set at one day, so there were 23 epochs. The Addresses data consist of the State of the Union addresses from

[1] http://digg.com



(a) NIPS

(b) PNAS

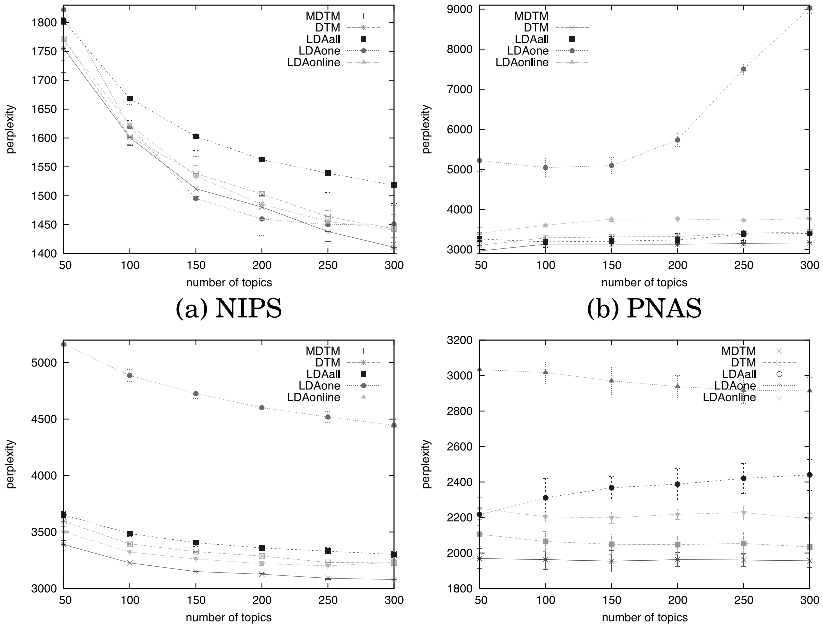
(c) Digg

(d) Addresses

Fig. 7. Perplexities for each epoch.

1790 to 2002. We increased the number of documents by splitting each transcript into 3-paragraph documents as done in Wang and McCallum [2006]. We omitted words that occurred in fewer than 10 documents. There were 6413 documents, and the vocabulary size was 6759. The unit epoch was set at one year, and excluding the years for which data were missing, there were 205 epochs. We omitted stop-words from all data sets.

We compared MDTM with DTM, LDAall, LDAone, and LDAonline. DTM is a dynamic topic model with online inference, which does not take multiscale distributions



(a) NIPS (b) PNAS

(c) Digg (d) Addresses

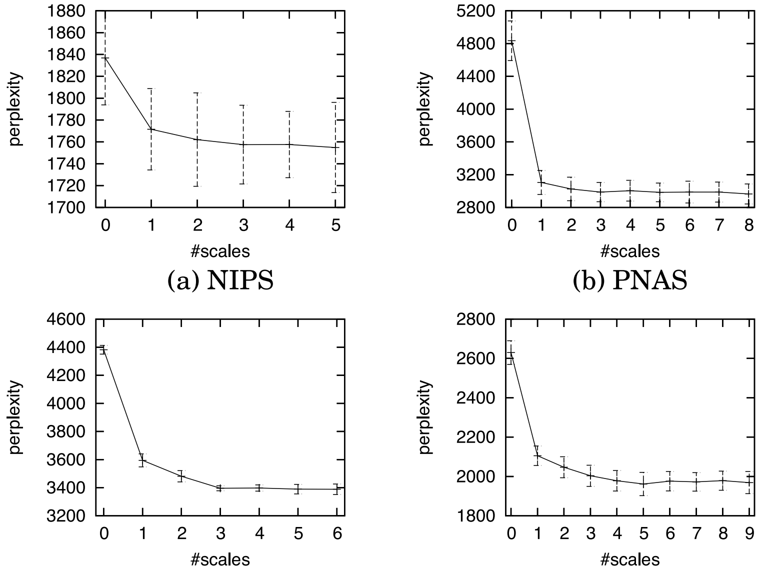
Fig. 8. Average perplexities with different numbers of topics and their standard deviations.

into consideration; it corresponds to MDTM with $S = 1$. Note that the DTM used here models dynamics with Dirichlet priors while the original DTM used Gaussian priors. LDAall, LDAone, and LDAonline are based on LDA, and therefore do not model the dynamics. LDAall is an LDA model, which uses all past data for inference. LDAone is an LDA model, which uses just the current data for inference. LDAonline is an online learning extension of LDA, in which the parameters are estimated using those of the previous epoch and the new data [Banerjee and Basu 2007]. The time complexity of Gibbs sampling for LDAone and LDAonline are $O(D_t N_t Z)$, which is the same as that for the proposed model. That for LDAall is $O(D_{1,t} N_t Z)$, where $D_{1,t} = \sum_{\tau=1}^{t} D_\tau$. For a fair comparison, the hyperparameters in these LDAs were optimized using stochastic EM. We set the number of latent topics at $Z = 50$ for all models, and we iterated an E- and M-step 500 times for each epoch for all models. In MDTM, we used $\gamma = 1$, and we estimated the Dirichlet prior for topic proportions subject to $\alpha_{t,z} \geq 10^{-2}$ to avoid overfitting. We set the scale size at $L_s = 2^{s-1}$, and the number of scales so that one of the multiscale distributions covered the entire period, or $S = \lceil \log_2 T + 1 \rceil$, where T is the number of epochs. We used Algorithm 1 for updating the parameters.

We evaluated the predictive performance of each model using the perplexity of held-out word.

$$\text{Perplexity} = \exp\left(- \frac{\sum_d \sum_{n=1}^{N_{t,d}^{\text{test}}} \log P\left(w_{t,d,n}^{\text{test}} | t, d, \mathcal{D}_t\right)}{\sum_d N_{t,d}^{\text{test}}} \right), \tag{19}$$

where $N_{t,d}^{\text{test}}$ is the number of held-out words in the dth document at epoch t, $w_{t,d,n}^{\text{test}}$ is the nth held-out words in the document, and $\mathcal{D}_t$ represents training samples until epoch



(a) NIPS

(b) PNAS

(c) Digg

(d) Addresses

Fig. 9. Average perplexities of MDTM with different numbers of scales and their standard deviations.

t. A lower perplexity represents higher predictive performance. The word probability can be calculated as follows,

$$P(w|t, d, \mathcal{D}_t) = \sum_z \hat{\theta}_{t,d,z}\hat{\phi}_{t,z,w}, \tag{20}$$

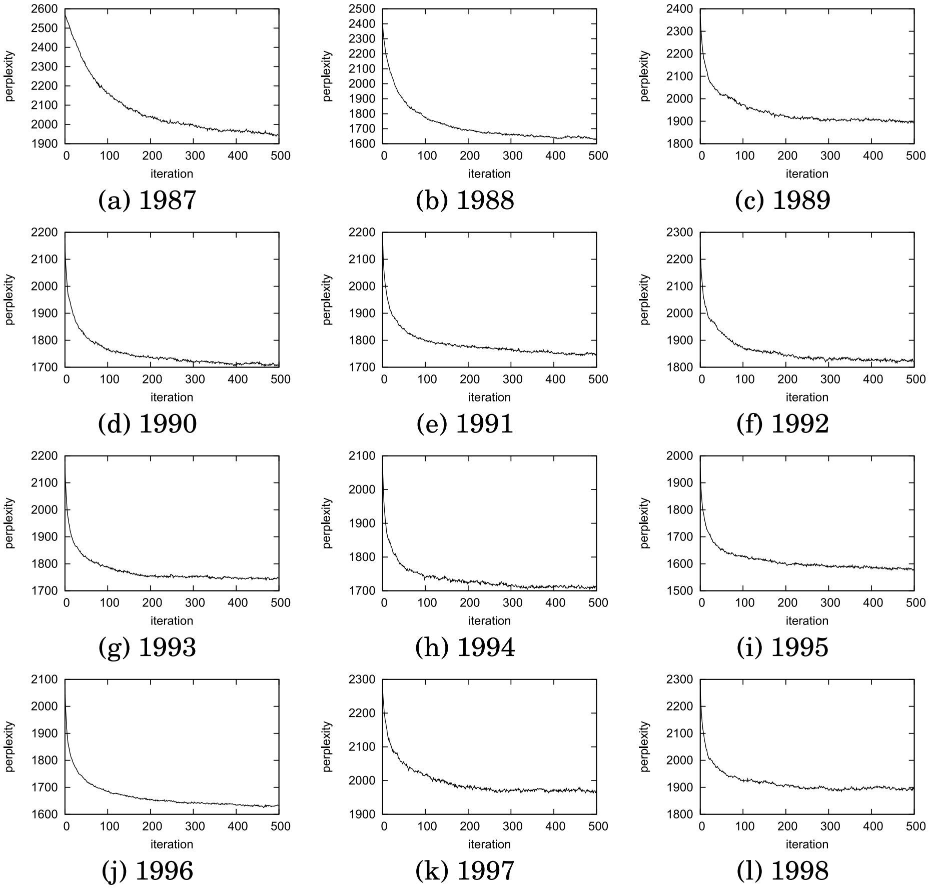
where

$$\hat{\theta}_{t,d,z} = \frac{N_{t,d,z} + \alpha_{t,z}}{N_{t,d} + \sum_{z'} \alpha_{t,z'}}. \tag{21}$$

We used half of the words in 10% of the documents as held-out words for each epoch, and used the other words as training samples. We created ten sets of training and test data by random sampling, and evaluated the average perplexity over the ten data sets.

4.2 Results

The average perplexities over the epochs are shown in Table II, and the perplexities for each epoch are shown in Figure 7. For all data sets, MDTM achieved the lowest perplexity, which implies that MDTM can appropriately model the dynamics of various types of data sets through its use of multiscale properties. DTM had a higher perplexity than MDTM because it could not model the long-timescale dependencies. LDAall and LDAonline have high perplexities because they do not consider the dynamics. The perplexity achieved by LDAone is high because it uses only current data and ignores past information. The average perplexities over epochs with different numbers of topics are shown in Figure 8. Under the same number of topics, MDTM achieved the lowest perplexities in every case except when $Z = 150$ and 200 in the NIPS data, and achieved the second lowest perplexities when $Z = 150$ and 200 in the NIPS data. LDAone achieved the lowest perplexity with $Z = 150$ and 200 in the NIPS data because the number of words in each document in the NIPS data is large compared with other data, and the model can be learned adequately by using data at each epoch. The



(a) 1987 (b) 1988 (c) 1989

(d) 1990 (e) 1991 (f) 1992

(g) 1993 (h) 1994 (i) 1995

(j) 1996 (k) 1997 (l) 1998

Fig. 10. Average perplexity of MDTM over iterations on the inference for each epoch in NIPS data.

average perplexities over epochs with different numbers of scales in MDTM are shown in Figure 9. Note that $s = 0$ uses only the uniform distribution, while $s = 1$ uses the uniform distribution and the previous epoch's distribution. The perplexities decreased as the number of scales increased. This result indicates the importance of considering multiscale distributions. Although the number of parameters increases as the number of scales increases, overfitting did not occur. This result suggests that we can use as many scales as we wish as long as we can bear the computational and memory costs.

Figure 10 shows the perplexities of the proposed model over iterations on inference in NIPS data, which are averaged over ten different initializations. As the number of iterations increases, the perplexity decreases, and eventually converges to a certain point.

Figure 11 shows the average computational time per epoch when using a computer with a Xeon5355 2.66GHz CPU and a 16GB memory. The computational time for MDTM is roughly linear against the number of scales. Even though MDTM considers multiple timescale distributions, its computational time is much smaller than that of LDAall, which considers a single timescale distribution. This is because MDTM only uses current samples for inference. In contrast, LDAall uses all the samples for inference.

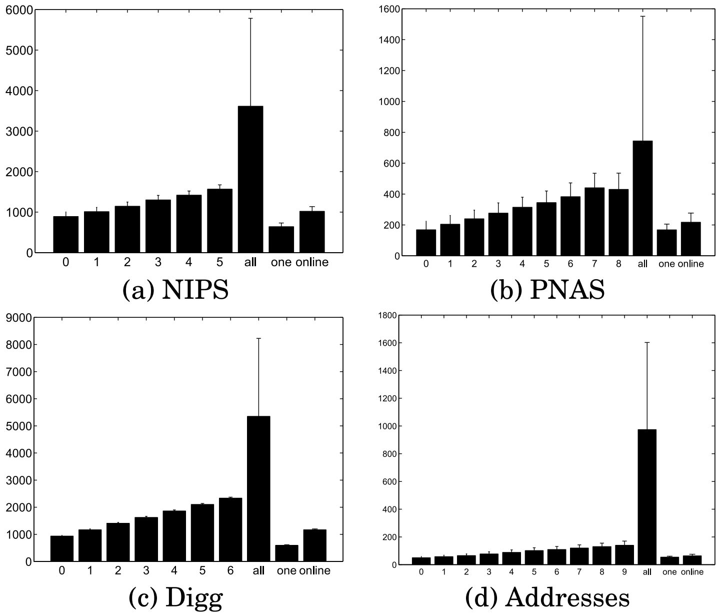


Fig. 11. Average computational time (sec) of MDTM per epoch with different numbers of scales, LDAall, LDAone, and LDAonline and their standard deviations.

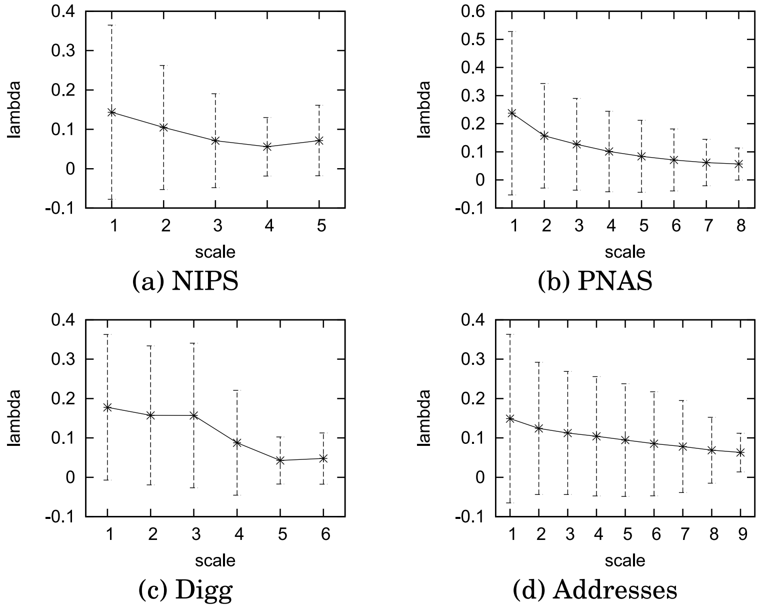
When the data in an epoch do not depend on data in other epochs, LDAone is an appropriate model. When the data for all epochs are generated from a distribution, or the data have no dynamics, LDAall and LDAonline are both appropriate models. MDTM can flexibly model data that have dynamics, where data in consecutive epochs exhibit some dependencies. In particular, MDTM with $\lambda_{t,z,s} = 0$ for $s > 0$ corresponds to LDAone, and MDTM with $\lambda_{t,z,s} = 0$ for all s except when the timescale is infinite, corresponds to LDAonline.

Figure 12 shows the average estimated $\lambda_{t,z,s}$ with different numbers of scales s in MDTM. The sum of the values for each epoch and for each topic are normalized to one. The parameters decrease as the timescale lengthens. This result implies that recent distributions are more informative as regards estimating current distributions, which is intuitively reasonable.

In Section 3.5, we described an algorithm for improving the approximation of an estimation by using multiple buffers. We experimentally investigated how the number of buffers affects the predictive performance. Figure 13 shows the average perplexities achieved by MDTM when the number of topics $Z = 50$, the number of scales $S = 1$, and the scale size $L_s = 10$. As the number of buffers increased, the perplexity decreased, although the memory requirement increased. This result suggests that we can improve predictive performance by using as many buffers as possible.

4.3 Application of MDTM

MDTM can extract the topic evolution in multiple timescales. For experiments in this section, we used MDTM, where the number of topics $Z = 50$ and the number of scales $S = 4$. Figure 14 shows two topic examples of the multiscale topic evolution in NIPS data analyzed by MDTM. Note that we omit words that appeared in the longer timescales from the table. In the longest timescale, basic words for the research



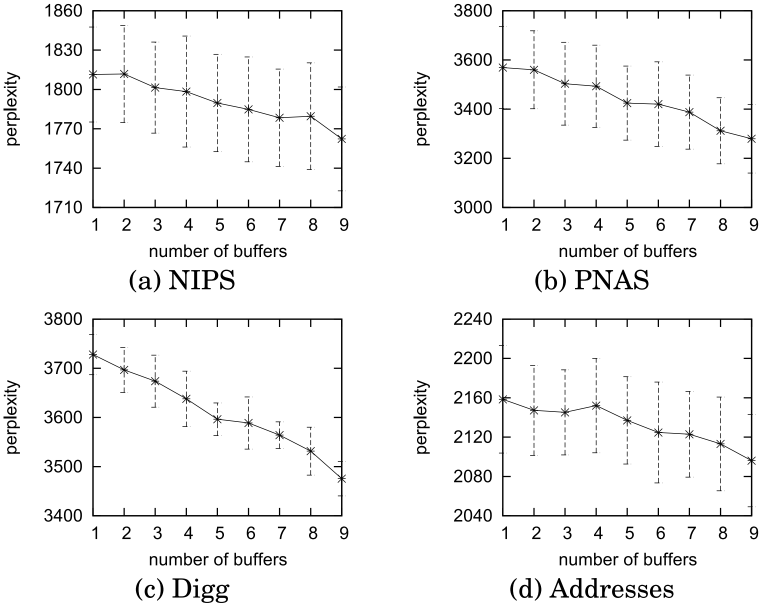
(a) NIPS

(b) PNAS

(c) Digg

(d) Addresses

Fig. 12. Average normalized weights λ with different scales estimated in MDTM and their standard deviations.



(a) NIPS

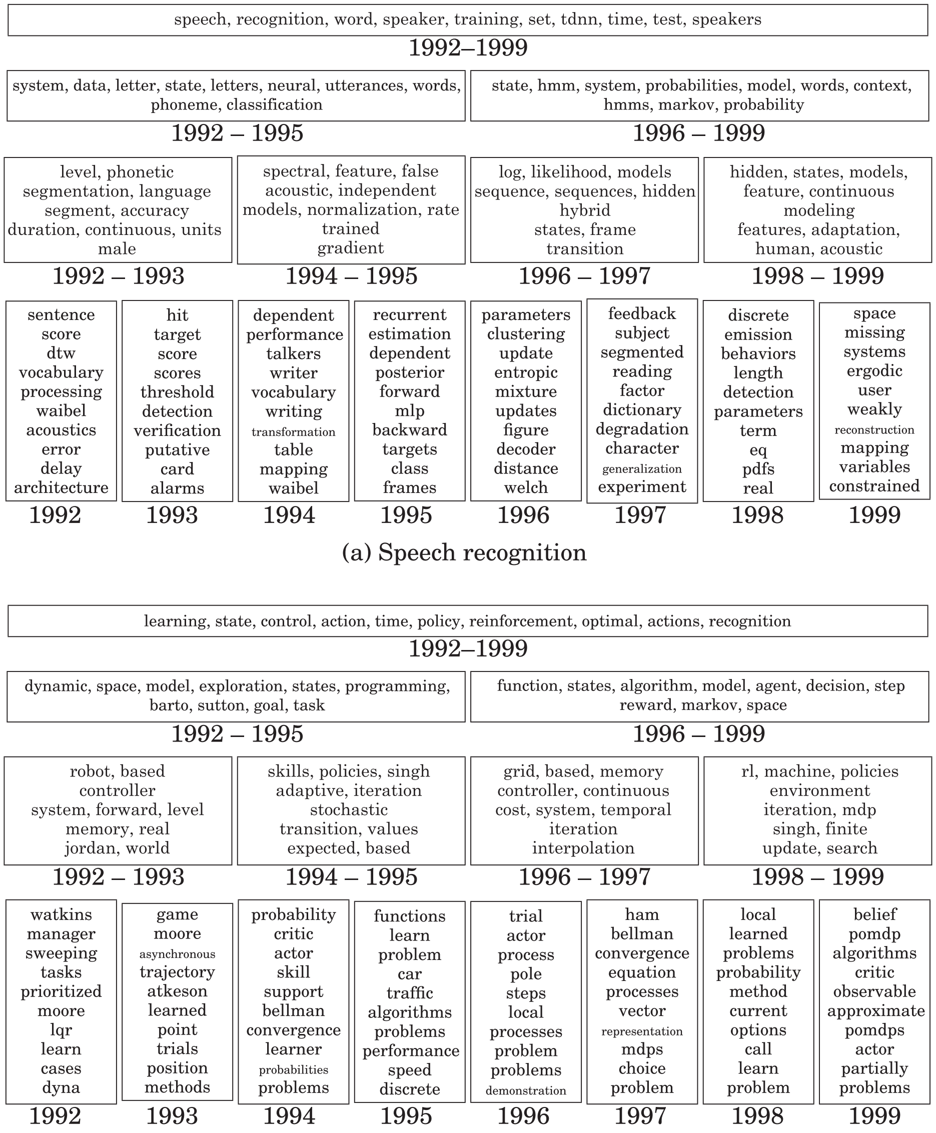
(b) PNAS

(c) Digg

(d) Addresses

Fig. 13. Average perplexities with different numbers of buffers in MDTM with multiple buffers and their standard deviations.

field are appropriately extracted, such as “speech,” “recognition,” and “speaker” in the speech recognition topic, “control,” “action,” “policy,” and “reinforcement” in the reinforcement learning topic. In the shorter timescale, we can see the evolution of trends in the research. For example, in speech recognition research, phoneme classification


speech, recognition, word, speaker, training, set, tdnn, time, test, speakers
1992–1999
system, data, letter, state, letters, neural, utterances, words, phoneme, classification
1992 – 1995
state, hmm, system, probabilities, model, words, context, hmms, markov, probability
1996 – 1999
level, phonetic segmentation, language segment, accuracy duration, continuous, units male
1992 – 1993
spectral, feature, false acoustic, independent models, normalization, rate trained gradient
1994 – 1995
log, likelihood, models sequence, sequences, hidden hybrid states, frame transition
1996 – 1997
hidden, states, models, feature, continuous modeling features, adaptation, human, acoustic
1998 – 1999
sentence score dtw vocabulary processing waibel acoustics error delay architecture
1992
hit target score scores threshold detection verification putative card alarms
1993
dependent performance talkers writer vocabulary writing transformation table mapping waibel
1994
recurrent estimation dependent posterior forward mlp backward targets class frames
1995
parameters clustering update entropic mixture updates figure decoder distance welch
1996
feedback subject segmented reading factor dictionary degradation character generalization experiment
1997
discrete emission behaviors length detection parameters term eq pdfs real
1998
space missing systems ergodic user weakly reconstruction mapping variables constrained
1999

(a) Speech recognition

learning, state, control, action, time, policy, reinforcement, optimal, actions, recognition
1992–1999
dynamic, space, model, exploration, states, programming, barto, sutton, goal, task
1992 – 1995
function, states, algorithm, model, agent, decision, step reward, markov, space
1996 – 1999
robot, based controller system, forward, level memory, real jordan, world
1992 – 1993
skills, policies, singh adaptive, iteration stochastic transition, values expected, based
1994 – 1995
grid, based, memory controller, continuous cost, system, temporal iteration interpolation
1996 – 1997
rl, machine, policies environment iteration, mdp singh, finite update, search
1998 – 1999
watkins manager sweeping tasks prioritized moore lqr learn cases dyna
1992
game moore asynchronous trajectory atkeson learned point trials position methods
1993
probability critic actor skill support bellman convergence learner probabilities problems
1994
functions learn problem car traffic algorithms problems performance speed discrete
1995
trial actor process pole steps local processes problem problems demonstration
1996
ham bellman convergence equation processes vector representation mdps choice problem
1997
local learned problems probability method current options call learn problem
1998
belief pomdp algorithms critic observable approximate pomdps actor partially problems
1999

(b) Reinforcement learning

Fig. 14. Two topic examples of the multiscale topic evolution in NIPS data analyzed by MDTM: (a) speech recognition, and (b) reinforcement learning topics. The ten most probable words for each epoch, timescale, and topic are shown.

was a common task until 1990, and since 1991 probabilistic approaches such as hidden Markov models (HMM) have been frequently used.

MDTM can also be used for tracking the popularity of each topic. Figure 15 shows the estimated $\alpha_{t,z}$ of four topics from the NIPS data analyzed by MDTM, which represents the popularity dynamics for each topic. The most probable words on the long-timescale for topics “neural network” and “probabilistic model” are shown in Table III.

Table III. Two Topic Examples in NIPS Data Analyzed by MDTM

neural network	layer, net, neural, nets, system, figure, activation, signal, error, noise, backpropagation
probabilistic model	data, model, training, error, probability, parameters, set, distribution, models, Gaussian, variance

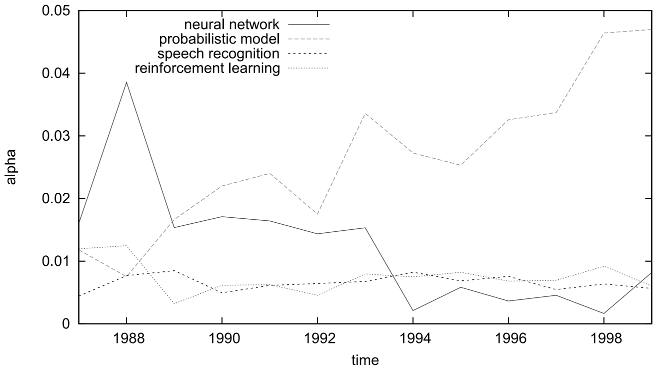


Fig. 15. Popularity dynamics of four topics in NIPS data analyzed by MDTM.

We can see that the popularity of the topics "neural network" has decreased over time, on the other hand, that of the "probabilistic model" has increased.

The estimated parameters $\lambda'_{t,z,t-1}$ in (18) represent the dependency of the word distribution of topic z between t and $t-1$. Therefore, we can analyze the magnitude of the change in the topic by using $\lambda'_{t,z,t-1}$. Usually, the change in a topic can be calculated by using the KL divergence between the current word distribution and the previous word distribution $\mathrm{KL}(\boldsymbol{\phi}_{t-1,z}||\boldsymbol{\phi}_{t,z})$. The correlation coefficients between $1/\log(\lambda'_{t,z,t-1})$ and the KL divergences were 0.625, 0.216, 0.729, and 0.152 for NIPS, PNAS, Digg, and Addresses data sets respectively, which were calculated using all time intervals and all topics. This result shows that we can analyze the magnitude of changes by using $\lambda'_{t,z,t-1}$ without calculating the KL divergence.

Figure 16 shows the estimated $1/\log(\lambda_{t,z,t-1})$ of four topics in the NIPS data analyzed by MDTM, where this value represents how the estimated topics change compared with those in the previous epoch. In general, the values in late epochs are low, since the inference of topics becomes stable when we use more data for the inference. The value for a "neural network" in 1995 was high because it was at the end of the neural network boom.

5. CONCLUSION

In this article, we have proposed a topic model with multiscale dynamics and efficient online inference procedures. We have experimentally confirmed that the proposed method can appropriately model the dynamics in document data by considering multiscale properties, and that it is computationally efficient.

We assumed that the number of topics was known and fixed over time. We can automatically infer the number of topics by extending the model to a nonparametric Bayesian model such as the Dirichlet process mixture model [Ren et al. 2008; Teh et al. 2006]. In future work, we plan to determine the unit time interval, the length of scale, and the number of scales, automatically from the given data. The problem of the unit time interval can be solved by using continuous time dynamic topic models [Wang

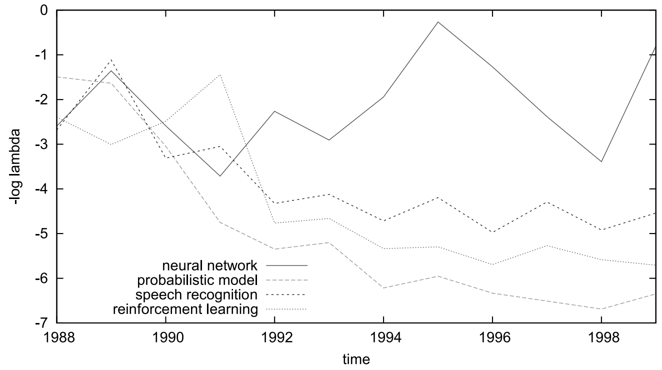


Fig. 16. Topic change dynamics of four topics in NIPS data analyzed by MDTM.

et al. 2008]. The length of scale and the number of scales can be determined using nonparametric Bayesian methods. The length of scale can be continuous as in [Wang and McCallum 2006], which treats time dependence as a continuous beta random variable. The proposed model can be extended to find influential documents by modeling the influence for each document as described in Gerrish and Blei [2010]. Since the proposed method is applicable to various kinds of discrete data with timestamps, such as Web access logs, blogs, and e-mail, we will evaluate the proposed method further by applying it to other data sets.

APPENDIX

A. DERIVATION OF (9)

In this appendix, we give the derivation of (9).

$$\begin{aligned} &P(z_j = k|\boldsymbol{W}_t, \boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \\ &\propto P(z_j = k, w_j|\boldsymbol{W}_{t\backslash j}, \boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \\ &= P(z_j = k|\boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t) P(w_j|\boldsymbol{W}_{t\backslash j}, z_j = k, \boldsymbol{Z}_{t\backslash j}, \Xi_{t-1}, \boldsymbol{\Lambda}_t). \end{aligned} \tag{22}$$

The first factor of (22) becomes,

$$\begin{aligned} &P(z_j = k|\boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t) \\ &= \int P(z_j = k|\boldsymbol{\theta}_{t,d}) P(\boldsymbol{\theta}_{t,d}|\boldsymbol{Z}_{t\backslash j}, \boldsymbol{\alpha}_t) d\boldsymbol{\theta}_{t,d} \\ &= \int \theta_{t,d,k} \frac{\Gamma\left(\sum_z \left[N_{t,d,z\backslash j} + \alpha_{t,z}\right]\right)}{\prod_z \Gamma\left(N_{t,d,z\backslash j} + \alpha_{t,z}\right)} \prod_z \theta_{t,d,z}^{N_{t,d,z\backslash j}+\alpha_z-1} d\boldsymbol{\theta}_{t,d} \\ &= \frac{\Gamma\left(\sum_z \left[N_{t,d,z\backslash j} + \alpha_{t,z}\right]\right)}{\prod_z \Gamma\left(N_{t,d,z\backslash j} + \alpha_{t,z}\right)} \int \prod_{z \neq k} \theta_{t,d,z}^{N_{t,d,z\backslash j}+\alpha_z-1} \theta_{t,d,k}^{N_{t,d,k\backslash j}+\alpha_k} d\boldsymbol{\theta}_{t,d} \\ &= \frac{\Gamma\left(\sum_z [N_{t,d,z\backslash j} + \alpha_{t,z}]\right)}{\prod_z \Gamma\left(N_{t,d,z\backslash j} + \alpha_{t,z}\right)} \frac{\prod_{z\neq k} \Gamma\left(N_{t,d,z\backslash j} + \alpha_{t,z}\right) \Gamma\left(N_{t,d,k\backslash j} + \alpha_{t,k} + 1\right)}{\Gamma\left(\sum_z \left[N_{t,d,z\backslash j} + \alpha_{t,z}\right] + 1\right)} \\ &= \frac{N_{t,d,k\backslash j} + \alpha_{t,k}}{N_{t,d\backslash j} + \sum_z \alpha_{t,z}}, \end{aligned} \tag{23}$$

where we used $\int \prod_z \theta_z^{\alpha_z - 1} d\boldsymbol{\theta} = \frac{\prod_z \Gamma(\alpha_z)}{\Gamma(\sum_z \alpha_z)}$ in the fourth equation, which is the normalizing constant of the Dirichlet distribution, and $\Gamma(x+1) = x\Gamma(x)$ in the fifth equation. In a similar way, the second factor of (22) becomes,

$$
\begin{aligned}
& P(w_j | \mathbf{W}_{t\backslash j}, z_j = k, \mathbf{Z}_{t\backslash j}, \boldsymbol{\Xi}_{t-1}, \boldsymbol{\Lambda}_t) \\
&= \int P(w_j | z_j = k, \boldsymbol{\phi}_{t,k}) P(\boldsymbol{\phi}_{t,k} | \mathbf{W}_{t\backslash j}, \mathbf{Z}_{t\backslash j}, \boldsymbol{\Xi}_{t-1}, \boldsymbol{\Lambda}_t) d\boldsymbol{\phi}_{t,k} \\
&= \int \phi_{t,k,w_j} \frac{\Gamma(\sum_w [N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}])}{\prod_w \Gamma(N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w})} \prod_w \phi_{t,k,w}^{N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w} - 1} d\boldsymbol{\phi}_{t,k} \\
&= \frac{\Gamma\left(\sum_w \left[N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right]\right)}{\prod_w \Gamma\left(N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right)} \\
&\quad \times \int \prod_{w \neq w_j} \phi_{t,k,w}^{N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w} - 1} \phi_{t,k,w_j}^{N_{t,k,w_j\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w_j}} d\boldsymbol{\phi}_{t,k} \\
&= \frac{\Gamma\left(\sum_w \left[N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right]\right)}{\prod_w \Gamma\left(N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right)} \\
&\quad \times \frac{\prod_{w \neq w_j} \Gamma\left(N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right) \Gamma\left(N_{t,k,w_j\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w_j} + 1\right)}{\Gamma\left(\sum_w \left[N_{t,k,w\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w}\right] + 1\right)} \\
&= \frac{N_{t,k,w_j\backslash j} + \sum_s \lambda_{t,k,s}\xi^{(s)}_{t-1,k,w_j}}{N_{t,k\backslash j} + \sum_s \lambda_{t,k,s}}. \qquad (24)
\end{aligned}
$$

B. DERIVATION OF (10)

In this appendix, we give the derivation of (10). Consider the following function.

$$f(x) = \log \frac{\Gamma(x)}{\Gamma(n+x)}, \qquad (25)$$

where $x > 0$ and $n > 0$. The tangent of $f(x)$ at $\hat{x}$ is as follows.

$$g(x) = \log \frac{\Gamma(\hat{x})}{\Gamma(n+\hat{x})} + \left(\Psi(\hat{x}) - \Psi(n+\hat{x})\right)(x - \hat{x}). \qquad (26)$$

Since $f(x)$ is convex downward, the tangent is the lower bound, $f(x) \geq g(x)$, and we get the following inequality.

$$\log \frac{\Gamma(x)}{\Gamma(n+x)} \geq \log \frac{\Gamma(\hat{x})}{\Gamma(n+\hat{x})} + \left(\Psi(\hat{x}) - \Psi(n+\hat{x})\right)(x - \hat{x}). \qquad (27)$$

Additionally, consider the following function.

$$
\begin{aligned}
h(x) &= \log \frac{\Gamma(n+x)}{\Gamma(x)} - \log \frac{\Gamma(n+\hat{x})}{\Gamma(\hat{x})} - \hat{x}[\Psi(n+\hat{x}) - \Psi(\hat{x})] \log \frac{x}{\hat{x}} \\
&= \sum_{m=0}^{n-1} \left[\log(m+x) - \log(m+\hat{x}) - \frac{\hat{x}}{m+\hat{x}} \log \frac{x}{\hat{x}} \right], \qquad (28)
\end{aligned}
$$

where $x > 0$, $\hat{x} > 0$, $n > 0$, and we used $\log \frac{\Gamma(n+x)}{\Gamma(x)} = \sum_{m=0}^{n-1} \log(m+x)$ and $\Psi(n+x) - \Psi(x) = \sum_{m=0}^{n-1} \frac{1}{m+x}$ in the second equality. The value in each summation is nonnegative.

$$k(x) = \log(m+x) - \log(m+\hat{x}) - \frac{\hat{x}}{m+\hat{x}} \log \frac{x}{\hat{x}} \geq 0, \tag{29}$$

because

$$\frac{\partial k(x)}{\partial x} = \frac{m(x-\hat{x})}{(m+x)x(m+\hat{x})}, \tag{30}$$

and therefore $k(x)$ monotonically decreases (increases) when $x < \hat{x}$ ($x > \hat{x}$), and $k(\hat{x}) = 0$. By using (28) and (29), we can obtain the following inequality.

$$\log \frac{\Gamma(n+x)}{\Gamma(x)} \geq \log \frac{\Gamma(n+\hat{x})}{\Gamma(\hat{x})} + \hat{x}[\Psi(n+\hat{x}) - \Psi(\hat{x})] \log \frac{x}{\hat{x}}. \tag{31}$$

The joint log likelihood can be written as follows, using (4), (5), (6), and (7).

$$\begin{aligned}
&\log P(\boldsymbol{W}_t, \boldsymbol{Z}_t, \boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{t-1}, \gamma, \boldsymbol{\Xi}_{t-1}, \boldsymbol{\Lambda}_t) \\
&= \log P(\boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{t-1}, \gamma) + \log P(\boldsymbol{Z}_t | \boldsymbol{\alpha}_t) + C \\
&= (\gamma \alpha_{t-1,z} - 1) \log \alpha_{t,z} - \gamma \alpha_{t,z} \\
&\quad + \sum_d \left[\log \Gamma \left(\sum_z \alpha_{t,z} \right) - \log \Gamma \left(N_{t,d} + \sum_z \alpha_{t,z} \right) \right] \\
&\quad + \sum_d \left[\log \Gamma \left(N_{t,d,z} + \alpha_{t,z} \right) - \log \Gamma \left(\alpha_{t,z} \right) \right] + C',
\end{aligned} \tag{32}$$

where C and C' are the constants that do not depend on $\alpha_{t,z}$. The second term of (32) satisfies the following inequality.

$$\begin{aligned}
&\sum_d \left[\log \Gamma (\sum_z \alpha_{t,z}) - \log \Gamma \left(N_{t,d} + \sum_z \alpha_{t,z} \right) \right] \\
&\geq \sum_d \left[\log \Gamma \left(\sum_z \alpha_{t,z}^{\text{old}} \right) - \log \Gamma \left(N_{t,d} + \sum_z \alpha_{t,z}^{\text{old}} \right) \right] \\
&\quad + \sum_d \left(\left[\Psi \left(N_{t,d} + \sum_z \alpha_{t,z}^{\text{old}} \right) - \Psi \left(\sum_z \alpha_{t,z}^{\text{old}} \right) \right] \sum_z \left(\alpha_{t,z}^{\text{old}} - \alpha_{t,z} \right) \right),
\end{aligned} \tag{33}$$

where we used (27) by letting $x = \sum_z \alpha_{t,z}$, $n = N_{t,z}$ and $\hat{x} = \sum_z \alpha_{t,z}^{\text{old}}$. The third term of (32) satisfies the following inequality.

$$\begin{aligned}
&\sum_d [\log \Gamma(N_{t,d,z} + \alpha_{t,z}) - \log \Gamma(\alpha_{t,z})] \\
&\geq \sum_d \left(\log \Gamma \left(N_{t,d,z} + \alpha_{t,z}^{\text{old}} \right) - \log \Gamma \left(\alpha_{t,z}^{\text{old}} \right) \right. \\
&\quad \left. + \left[\Psi \left(N_{t,d,z} + \alpha_{t,z}^{\text{old}} \right) - \Psi \left(\alpha_{t,z}^{\text{old}} \right) \right] \alpha_{t,z}^{\text{old}} \left(\log \alpha_{t,z} - \log \alpha_{t,z}^{\text{old}} \right) \right),
\end{aligned} \tag{34}$$

where we used (31) by letting $x = \alpha_{t,z}$, $n = N_{t,d,z}$, and $\hat{x} = \alpha_{t,z}^{\text{old}}$. By using (33) and (34), we can get the following lower bound of the joint log likelihood.

$$\begin{aligned}
&\log P(\boldsymbol{W}_t, \boldsymbol{Z}_t, \boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{t-1}, \gamma, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \\
&\geq (\gamma\alpha_{t-1,z} - 1)\log\alpha_{t,z} - \gamma\alpha_{t,z} - \alpha_{t,z}\sum_d \left[\Psi\left(N_{t,d} + \sum_z \alpha_{t,z}^{\text{old}}\right) - \Psi\left(\sum_z \alpha_{t,z}^{\text{old}}\right)\right] \\
&\quad + \sum_d \left[\Psi\left(N_{t,d,z} + \alpha_{t,z}^{\text{old}}\right) - \Psi\left(\alpha_{t,z}^{\text{old}}\right)\right]\alpha_{t,z}^{\text{old}}\log\alpha_{t,z} + C''' \equiv F(\alpha_{t,z}),
\end{aligned} \tag{35}$$

where C''' is a constant that does not depend on $\alpha_{t,z}$. The derivative of $F(\alpha_{t,z})$ is as follows.

$$\begin{aligned}
\frac{\partial F(\alpha_{t,z})}{\partial \alpha_{t,z}} &= (\gamma\alpha_{t-1,z} - 1)\frac{1}{\alpha_{t,z}} - \gamma - \sum_d \left[\Psi\left(N_{t,d} + \sum_z \alpha_{t,z}^{\text{old}}\right) - \Psi\left(\sum_z \alpha_{t,z}^{\text{old}}\right)\right] \\
&\quad + \sum_d \left[\Psi\left(N_{t,d,z} + \alpha_{t,z}^{\text{old}}\right) - \Psi\left(\alpha_{t,z}^{\text{old}}\right)\right]\alpha_{t,z}^{\text{old}}\frac{1}{\alpha_{t,z}}.
\end{aligned} \tag{36}$$

The function $F(\alpha_{t,z})$ is concave, and the lower bound of the joint likelihood can be maximized by finding the value where $\frac{\partial F(\alpha_{t,z})}{\partial \alpha_{t,z}} = 0$. Thus, we can obtain (10).

C. DERIVATION OF (11)

In this appendix, we give the derivation of the update rule for $\lambda_{t,z,s}$, (11), which can also be derived in the same way as the update rule for $\alpha_{t,z}$.

The joint log likelihood can be written as follows, using (4) and (8).

$$\begin{aligned}
&\log P(\boldsymbol{W}_t, \boldsymbol{Z}_t, \boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{t-1}, \gamma, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \\
&= \log P(\boldsymbol{W}_t | \boldsymbol{Z}_t, \Xi_{t-1}, \boldsymbol{\Lambda}_t) + C \\
&= \sum_z \left[\log\Gamma\left(\sum_s \lambda_{t,z,s}\right) - \log\Gamma\left(N_{t,z} + \sum_s \lambda_{t,z,s}\right)\right] \\
&\quad + \sum_z\sum_w \left[\log\Gamma\left(N_{t,z,w} + \sum_s \lambda_{t,z,s}\xi_{t-1,z,w}^{(s)}\right) - \log\Gamma\left(\sum_s \lambda_{t,z,s}\xi_{t-1,z,w}^{(s)}\right)\right] + C,
\end{aligned} \tag{37}$$

where C is a constant that does not depend on $\lambda_{t,z,s}$. By using (27) and (31), we can show that (37) satisfies the following inequality.

$$\begin{aligned}
&\log P(\boldsymbol{W}_t, \boldsymbol{Z}_t, \boldsymbol{\alpha}_t | \boldsymbol{\alpha}_{t-1}, \gamma, \Xi_{t-1}, \boldsymbol{\Lambda}_t) \\
&\geq \sum_z -\lambda_{t,z,s}\left[\Psi\left(\sum_s \lambda_{t,z,s}^{\text{old}}\right) - \Psi\left(N_{t,z} + \sum_s \lambda_{t,z,s}^{\text{old}}\right)\right] \\
&\quad + \sum_w \left[\Psi\left(\sum_s \lambda_{t,z,s}^{\text{old}}\xi_{t-1,z,w}^{(s)}\right) - \Psi\left(N_{t,z,w} + \sum_s \lambda_{t,z,s}^{\text{old}}\xi_{t-1,z,w}^{(s)}\right)\right]\lambda_{t,z,s}^{\text{old}}\xi_{t-1,z,w}^{(s)}\log\lambda_{t,z,s} \\
&\quad + C' \equiv F\left(\lambda_{t,z,s}\right),
\end{aligned} \tag{38}$$

where we used (27) and (31), and C' is a constant that does not depend on $\lambda_{t,z,s}$. The derivative of $F(\lambda_{t,z,s})$ is as follows.

$$\begin{aligned}\frac{\partial F(\lambda_{t,z,s})}{\partial \lambda_{t,z,s}} &\\ = -&\left[\Psi\left(\sum_s \lambda^{\text{old}}_{t,z,s}\right) - \Psi\left(N_{t,z} + \sum_s \lambda^{\text{old}}_{t,z,s}\right)\right]\\ &+\sum_w \left[\Psi\left(\sum_s \lambda^{\text{old}}_{t,z,s}\xi^{(s)}_{t-1,z,w}\right) - \Psi\left(N_{t,z,w} + \sum_s \lambda^{\text{old}}_{t,z,s}\xi^{(s)}_{t-1,z,w}\right)\right]\lambda^{\text{old}}_{t,z,s}\xi^{(s)}_{t-1,z,w}\frac{1}{\lambda_{t,z,s}}.\end{aligned} \tag{39}$$

The function $F(\lambda_{t,z,s})$ is concave, and the lower bound of the joint likelihood can be maximized by finding the value where $\frac{\partial F(\lambda_{t,z,s})}{\partial \lambda_{t,z,s}} = 0$. Thus, we can get (11).

D. DERIVATION OF (16)

In this appendix, we give the derivation of (16). Let $\hat{N}^{t-1}_{t-2^{s-1},z} = \sum_w \sum^{t-1}_{t'=t-2^{s-1}} \hat{N}_{t',z,w}$, and $\hat{N}_{t,z} = \sum_w \hat{N}_{t,z,w}$. The Dirichlet prior parameter of the word distribution can be rewritten as the weighted sum of the word distributions for each epoch using (12), as follows.

$$\begin{aligned}&\sum_{s=1}^{S} \lambda_{t,z,s}\xi^{(s)}_{t-1,z,w}\\ &= \sum_{s=1}^{S} \lambda_{t,z,s}\frac{\sum^{t-1}_{t'=t-2^{s-1}}\hat{N}_{t',z,w}}{\hat{N}^{t-1}_{t-2^{s-1},z}}\\ &= \sum_{s=1}^{S}\sum_{t'=t-2^{s-1}}^{t-1} \frac{\lambda_{t,z,s}}{\hat{N}^{t-1}_{t-2^{s-1},z}}\hat{N}_{t',z,w}\\ &= \sum_{t'=t-2^{S-1}}^{t-1}\sum_{s=\lceil \log_2(t-t')+1\rceil}^{S} \frac{\lambda_{t,z,s}}{\hat{N}^{t-1}_{t-2^{s-1},z}}\hat{N}_{t',z,w}\\ &= \sum_{t'=t-2^{S-1}}^{t-1}\sum_{s=\lceil \log_2(t-t')+1\rceil}^{S} \frac{\lambda_{t,z,s}\hat{N}_{t',z}}{\hat{N}^{t-1}_{t-2^{s-1},z}}\frac{\hat{N}_{t',z,w}}{\hat{N}_{t',z}}\\ &= \sum_{t'=t-2^{S-1}}^{t-1} \lambda'_{t,z,t'}\hat{\phi}_{t',z,w}.\end{aligned} \tag{40}$$

REFERENCES

AHMED, A. AND XING, E. P. 2008. Dynamic non-parametric mixture models and the recurrent Chinese restaurant process: With applications to evolutionary clustering. In *Proceedings of the Siam International Conference on Data Mining (SDM)*.

AHMED, A. AND XING, E. P. 2010. Timeline: A dynamic hierarchical Dirichlet process model for recovering birth/death and evolution of topics in text stream. In *Proceedings of the 26th International Conference on Uncertainty in Artificial Intelligence (UAI)*.

ALSUMAIT, L., BARBARA, D., AND DOMENICONI, C. 2008. On-line LDA: Adaptive topic models for mining text streams with applications to topic detection and tracking. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. 3–12.

ANDRIEU, C., DE FREITAS, N., DOUCET, A., AND JORDAN, M. I. 2003. An introduction to MCMC for machine learning. *Mach. Learn. 50,* 1, 5–43.

BANERJEE, A. AND BASU, S. 2007. Topic models over text streams: A study of batch and online unsupervised learning. In *Proceedings of the Siam International Conference on Data Mining (SDM)*.

BLEI, D. AND FRAZIER, P. 2010. Distance dependent Chinese restaurant processes. In *Proceedings of the International Conference on Machine Learning (ICML)*.

BLEI, D. M. AND LAFFERTY, J. D. 2006. Dynamic topic models. In *Proceedings of the International Conference on Machine Learning (ICML). 113–120*.

BLEI, D. M., NG, A. Y., AND JORDAN, M. I. 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res. 3*, 993–1022.

CANINI, K. R., SHI, L., AND GRIFFITHS, T. L. 2009. Online inference of topics with latent Dirichlet allocation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*. Vol. 5, 65–72.

GERRISH, S. AND BLEI, D. 2010. A language-based approach to measuring scholarly impact. In *Proceedings of the International Conference on Machine Learning (ICML)*.

GRIFFITHS, T. L. AND STEYVERS, M. 2004. Finding scientific topics. *Proc. Nat. Acad. Sci. 101 Suppl. 1*, 5228–5235.

HOFFMAN, M., BLEI, D., AND BACH, F. 2010. Online learning for latent Dirichlet allocation. In *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*.

HOFMANN, T. 1999. Probabilistic latent semantic analysis. In *Proceedings of the International Conference on Uncertainty in Artificial Intelligence (UAI)*. 289–296.

HOFMANN, T. 2003. Collaborative filtering via Gaussian probabilistic latent semantic analysis. In *Proceedings of the Annual ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 259–266.

IWATA, T., WATANABE, S., YAMADA, T., AND UEDA, N. 2009. Topic tracking model for analyzing consumer purchase behavior. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 1427–1432.

IWATA, T., YAMADA, T., AND UEDA, N. 2008. Probabilistic latent semantic visualization: Topic model for visualizing documents. In *Proceedings of the International SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 363–371.

MINKA, T. 2000. Estimating a Dirichlet distribution. Tech. rep., MIT.

NALLAPATI, R., COHEN, W., DITMORE, S., LAFFERTY, J., AND UNG, K. 2007. Multiscale topic tomography. In *Proceedings of the International SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 520–529.

PAPADIMITRIOU, S., BROCKWELL, A., AND FALOUTSOS, C. 2003. Adaptive, hands-off stream mining. In *Proceedings of the International Conference on Very Large Databases (VLDB)*. 560–571.

PAPADIMITRIOU, S., SUN, J., AND FALOUTSOS, C. 2005. Streaming pattern discovery in multiple time-series. In *Proceedings of the International Conference on Very Large Databases (VLDB)*. 697–708.

REN, L., DUNSON, D. B., AND CARIN, L. 2008. The dynamic hierarchical Dirichlet process. In *Proceedings of the International Conference on Machine Learning (ICML)*. 824–831.

SAKURAI, Y., PAPADIMITRIOU, S., AND FALOUTSOS, C. 2005. Braid: Stream mining through group lag correlations. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD)*. 599–610.

SATO, I., KURIHARA, K., AND NAKAGAWA, H. 2010. Deterministic single-pass algorithm for LDA. In *Proceedings of the Conference on Advances in Neural Information Processing Systems (NIPS)*.

STEPHENS, M. 2000. Dealing with label switching in mixture models. *J. Royal Statist. Society B 62*, 795–809.

TEH, Y. W., JORDAN, M. I., BEAL, M. J., AND BLEI, D. M. 2006. Hierarchical Dirichlet processes. *J. Amer. Statist. Assoc. 101*, 476, 1566–1581.

WANG, C., BLEI, D. M., AND HECKERMAN, D. 2008. Continuous time dynamic topic models. In *Proceedings of the International Conference on Uncertainty in Artificial Intelligence (UAI)*. 579–586.

WANG, X. AND MCCALLUM, A. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In *Proceedings of the International SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*. 424–433.

WATANABE, S., IWATA, T., HORI, T., SAKO, A., AND ARIKI, Y. 2011. Topic tracking language model for speech recognition. *Comput. Speech Lang. 25*, 2, 440–461.

WEI, X., SUN, J., AND WANG, X. 2007. Dynamic mixture models for multiple time-series. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2909–2914.

ZHANG, J., SONG, Y., ZHANG, C., AND LIU, S. 2010. Evolutionary hierarchical Dirichlet processes for multiple correlated time-varying corpora. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. 1079–1088.

Received January 2011; revised June 2011; accepted September 2011