Tracking Trends: Incorporating Term Volume into Temporal Topic Models

Liangjie Hong† Dawei Yin† Jian Guo§ Brian D. Davison†
† Dept. of Computer Science and Engineering, Lehigh University, Bethlehem, PA, USA
§ Dept. of Statistics, University of Michigan, Ann Arbor, MI, USA
† {lih307,day207,davison}@cse.lehigh.edu, § guojian@umich.edu

ABSTRACT

Text corpora with documents from a range of time epochs are natural and ubiquitous in many fields, such as research papers, newspaper articles and a variety of types of recently emerged social media. People not only would like to know what kind of topics can be found from these data sources but also wish to understand the temporal dynamics of these topics and predict certain properties of terms or documents in the future. Topic models are usually utilized to find latent topics from text collections, and recently have been applied to temporal text corpora. However, most proposed models are general purpose models to which no real tasks are explicitly associated. Therefore, current models may be difficult to apply in real-world applications, such as the problems of tracking trends and predicting popularity of keywords. In this paper, we introduce a real-world task, tracking trends of terms, to which temporal topic models can be applied. Rather than building a general-purpose model, we propose a new type of topic model that incorporates the volume of terms into the temporal dynamics of topics and optimizes estimates of term volumes. In existing models, trends are either latent variables or not considered at all which limits the potential for practical use of trend information. In contrast, we combine state-space models with term volumes with a supervised learning model, enabling us to effectively predict the volume in the future, even without new documents. In addition, it is straightforward to obtain the volume of latent topics as a by-product of our model, demonstrating the superiority of utilizing temporal topic models over traditional time-series tools (e.g., autoregressive models) to tackle this kind of problem. The proposed model can be further extended with arbitrary word-level features which are evolving over time. We present the results of applying the model to two datasets with long time periods and show its effectiveness over non-trivial baselines.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*clustering*

KDD'11, August 21–24, 2011, San Diego, California, USA.
Copyright 2011 ACM 978-1-4503-0813-7/11/08 ...$10.00.

General Terms

Algorithms, Experimentation, Theory

Keywords

Topic models, Text mining, Temporal dynamics

1. INTRODUCTION

Text corpora with documents covering a long time-span are natural and ubiquitous in many application fields, and include such data as research papers and newspaper articles. Mining from these collections, discovering and understanding underlying topics and ideas, continues to be an important task. In addition to traditional text collections, many types of content in social media make applying machine learning techniques to these new data sources more challenging, such as forums, question answering communities and blog entries. People not only would like to know what kind of topics can be found from these data sources but also wish to understand the temporal dynamics of these topics, and hopefully predict certain properties of terms or documents in the future.

Topic models (e.g., [5]), as a class of newly developed machine learning tools, have been studied extensively in recent years. From the seminal work done by Blei et al. [5], a large body of literature about topic models has been established. Multiple disciplines of computer science, ranging from information retrieval (e.g., [24]), computer vision (e.g., [19]) to collaborative filtering (e.g., [1]) have applied topic models to their problems. For text modeling, topic models are applied to find latent topics from text collections, which is particularly useful for temporal text corpora where discovered latent topics can help researchers visualize and understand the thematic evolution of the corpora over time. This has led to the recent development of incorporating temporal dynamics into topic models (e.g., [14, 3, 21, 13, 15, 22, 20, 12, 23, 25, 2, 9, 10]). These models enable us to browse and explore datasets with temporal changes in a convenient way and open future directions for utilizing these models in a more comprehensive fashion. One drawback of these existing models is that most of them are general purpose models with which no real tasks are explicitly associated. Therefore, it might be difficult to employ these models in real-world applications, such as the problems of tracking trends and predicting popularity of keywords. As a result of the lack of a particular task, there is also no consensus on how these models should be evaluated and compared. Although perplexity is widely used in these papers, as pointed out in [6], this measure may not have correlations with the quality (e.g., coherence) of topics discovered. Furthermore, no empirical or theoretical work has been done as far as we know to show the the correlations between the low perplexity values and high performance in third-party tasks such classification, regression and clustering. In this paper, we argue that temporal topic models should be evaluated on specific real-world tasks and propose such a task to compare how they can contribute to applications. Some recent extensions of topic models (e.g., [4, 11, 26, 18]) have tried to incorporate side information, such as document-level labels and word-level features (e.g., [17]) into models in order to perform classification and regression tasks. A basic conclusion made from these attempts is that these special-purposed models, aiming to optimize particular tasks, perform better than general-purpose models, on the tasks they evaluated. We share a similar spirit in this paper, showing that temporal topic models for special tasks perform better than general-purpose models.

In this paper, we introduce a real-world task — tracking trends of terms — to which temporal topic models can be applied. Rather than building a general-purpose model, we propose a new type of topic model incorporating the volume of terms into the temporal dynamics of topics and directly optimize for the task. Unlike existing models in which trends are either latent variables or not considered at all and thus are difficult to apply in practice, we combine state-space models with term volumes in a supervised learning fashion which enables us to effectively predict volumes in the future, even without new documents. In addition, it is straightforward to obtain the volumes of latent topics as a by-product of our model, demonstrating the superiority of utilizing temporal topic models over traditional time-series tools (e.g., autoregressive models) to tackle this kind of problem. The proposed model can be further extended with arbitrary word-level features which are evolving over time. We present the results of applying the model to two datasets with long time periods and show its effectiveness over non-trivial baselines. Our contributions are threefold:

- Introduce a task — volume tracking — that can be used as a standard evaluation method for temporal topic models
- Propose a temporal topic model that directly optimizes the task introduced
- Demonstrate the effectiveness of the model as compared to state-of-the-art algorithms by experimenting on two real-world datasets

We organize the paper as follows. In Section 2, we review some related developments of topic models and existing evaluation methods for temporal topic models. In Section 3, we introduce the task of volume tracking, as a case of trend monitoring, and propose our model . In Section 4, we show how to utilize variational inference with Kalman Filter to estimate hidden parameters of the model. In Section 5, we discuss some other models that can be used in the volume tracking task. In Section 6, we demonstrate the experimental results on two datasets and conclude the paper in Section 7.

2. RELATED WORK

In this section, we review three directions of related work. First, we summarize all up-to-date topic models which try to incorporate temporal dynamics into the model. Then, we discuss the evaluation of these models and the potential to apply them in real-world applications. In the end, we present the attempts to embed side-information, or features into topic models.

To incorporate temporal dynamics into topic models, many models have been proposed. Note, as we mentioned, these attempts are general-purpose models, meaning that no real-world tasks are explicitly addressed. In general, all these models fall into two categories. The models in the first category do not impose a global distribution assumption about how topics evolve over time. In other words, these models assume that topics change over time depending on their previous conditions, effectively making "Markovian assumptions". The examples in this category are Dynamic Topic Model (DTM), proposed by Blei and Lafferty [3] and Continuous Time Dynamic Topic Models (cDTM), proposed by Wang et al. [20], embedding state-space models into topic models. Our work is inspired by this type of model. The second category of models usually imposes a global distribution of temporal dynamics. For instance, Wang et al. [21] introduce a beta distribution over timestamps and incorporate it into the standard topic model. Masada et al. [12] assume a Gaussian distribution over the whole time-line of topics. Although these models are proposed under different contexts, the drawback of this category is that the distributional assumption is hard to justify. Based on the two basic categories, other extensions are proposed. For example, Nallapati et al. [15] and Iwata et al. [9] focus on the problem of modeling topic spreading on timelines with multiple resolutions, namely how topics can be organized in a hierarchical way over time.

Table 1: Evaluation on Temporal Topic Models

(Temporal) Perplexity	[3, 15, 20, 23, 25, 2, 9, 10]
Timestamp Prediction	[21, 20, 10]
Classification/Clustering	[25]
Ad-Hoc	[21, 23, 25]

As in traditional topic models, the effectiveness of temporal topic models is difficult to evaluate in general. This is partly because these models are introduced without considering any tasks, making the process of evaluating them on third-party tasks ad-hoc. Due to a lack of evaluation tasks, comprehensive comparisons between models are seldom conducted. In order to better illustrate how temporal topic models have been evaluated, we show them in Table 1, according to the evaluation methods mentioned in papers. It is clear that temporal perplexity is a popular evaluation method. However, as pointed out in [6], perplexity may not have correlations with the quality (e.g., coherence) of latent topics. In addition, little is known, both theoretically and empirically, that a model achieving lower perplexity will perform better on real-world applications which we care about. Besides perplexity, several papers proposed some ad-hoc evaluation methods (named under "Ad-hoc" in the table) to demonstrate the potential capabilities of their models, such as the coherence of topics measured by K-L divergence, where these methods are not shared by other papers and are also not really task-driven. Nearly all papers show "anecdotal examples" of what kind of topics are found over time.

Since our model can be considered as an extension to incorporate side information, or features into topic models, we also review other similar attempts. Basically, two kinds of side information might be considered: document-level features and word-level features. For document-level features, models are proposed (e.g., [4, 11, 26, 18]) to incorporate them either conditioned on latent topic assignments or conditioned on per-document hyper-parameters. Either maximum conditional learning or max-margin learning is employed for inference. For word-level features, a recently proposed model [17] introduce a method to embed arbitrary word-level features. Unlike the ones for document-level features, this model is not a fully generative model and therefore we cannot easily infer these feature values.

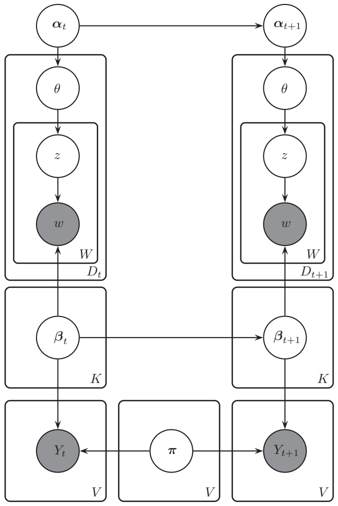

Figure 1: A graphical representation of the model with only two time epochs

3. TRACKING TRENDS BY INCORPORATING VOLUMES

In this section, we will introduce the task of volume prediction as a case of trend tracking. One reason that temporal topic models are favored is perhaps that these models can be potentially used as a tool to analyze trends and changes of keywords over time. However, these tasks are never evaluated directly or seriously in current literature.

The task of predicting the volume of terms is to predict the numeric volume of one or a set of keywords, given the historical data of these keywords in the past. This is a natural extension of tracking and monitoring keywords over time. Indeed, some commercial products provide such tools to allow users to browse and understand the rise and fall of keywords, such as `Google Trends`. One drawback of existing tools is that people usually only have a limited view of certain topics in which they are interested before they fully understand these topics. For instance, for the event of "World Cup", the phrase "World Cup" is certainly of interest. However, there are many more related terms to be explored, such as "FIFA", "South Africa" and "Ronaldo". Sometimes, users have these related terms in mind but usually they are unable to prepare them in advance. It would be great if users could track the trends (volume) of a topic as a whole and discover all those related terms at the same time. Moreover, the volume of terms in the same topic are correlated, which may help the model to find better topics. Overall, we would like to achieve three goals in tracking trends:

- Track and predict the volume of individual terms
- Obtain latent topics so that related terms can be grouped together
- Model the evolution of latent topics

The second goal will happen automatically through the modeling of topic models. The last goal can be achieved by temporal topic models, through either one of the assumptions mentioned in Section 2. The first goal is the center of this work. We believe that our work would help to track the volume of topics as a whole if the first goal can be achieved. Note, in terms of "prediction", we indicate the ability to estimate the volume of individual terms in the future where no documents are realized.

Two design issues need to be tackled when introducing term volumes into the model. First, they are word-level variables (if we treat features as random variables). Second, we need to predict values of these variables without documents. These two issues prevent these variables from being placed in the document plates, in terms of graphical modeling. This decision distinguishes our model from previous models (e.g., [4, 11, 26, 18]) where response variables are placed in document plates. Recently, Petterson et al. [17] demonstrate a technique to embed word-level features into topic models. Although our work shares similar ideas to theirs, their model is not a generative model for word features but only for words in the documents. In addition, their work is not to predict these word-level features. Since their work is for a static text corpus, it cannot be easily utilized to model temporal data. Therefore, we do not include this model in our experiments for comparison. Our model is a fully generative model for both word instantiations in documents and word-level features.

Before we further go to the formal description of our model, we discuss some intuitions behind the model. In standard topic models, each word v is associated with many latent topics $\boldsymbol{\beta}_{1:K}$. Each topic β_k is a distribution over all terms in the vocabulary V. Intuitively, the more a term appears in many topics, the more likely the term will have a high volume, such as some stop words and functional words. On the other hand, many terms only appear in a handful of topics and therefore these topics determine the volume of the term. If we think of $\boldsymbol{\beta}$ as another representation of terms, we would like to associate these latent variables with the term volumes. Following this intuition, we treat the volume of term v at time-stamp t, denoted as $Y_v^{(t)}$, as a function of latent topics $\boldsymbol{\beta}$. The simplest form of such functions is a linear function:

$$Y_v^{(t)} = \sum_{k=0}^{K} \pi_{(v,k)} \beta_{(k,v)}^{(t)} + \epsilon_v \tag{1}$$

where $\boldsymbol{\pi_v}$ is a vector of coefficients, $\beta_{(k,v)}^{(t)}$ is the probability that the term is "generated" from topic k at time stamp t, and ϵ_v is a per-term "error". In other words, the volume of a term v depends on its prevalence in all topics at that time point. If ϵ_v follows a normal distribution, namely $\epsilon_v \sim N(0, \sigma_v^2)$, we can express the generation process of $Y_V^{(t)}$ in terms of a Normal distribution as follows:

$$Y_v^{(t)} \mid \pi_{(v)}, \beta_{(*,v)}^{(t)} \sim \mathcal{N}\left(\pi_v^T \beta_{(*,v)}^{(t)}, \sigma_v^2\right) \tag{2}$$

Here, $Y_v^{(t)}$ is treated as a real valued variable. In our experiments, we use the raw counts of term v at time epoch t as $Y_v^{(t)}$.

In order to obtain Y_v at different time epochs, we need to have $\boldsymbol{\beta}$ for different time points. We mention two basic categories of approaches in Section 2 and here we adapt the first category, having a "Markovian assumption" on the evolution of topics over time. More specifically, topics $\boldsymbol{\beta}$ evolve according to a state-space model and the documents with their words are "generated" by the corresponding topics in the same time epoch. Embedding these intuitions into the model, the generative process of the model is as follows:

1. For each topic k in K:
   Draw topics $\beta_k^{(t)} \mid \beta_k^{(t-1)} \sim \mathcal{N}\left(\beta_k^{(t-1)}, \delta^2 I\right)$.
2. For each term v in V:
   Draw term volume $Y_v^{(t)} \sim \mathcal{N}\left(\pi_v^T \beta_{(*,v)}^{(t)}, \sigma^2\right)$.
3. For each document d in time epoch t:
   (a) Draw $\theta_d \sim \mathrm{Dir}(\boldsymbol{\alpha})$

(b) For each word n:

i. Draw $z_{(d,n)} \sim \text{Multi}(\theta)$.

ii. Draw $w_{(d,n)} \sim \text{Multi}\Big(f(\beta_z^{(t)})\Big)$

where function f maps the multinomial natural parameters to mean parameters. The graphical representation of the model is shown in Figure 1. Note, the model can be easily extended in multiple ways. For instance, we can also allow the hyper-parameters of topic proportions $\boldsymbol{\alpha}$ to evolve over time, according to a different state-space model, as already mentioned in [3]. In addition, the simple state-space model can be replaced by a Brownian motion model [20], allowing arbitrary granularity of time-series. We will explore these extensions in future work.

4. VARIATIONAL INFERENCE WITH KALMAN FILTERING

The central problem in topic modeling is posterior inference, i.e., determining the distribution of the latent topic structure conditioned on the observed documents. In our case, the latent structures comprise the per-document topic proportions θ_d, per-word topic assignments $z_{(d,n)}$, the K sequences of topic distributions $\beta_k^{(t)}$ and per-term coefficient vector $\boldsymbol{\pi}_v$ for characterizing term volumes. Similar to many topic models, the true posterior is intractable [3, 20], meaning that we must appeal to an approximation.

Several approximate inference approaches have been developed for topic models. The most widely used are variational inference (e.g., [5, 3, 20]) and collapsed Gibbs sampling (e.g., [7, 21]). As noted previously by others [3, 20], collapsed Gibbs sampling is not an option in the sequential setting because the distribution of words for each topic is not conjugate to the word probabilities. Therefore, we employ variational inference for the model.

The main idea behind variational inference is to posit a simple family of distributions over the latent variables, namely variational distributions, and to find the member of that family which is closest in Kullback-Leibler divergence to the true posterior. Variational inference has been successfully adopted in temporal topic models (e.g., [3, 15, 20]).

For the model descried above, we adapt variational Kalman filtering [3] to the sequential modeling setting. We employ the following variational distribution:

$$
\begin{aligned}
& q(\boldsymbol{\beta}_{1:T}, \boldsymbol{\theta}, \mathbf{Z} | \hat{\boldsymbol{\beta}}_{1:T}, \boldsymbol{\lambda}, \boldsymbol{\Phi}) = \\
& \prod_{k=1}^{K} q(\beta_k^1, \cdots, \beta_k^T | \hat{\beta}_k^1, \cdots, \hat{\beta}_k^T) \times \\
& \prod_{t=1}^{T} \Big(\prod_{d=1}^{D_t} q(\theta_d | \lambda_d) \prod_{n=1}^{N_d} q\big(z_{(d,n)} | \phi_{(d,n)}\big)\Big) \qquad (3)
\end{aligned}
$$

The variational parameters are a Dirichlet λ_d for the per-document topic proportions, multinomials ϕ for each word's topic assignment, and $\hat{\beta}$ variables, which are "observations" to a Variational Kalman Filter. The central idea of the variational Kalman filter is that variational parameters are treated as "observations" in a common Kalman filter setting, while true parameters, here $\boldsymbol{\beta}^{(t)}$, are treated as latent states of the model. By utilizing a Kalman filter, we can effectively estimate these "latent states" through "observations".

More specifically, our state space model is:

$$
\begin{aligned}
\beta_k^{(t)} \,|\, \beta_k^{(t-1)} &\sim \mathcal{N}\Big(\beta_k^{(t-1)}, \delta^2 I\Big) \\
\hat{\beta}_k^{(t)} \,|\, \beta_k^{(t)} &\sim \mathcal{N}\Big(\beta_k^t, \hat{\delta}_t^2 I\Big) \qquad (4)
\end{aligned}
$$

The variational parameters are $\hat{\beta}_k^{(t)}$ and $\hat{\delta}_t$. The key problem of Kalman filter is to derive the mean and variance for forward and backward equations, which can be used to calculate the lower bound in variational inference. Using the standard Kalman filter calculation, the forward mean and variance of the variational posterior are given by:

$$
\begin{aligned}
m_k^t &= \mathbb{E}[\beta_k^t | \hat{\boldsymbol{\beta}}_k^{1:t}] \\
&= \left(\frac{\hat{\delta}^2}{V_k^{t-1} + \delta^2 + \hat{\delta}^2} \right) m_k^{t-1} + \left(1 - \frac{\hat{\delta}^2}{V_k^{t-1} + \delta^2 + \hat{\delta}^2} \right) \hat{\beta}_k^t \\
V_k^t &= \mathbb{E}\Big[(\beta_k^t - m_k^t]) \,|\, \hat{\boldsymbol{\beta}}_k^{1:t} \Big] \\
&= \left(\frac{\hat{\delta}^2}{V_k^{t-1} + \delta^2 + \hat{\delta}^2} \right) (V_k^{t-1} + \delta^2) \qquad (5)
\end{aligned}
$$

with initial conditions specified by fixed m^0 and V^0. The backward recursion then calculates the marginal mean and variance of β_k^t given $\hat{\boldsymbol{\beta}}_k^{1:T}$ as:

$$
\begin{aligned}
\widetilde{m}_k^{t-1} &= \mathbb{E}[\beta_k^{t-1} | \hat{\boldsymbol{\beta}}_k^{1:T}] \\
&= \left(\frac{\delta^2}{V_k^{t-1} + \delta^2} \right) m_k^{t-1} + \left(1 - \frac{\delta^2}{V_k^{t-1}] + \delta^2} \right) \widetilde{m}_k^t \\
\widetilde{V}_k^{t-1} &= \mathbb{E}\Big[(\beta_k^{t-1} - \widetilde{m}_k^{t-1}]) \,|\, \hat{\boldsymbol{\beta}}_k^{1:T} \Big] \\
&= V_k^{t-1} + \left(\frac{V_k^{t-1}}{V_k^{t-1} + \delta^2} \right)^2 \Big(\widetilde{V}_k^t - (V_k^{t-1} + \delta^2) \Big) \qquad (6)
\end{aligned}
$$

with initial conditions $\widetilde{m}^T = m^T$ and $\widetilde{V}^T = V^T$.

With these forward and backward equations in hand, we turn to calculate the following lower bound (assuming $\boldsymbol{\Omega} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\pi}, \sigma^2\}$) with the help of variational distributions introduced in Equation 5:

$$
\begin{aligned}
& \log P(\mathbf{W}, \mathbf{Y} | \boldsymbol{\Omega}) \geq \mathbb{E}_q[\log p(\boldsymbol{\beta})] + \mathbb{E}_q[\log p(\mathbf{W}, \mathbf{Z}, \boldsymbol{\theta} | \boldsymbol{\beta}, \boldsymbol{\alpha})] \\
& + \mathbb{E}_q[\log p(\mathbf{Y} | \boldsymbol{\pi}, \boldsymbol{\beta}, \sigma^2)] + H(q) \\
& = \mathbb{E}_q[\log p(\boldsymbol{\beta})] + \mathbb{E}_q[\log p(\mathbf{W} | \mathbf{Z}, \boldsymbol{\beta})] + \mathbb{E}_q[\log p(\mathbf{Z} | \boldsymbol{\theta})] \\
& + \mathbb{E}_q[\log p(\boldsymbol{\theta} | \boldsymbol{\alpha})] + \mathbb{E}_q[\log p(\mathbf{Y} | \boldsymbol{\pi}, \boldsymbol{\beta}, \sigma^2)] + H(q) \qquad (7)
\end{aligned}
$$

where term $H(q)$ is the entropy. To tighten the above bound on the likelihood of the observations given by Jensen's inequality is equivalent to minimize KL-divergence. In the above bound, the term $\mathbb{E}_q[\log p(\mathbf{W}, \mathbf{Z}, \boldsymbol{\theta} | \boldsymbol{\beta}, \boldsymbol{\alpha})]$ is standard for topic models, when logistic-normal distribution is applied to represent topics (e.g., [3, 20]. The term $\mathbb{E}_q[\log p(\boldsymbol{\beta})]$ is standard for temporal topic models, which utilize the Kalman filter as a sequantial modeling tool. The term $\mathbb{E}_q[\log p(\mathbf{Y} | \boldsymbol{\pi}, \boldsymbol{\beta}, \sigma^2)]$ can be calculated similarly to the document-level response variables, introduced in [4]. We will discuss these expectations in detail.

For the first term of the last line in Equation 7, we utilize the forward and backward equations introduced in Equation 6 and follow the similar steps in [3]:

$$
\begin{aligned}
& \mathbb{E}_q[\log p(\boldsymbol{\beta})] = -\frac{VKT}{2}(\log \delta^2 + \log 2\pi) \\
& - \frac{1}{2\delta^2} \sum_{t=1}^{T} \sum_{k=1}^{K} \left[\Big(\widetilde{m}_k^t - \widetilde{m}_k^{t-1} \Big)^2 \right] - \frac{1}{\delta^2} \sum_{t=1}^{T} \sum_{k=1}^{K} \text{Tr}\Big(\widetilde{V}_k^t \Big) \\
& + \frac{1}{2\delta^2} \sum_{k=1}^{K} \text{Tr}\Big(\widetilde{V}_k^T \Big) - \frac{1}{2\delta^2} \sum_{k=1}^{K} \text{Tr}\Big(\widetilde{V}_k^0 \Big)
\end{aligned}
$$

For the second term in the same line, we have:

$$\begin{aligned}\mathbb{E}_q[\log p(\mathbf{W}|\mathbf{Z},\boldsymbol{\beta})] &= \sum_{t=1}^{T}\sum_{d=1}^{D_t}\sum_{n=1}^{N_d}\Bigg(\sum_{k=1}^{K}\phi_{(n,k)}\widetilde{m}^{t}_{(k,w)} \\ &- \sum_{k=1}^{K}\phi_{(n,k)}\mathbb{E}_q\Big[\log\sum_{w'}\exp(\beta_{(k,w')})\Big]\Bigg)\end{aligned}$$

where the second line demonstrates the essential problem of non-conjugacy of using the logistic-normal distribution for topics. In order to calculate $\mathbb{E}_q\Big[\log\sum_{w'}\exp(\beta_{(k,w')})\Big]$, we further obtain a lower bound by introducing another variational parameter ζ_t and upper bound the negative log normalizer with a Taylor expansion as follows:

$$\begin{aligned}\mathbb{E}_q\Big[\log\sum_{w'}\exp(\beta_{(k,w')})\Big] &\leq \zeta_t^{-1}\Big(\sum_{w'}\mathbb{E}_q[\exp(\beta_{(k,w')})]\Big) \\ &-1+\log(\zeta_t)\end{aligned}$$

where the expectation $\mathbb{E}_q[\exp(\beta_{(k,w')})]$ is the mean of a log normal distribution with the mean and variance obtained from the variational parameters, essentially Kalman Filters, in our case. For the third term of the last line in Equation 7, we have:

$$\begin{aligned}\mathbb{E}_q[\log p(\mathbf{Z}|\boldsymbol{\theta})] &= \sum_{t=1}^{T}\sum_{d=1}^{D_t}\sum_{n=1}^{N_d}\sum_{k=1}^{K}\phi_{(n,k)}\Big[\Psi(\lambda_{(d,k)}) \\ &-\Psi\Big(\sum_{k'=1}^{K}\lambda_{(d,k')}\Big)\Big]\end{aligned}$$

and for the fourth term, we have:

$$\begin{aligned}\mathbb{E}_q[\log p(\boldsymbol{\theta}|\boldsymbol{\alpha})] &= \sum_{t=1}^{T}\sum_{d=1}^{D_t}\Bigg\{\Bigg(\sum_{k=1}^{K}(\alpha_k-1)\Big[\Psi(\lambda_{(d,k)}) \\ &-\Psi\Big(\sum_{j=1}^{K}\lambda_{(d,j)}\Big)\Big]\Bigg)+\log\Gamma\Big(\sum_{k=1}^{K}\alpha_k\Big)-\sum_{k=1}^{K}\log\Gamma(\alpha_k)\Bigg\}\end{aligned}$$

For the last term in the same line, we have:

$$\begin{aligned}\mathbb{E}_q[\log p(Y_v^{(t)}|\pi_v,\beta^{(t)}_{(,v)},\sigma^2)] &= -\frac{1}{2}\log 2\pi-\frac{1}{2}\log\sigma^2 \\ &-\frac{\Big(Y_v^{(t)}\Big)^2}{2}+\frac{1}{\sigma^2}\Bigg[Y_v^{(t)}\sum_{k=1}^{K}\pi_{(v,k)}\widetilde{m}^{t}_{(k,v)} \\ &-\frac{1}{2}\sum_{i=1}^{K}\sum_{j=1}^{K}\pi_{(v,i)}\Big(\widetilde{m}^{t}_{(i,v)}\widetilde{m}^{t}_{(j,v)}\Big)\pi_{(v,j)}\Bigg]\end{aligned}$$

For the entropy term $H(q)$, we have:

$$\begin{aligned}-H(q) &= \mathbb{E}_q[\log q(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}})]+\mathbb{E}_q[\log q(\boldsymbol{\theta}|\boldsymbol{\lambda})]+\mathbb{E}_q[\log q(\mathbf{Z}|\boldsymbol{\Phi})] \\ &= \sum_{t=1}^{T}\sum_{k=1}^{K}\Big(\frac{T}{2}\log 2\pi\Big)+\frac{1}{2}\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{v=1}^{V}\log\widetilde{V}^{t}_{(k,v)}+\sum_{t=1}^{T}\sum_{d=1}^{D_t} \\ &\Bigg\{\Bigg(\sum_{k=1}^{K}(\lambda_{(d,k)}-1)\Big[\Psi(\lambda_{(d,k)})-\Psi\Big(\sum_{j=1}^{K}\lambda_{(d,j)}\Big)\Big]\Bigg) \\ &+\log\Gamma\Big(\sum_{k=1}^{K}\lambda_{(d,k)}\Big)-\sum_{k=1}^{K}\log\Gamma(\lambda_{(d,k)})\Bigg\} \\ &+\sum_{t=1}^{T}\sum_{d=1}^{D_t}\sum_{n=1}^{N_d}\sum_{k=1}^{K}\phi_{(n,k)}\log\phi_{(n,k)}\end{aligned}$$

Algorithm 1: Variational inference with Kalman filtering.

```
Initialize $\hat{\boldsymbol{\beta}}$ randomly.
while relative improvement in $L > 0.00001$ do
    E step:
    for $t = 1$ to $T$ do
        for $i = 1$ to $D$ do
            Update $\lambda_d$ according to Equation 8
            Update $\phi_d$ according to Equation 9
        Update $\zeta_t$ according to Equation 10
    M step:
    for $v = 1$ to $V$ do
        Update $\pi_v$ according to Equation 12
        Update $\sigma_v^2$ according to Equation 13
    Update $\hat{\beta}$ by using conjugate gradient descent
```

By using the expectations with respect to variational distributions, we can optimize the variational parameters as follows. For per-document parameters $\lambda_{(d,k)}$, per-word parameters ϕ_n and per time epoch parameters ζ_t, we have similar update equations as standard topic models:

$$\lambda_{(d,k)} = \alpha_k+\sum_{n=1}^{N_d}\phi_{(n,k)} \tag{8}$$

$$\begin{aligned}\phi_{(n,k)} &\propto \exp\Bigg(\Psi(\lambda_{(d,k)})-\Psi\Big(\sum_{k'=1}^{K}\lambda_{(d,k')}\Big)\Bigg)\times \\ &\exp\Bigg(\widetilde{m}^{t}_{(k,w)}-\mathbb{E}_q\Big[\log\sum_{w'}\exp(\beta_{(k,w')})\Big]\Bigg)\end{aligned} \tag{9}$$

$$\begin{aligned}\zeta_t &= \frac{1}{N_t}\sum_{d=1}^{D_t}\sum_{n=1}^{N_d}\Bigg(\sum_{k=1}^{K}\phi_{(n,k)}\sum_{w}\exp\Big(\widetilde{m}^{t}_{(k,w)} \\ &+\widetilde{V}^{t}_{(k,w)}/2\Big)\Bigg)\end{aligned} \tag{10}$$

Since π_v is a vector of coefficients across all time epochs T, we gather the $\beta^{*}_{(*,v)}$ from all time epochs and form a $T\times K$ matrix X where each row is a vector of β values discussed before. We can obtain the following equation by using the notation of X:

$$\mathbb{E}_q[\mathbf{X}^T\mathbf{X}]\pi_v = \mathbb{E}_q[\mathbf{X}]^T\mathbf{Y}_v \tag{11}$$

and therefore, we have

$$\pi_v = \Big(\mathbb{E}_q[\mathbf{X}^T\mathbf{X}]\Big)^{-1}\mathbb{E}_q[\mathbf{X}]^T\mathbf{Y}_v \tag{12}$$

where the t^{th} row of $\mathbb{E}_q[\mathbf{X}]$ is just $\mathbb{E}_q[\beta^{t}_{(,v)}]$. Similar to linear regression but in the expected version, we can obtain the update equation for σ_v^2 as:

$$\sigma_v^2 = \frac{1}{T}\Bigg(\mathbf{Y}_v^T\mathbf{Y}_v-2\mathbf{Y}_v^T\mathbb{E}_q[\mathbf{X}]\pi_v+\pi_v^T\mathbb{E}_q[\mathbf{X}^T\mathbf{X}]\pi_v\Bigg) \tag{13}$$

where π_v is the new estimate value.

The real computational hurdle is to calculate the updates of $\hat{\boldsymbol{\beta}}$. Gathering all terms in the lower bound involving $\boldsymbol{\beta}$ and differenti-

Table 2: `AR` model on `NIPS` dataset

p	2007	2008	2009	Avg.
1	98.57	90.51	99.42	96.17
2	101.72	83.20	**91.06**	92.00
3	**97.66**	77.31	97.00	**90.39**
4	112.83	**75.62**	95.98	94.81
5	118.10	91.64	108.33	106.03
6	118.65	99.00	108.34	108.66
7	118.76	98.99	117.50	111.75
8	122.73	95.93	116.72	111.79
9	122.55	96.23	115.85	111.54
10	143.17	100.71	124.40	122.76

Table 3: `AR` model on `ACL` dataset

p	2005	2006	2007	2008	2009	Avg.
1	**131.85**	524.04	**39.57**	592.91	**126.29**	282.93
2	210.74	316.38	106.31	434.15	181.98	249.91
3	247.73	248.17	104.72	**381.84**	140.87	**224.65**
4	258.74	246.58	114.23	447.71	166.09	246.67
5	244.41	**223.99**	53.12	428.17	185.00	226.94
6	250.49	297.98	42.74	385.26	209.24	237.14
7	169.25	328.75	51.14	345.98	262.54	231.53
8	168.54	332.20	51.58	396.08	291.13	247.90
9	155.96	326.73	47.11	400.96	291.60	244.47
10	156.59	355.13	49.15	399.28	310.65	254.16

ating them with respect to $\hat{\beta}^t_{(k,v)}$, we have:

$$
\begin{aligned}
&-\frac{1}{\delta^2}\sum_{t=1}^{T}\Big(\widetilde{m}^t_{(k,v)} - \widetilde{m}^{t-1}_{(k,v)}\Big)\left(\frac{\partial\widetilde{m}^t_{(k,v)}}{\partial\hat{\beta}^t_{(k,v)}} - \frac{\partial\widetilde{m}^{t-1}_{(k,v)}}{\partial\hat{\beta}^t_{(k,v)}}\right) \\
&+\sum_{t=1}^{T}\Bigg(N_{(t,v)}\phi_{(v,k)} - \sum_{v=1}^{V} N_{(t,v)}\phi_{(v,k)}\zeta_t^{-1}\exp\Big(m^t_{(k,v)} \\
&+V^t_{(k,v)}/2\Big)\Bigg)\frac{\partial\widetilde{m}^t_{(k,v)}}{\partial\hat{\beta}^t_{(k,v)}} + \frac{1}{\sigma^2}\sum_{t=1}^{T} Y^t_v \pi_{(v,k)}\frac{\partial\widetilde{m}^t_{(k,v)}}{\partial\hat{\beta}^t_{(k,v)}} \\
&-\left[\frac{1}{2\sigma^2}\sum_{i=1}^{K}\sum_{j=1}^{K}\pi_{(v,i)}\Big(\widetilde{m}^t_{(i,v)}\widetilde{m}^t_{(j,v)}\Big)\pi_{(v,j)}\right]\frac{\partial\widetilde{m}^t_{(k,v)}}{\partial\hat{\beta}^t_{(k,v)}}
\end{aligned}
$$

Unfortunately, no closed-form solution for $\hat{\beta}$ can be found. We adapt optimization techniques to obtain a local optimum of the $\hat{\beta}$ values. In our experiments, we utilize the conjugate gradient algorithm implemented in `GSL` library[1], which requires us to provide the gradients. The forward-backward equations for $\mathbb{E}_q$ can be used to derive a recurrence for the gradients. The forward recurrence is:

$$
\begin{aligned}
\frac{\partial m^t_{(k,v)}}{\partial\hat{\beta}^{(s)}_{(k,v)}} &= \left(\frac{\hat{\delta}^2}{V_k^{t-1} + \delta^2 + \hat{\delta}^2}\right)\frac{\partial m_k^{t-1}}{\partial\hat{\beta}^s_{(k,v)}} \\
&+ \left(1 - \frac{\hat{\delta}^2}{V_k^{t-1} + \delta^2 + \hat{\delta}^2}\right)\mathbb{I}[s == t]
\end{aligned}
$$

with the initial condition $\partial m_k^0/\partial\hat{\beta_k}^s = 0$. The backward recurrence is then:

$$
\begin{aligned}
\frac{\partial\widetilde{m}^t_k}{\partial\hat{\beta_k}^s} &= \left(\frac{\delta^2}{V_k^{t-1} + \delta^2}\right)\frac{\partial m_k^{t-1}}{\partial\hat{\beta_k}^s} \\
&+ \left(1 - \frac{\delta^2}{V_k^{t-1} + \delta^2}\right)\frac{\partial m^t_k}{\partial\hat{\beta_k}^{(s)}}
\end{aligned}
$$

with the initial condition $\partial\widetilde{m}^T_k/\partial\hat{\beta_k}^s = \partial m^T_k/\partial\hat{\beta_k}^s$. We outline the overall inference algorithm in Algorithm (1).

For prediction, since no documents are observed at test time, we initialize β values with their expected values, according to Equation 4 and then obtain the mean of the posterior distribution by the Kalman filter algorithm, as a standard problem. By using the learned $\boldsymbol{\pi}$ values, we could easily predict the volume of terms through Equation 1.

5. BASELINE MODELS

Time series analysis has been long studied in many fields. Here, we discuss the possibility to employ one traditional time series tool, autoregressive model, to track the volume of terms. In univariate autoregressive model `AR`(p), a response X_t can depend on its previous values, ranging from X_{t-1} to X_{t-p}:

$$X_t = w + \sum_{k=1}^{p}\pi_k X_{t-k} \tag{14}$$

where w is a constant and $\boldsymbol{\pi}$ is a vector of coefficients. Similar to linear regression, the aim of `AR`(p) is to learn w and $\boldsymbol{\pi}$, as well as the optimal choice of p, sometimes. If we treat the volume of each term as X, it is obvious that the volume of terms are independent with each other. A slightly more complicated model, Multivariate AutoRegressive model `MAR`(p), captures the correlations between M variables and preserves the simplicity of the model:

$$\mathbf{X}_t = \mathbf{w} + \sum_{k=1}^{p}\mathbf{A}_k\mathbf{X}_{t-k} \tag{15}$$

where $\mathbf{X}$ and $\mathbf{w}$ are both M dimensional vectors and each $\mathbf{A}$ is a $M \times M$ matrix, encoding the correlations. Although it first seems appealing, some limitations of the model prevent it from being applied in text mining scenarios. One of the drawbacks is that the model usually requires the number of variables to be smaller than the time stamps, which is not a problem in many traditional fields (e.g., temperature and humidity over time). However, in many text corpora, we wish to track thousands, or even millions of terms (e.g., in Twitter) while the total number of time epochs to be measured is significantly smaller (e.g., in year, months, days). In that case, it is impossible to solve the Equation 15, according to Neumaier and Schneider [16]. Therefore, we do not use `MAR` in our experiments.

The second baseline used in experiments is Latent Dirichlet Allocation (`LDA`) [5]. We run `LDA` for the whole dataset. For each time epoch t, we obtain empirical topic distributions on t, $\boldsymbol{\beta}^t$. For each term v, we treat $\beta_{(,v)}$ as features and $Y_v^{(t)}$ as the response, building a regression model on them. Note, this model is unrealistic because in reality, we cannot obtain empirical topic distributions from the test set due to the fact that no documents should be observed from the test set. However, we include this model in the experiments for the purpose to show that topic representations can help volume prediction. A more realistic state-of-the-art model, `DTM`, is also used in the experiments. Like our model, $\boldsymbol{\beta}$ values on the test time epoch are estimated by the Kalman filter algorithm. Similar to `LDA`, the topic distributions obtained by `DTM` are treated as features and we build a regression model based upon these features. The regression model used in experiments is Support Vector Regression (`SVR`), implemented in `libSVM`[2].

[1] http://www.gnu.org/software/gsl/

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

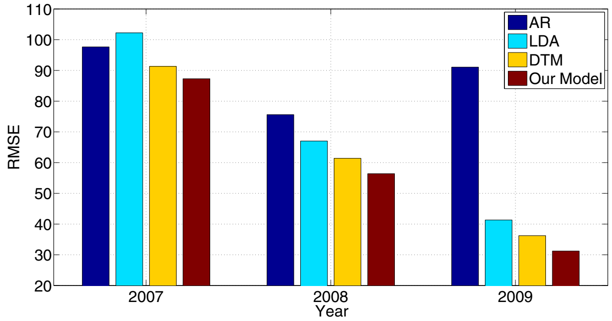

Figure 2: Performance comparison on the NIPS dataset. The best RMSE values achieved by each model are shown for the last three years.

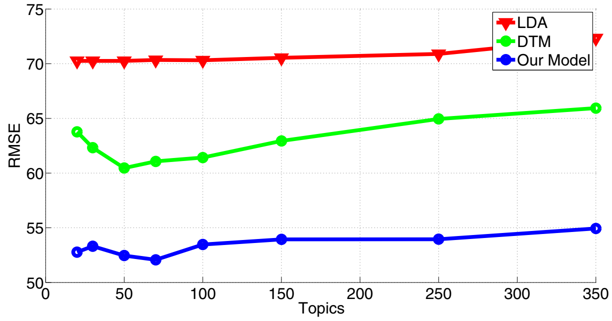

Figure 3: Performance comparison by varying the number of topics K on the NIPS dataset.

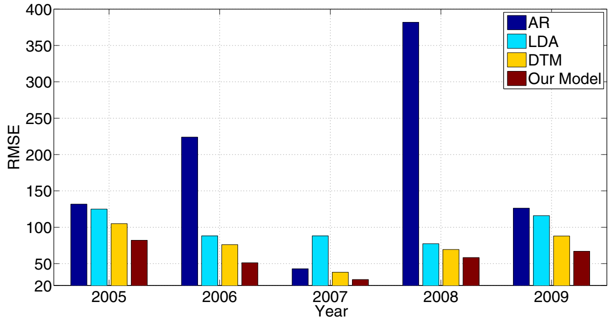

Figure 4: Performance comparison on the ACL dataset. The best RMSE values achieved by each model are shown for the last five years.

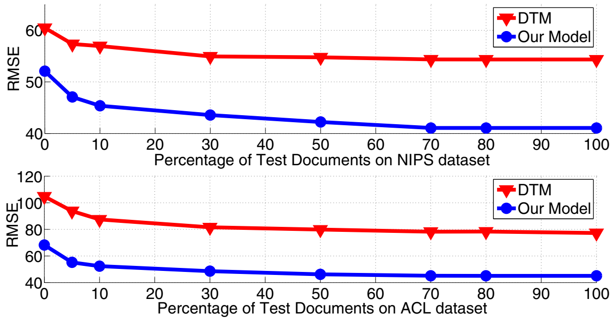

Figure 5: Performance when a fraction of the test documents is provided to the model.

6. EXPERIMENTS

Two datasets of scientific papers are used in our experiments. One is from the NIPS conference series. We downloaded all electronic copies of papers from online proceedings[3] and converted into text format using pdftotext. We tokenize the converted files and keep the terms with frequency larger than 10, resulting in to 38,029 distinct terms and 4,360 papers in total, spanning 24 years. The second dataset is from the 2009 release of The ACL Anthology[4], consisting of text format of papers published in the community of computational linguistics. This dataset has 14,590 papers with 74,189 distinct terms (frequency more than 10), ranging over 37 years. Both datasets have timelines that are long enough such that some topics have changed over time.

The major evaluation measure is of course the accuracy of the predicted volume of terms. In this work, we denote the estimated volume of term v at time stamp t as $\hat{Y}_v^{(t)}$. Therefore, we measure the estimation error by calculating the Root Mean Square Error (RMSE) between estimated values and real values:

$$\mathrm{RMSE}_t = \sqrt{\frac{1}{V}\sum_v \left(\hat{Y}_v^{(t)} - Y_v^{(t)}\right)^2}$$

For both datasets, we adapt an "incremental" evaluation process, mimicking real application scenarios. In order to predict the volume at time t, we use the documents in all possible previous years for training. We sequentially train and test the model in multiple years and average the RMSE over these time periods. We conduct experiments on the last three years for the NIPS dataset and the last five years for the ACL dataset. For hyper-parameters, α is set to $50/K$, δ^2 is set to 0.1 and $\hat{\delta}^2$ is set to 1.0, similar as [3], for all experiments.

6.1 Volume Prediction

As discussed in Section 5, the first baseline we consider is the AR model for terms. In our case, we essentially build an AR model for each term. Rather than choosing the optimal p by some criteria, such as Bayesian information criterion (BIC)[5] or Akaike information criterion (AIC)[6], we simply show the predictive performance by varying p values. Therefore, it is possible that the optimal p value is out of the ranges demonstrated here. The results for the AR model on the NIPS dataset are shown in Table 2 and the results on the ACL dataset are shown in Table 3, where the optimal performance is in bold. Several conclusions can be made regarding these results. First, for both datasets, the optimal performance is not always obtained on $p = 1$, when the volume of terms only depends on the previous year. On average, $p = 3$ gives optimal performance on both datasets, meaning that the volume of terms in the year t depends on the previous three years. For the NIPS dataset, after the optimal point, the performance decreases as p increases, which indicates that for the AR model, no additional advantages can be obtained if we consider higher order dependencies on this particular dataset. This observation might also indicate that the latent relationships among terms, essentially topics, may change over time. Some new terms are introduced and some old concepts are outdated. For the ACL dataset, this is more complicated since the performance fluctuates significantly as p varies. Unlike the the

[3] http://books.nips.cc/

[4] http://clair.si.umich.edu/clair/anthology/

[5] http://en.wikipedia.org/wiki/Bayesian_information_criterion

[6] http://en.wikipedia.org/wiki/Akaike_information_criterion

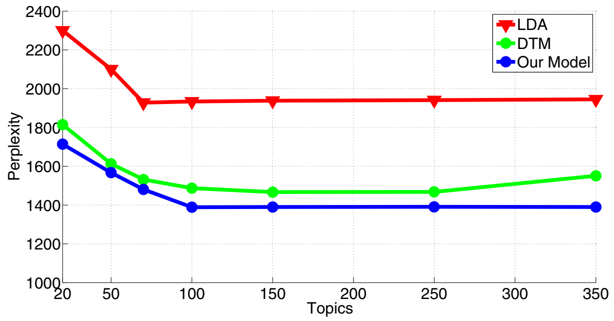

Figure 6: Perplexity comparison on NIPS dataset.

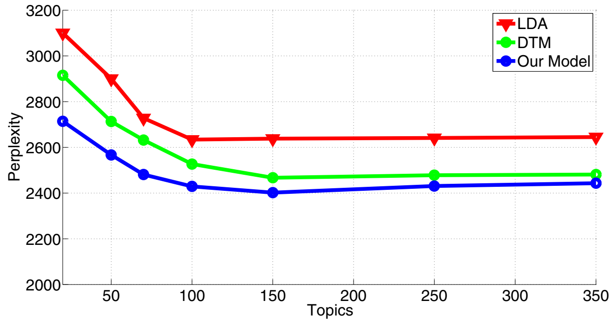

Figure 7: Perplexity comparison on ACL dataset.

NIPS dataset in which performance is relatively consistent over the recent three years, predictive performance on the ACL dataset differs significantly from year to year.

We run LDA, DTM and our model on both datasets while varying the number of topics, K. The results for the NIPS and the ACL datasets are shown in Figures 2 and 4, respectively. For each model, we only report its best performance. In addition, for both datasets, we also compare these models to the best performance achieved by the AR model. Note, as we mentioned before, LDA is unrealistic since $\boldsymbol{\beta}$ values for the test years are from test documents while in reality these values should be estimated from the past, assuming no documents observed in these test years. However, the purpose of showing the results from plain LDA is to demonstrate that the volume predictive performance can be greatly improved by treating topic probabilities as features if we can obtain them "correctly". For DTM and our model, these $\boldsymbol{\beta}$ values are estimated by the Kalman filter algorithm, mentioned in Section 3, which do not depend on the test documents at all. The first observation is that the overall performance is significantly improved over the AR model, in general. LDA is usually, but not always, better than AR in terms of average performance. For DTM and our model, which both consider temporal smoothing on topics, the performance is consistently better than both LDA and AR. Our model is also better than DTM on both datasets not only in terms of average performance but also in terms of performance on individual years.

In order to better understand the performance of topic models, we plot the performance on different K values averaged over the test years for the NIPS dataset in Figure 3. It is clear that performance is relatively stable compared to the AR model, where it is sensitive to the p value, shown in Table 2. However, for all models, as K increases, the performance slightly decreases, indicating that a higher value of K may lead models to over-fit. In any case, optimal performance is obtained from 50-70 topics for DTM and our model, which seems reasonable since NIPS is a relatively small research community and the topics are consistent over consecutive years. Similar conclusions can also be made for the ACL dataset.

Since DTM and our model prediction are performed on the year in which no documents are observed, it may be interesting to see whether performance would be improved if we partially observe the test documents. We pick the best K from the above experiments and feed a given fraction of test documents in a particular year to both models. The results are shown in Figure 5. As expected, performance improves on both datasets for both models if we observe partial data. However, when around 30% to 50% of test documents are observed, performance stabilizes.

6.2 Temporal Perplexity

Although we argue in Section 2 that perplexity may not be an appropriate evaluation method for temporal topic models, or for topic models in general, we still provide a comparison of perplexity between LDA, DTM and our model. Note, the performance on perplexity might be misleading because this measure is to evaluate how words in the documents can be assessed. Therefore, we perform the standard steps to calculate perplexity on documents in test years. As mentioned earlier, the real performance of these models should be considered when test documents are not available and how reliably the models can predict the response variables, not words. We show perplexity on the NIPS and ACL datasets in Figures 6 and 7, respectively. Overall, the perplexity values of DTM and our model are lower than LDA, for different K values, which confirms the observations in [3, 20]. In addition, perplexity decreases as K increases in general, indicating that a larger K may explain words better. However, the difference of perplexity between DTM and our model is relatively small, compared to the volume predictive performance. This is not unexpected because our model shares the same "generative" process for words in documents as DTM. Therefore, this observation also confirms that perplexity may not be appropriate to truly reflect the performance of different models, in terms of the tasks we care about. However, we do believe that a thorough study of the relationships of perplexity and the performance of third-party tasks for topic models is needed.

7. CONCLUSION

In this paper, we introduced a real-world task—tracking the volume of terms—to which temporal topic models can be applied. We proposed a new type of topic model incorporating the volumes of terms into the temporal dynamics of topics and directly optimize for the task. We combined state-space models and the volume of terms in a supervised learning fashion which enables us to effectively predict the volume in the future. The volumes of latent topics are by-products of our model, demonstrating the superiority of utilizing temporal topic models over traditional time-series tools (e.g., autoregressive models) to tackle this kind of problem. The proposed model can be further extended with arbitrary word-level features which are evolving over time. We presented the results of applying the model to two datasets with long time periods and showed its effectiveness over non-trivial baselines. Future work might include the adoption of recently developed online variational inference algorithms [8] to our model, enabling the processing of large scale datasets.

Acknowledgements

This material is based upon work supported in part by the National Science Foundation under Grant Numbers IIS-0545875 and IIS-0803605.

8. REFERENCES

[1] D. Agarwal and B.-C. Chen. fLDA: matrix factorization through Latent Dirichlet Allocation. In *Proceedings of the third ACM International Conference on Web Search and Data Mining (WSDM)*, pages 91–100, New York, NY, USA, 2010. ACM.

[2] A. Ahmed and E. P. Xing. 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 Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010.

[3] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006.

[4] D. M. Blei and J. D. Mcauliffe. Supervised topic models. In *Advances in Neural Information Processing Systems 21*, 2007.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] J. Boyd-Graber, J. Chang, S. Gerrish, C. Wang, and D. Blei. Reading Tea Leaves: How Humans Interpret Topic Models. In *Neural Information Processing Systems (NIPS)*, 2009.

[7] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 2004.

[8] M. Hoffman, D. Blei, and F. Bach. Online learning for latent dirichlet allocation. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 856–864, 2010.

[9] T. Iwata, T. Yamada, Y. Sakurai, and N. Ueda. Online multiscale dynamic topic models. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.

[10] N. Kawamae and R. Higashinaka. Trend detection model. In *Proceedings of the 19th International Conference on World Wide Web (WWW)*, pages 1129–1130, New York, NY, USA, 2010. ACM.

[11] S. Lacoste-Julien, F. Sha, and M. I. Jordan. DiscLDA: Discriminative Learning for Dimensionality Reduction and Classification. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.

[12] T. Masada, D. Fukagawa, A. Takasu, T. Hamada, Y. Shibata, and K. Oguri. Dynamic hyperparameter optimization for Bayesian topical trend analysis. In *Proceeding of the 18th ACM Conference on Information and Knowledge Management (CIKM)*, 2009.

[13] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *Proceedings of the 15th International Conference on World Wide Web (WWW)*, 2006.

[14] Q. Mei and C. Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proceedings of the eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2005.

[15] R. M. Nallapati, S. Ditmore, J. D. Lafferty, and K. Ung. Multiscale topic tomography. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.

[16] A. Neumaier and T. Schneider. Estimation of parameters and eigenmodes of multivariate autoregressive models. *ACM Transactions on Mathematical Software*, 27:27–57, March 2001.

[17] J. Petterson, A. Smola, T. Caetano, W. Buntine, and S. Narayanamurthy. Word features for latent dirichlet allocation. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1921–1929. 2010.

[18] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 248–256, Singapore, August 2009. Association for Computational Linguistics.

[19] C. Wang, D. Blei, and F.-F. Li. Simultaneous image classification and annotation. *IEEE Conference on Computer Vision and Pattern Recognition*, 0:1903–1910, 2009.

[20] C. Wang, D. M. Blei, and D. Heckerman. Continuous time dynamic topic models. In *Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence (UAI)*, 2008.

[21] X. Wang and A. McCallum. Topics over time: a non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2006.

[22] X. Wang, C. Zhai, X. Hu, and R. Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.

[23] X. Wang, K. Zhang, X. Jin, and D. Shen. Mining common topics from multiple asynchronous text streams. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining (WSDM)*, 2009.

[24] X. Wei and W. B. Croft. LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185, New York, NY, USA, 2006. ACM.

[25] J. Zhang, Y. Song, C. Zhang, and S. Liu. 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*, 2010.

[26] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: maximum margin supervised topic models for regression and classification. In *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, pages 1257–1264, New York, NY, USA, 2009. ACM.