On-Line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking

Loulwah AlSumait, Daniel Barbará, Carlotta Domeniconi
Department of Computer Science
George Mason University
Fairfax - VA, USA
lalsumai@gmu.edu, dbarbara@gmu.edu, carlotta@cs.gmu.edu

Abstract

This paper presents Online Topic Model *(OLDA), a topic model that automatically captures the thematic patterns and identifies emerging topics of text streams and their changes over time. Our approach allows the topic modeling framework, specifically the Latent Dirichlet Allocation (LDA) model, to work in an online fashion such that it incrementally builds an up-to-date model (mixture of topics per document and mixture of words per topic) when a new document (or a set of documents) appears. A solution based on the Empirical Bayes method is proposed. The idea is to incrementally update the current model according to the information inferred from the new stream of data with no need to access previous data. The dynamics of the proposed approach also provide an efficient mean to track the topics over time and detect the emerging topics in real time. Our method is evaluated both qualitatively and quantitatively using benchmark datasets. In our experiments, the OLDA has discovered interesting patterns by just analyzing a fraction of data at a time. Our tests also prove the ability of OLDA to align the topics across the epochs with which the evolution of the topics over time is captured. The OLDA is also comparable to, and sometimes better than, the original LDA in predicting the likelihood of unseen documents.*

1 Introduction

As electronic documents become available in streams over time, their content contains a strong temporal ordering. Considering the time information is essential to better understand the underlying topics and track their evolution and spread within their domain. In addition, instead of analyzing large collections of time-stamped text documents as archives in an off-line fashion, it is more practical for genuine applications to analyze, summarize, and categorize the stream of text data at the time of its arrival. For example, as news arrive in streams, organizing it as threads of relevant articles is more efficient and convenient. In addition, there is a great potential to rely on automated systems to track current topics of interest and identify emerging trends in online digital libraries and scientific literature. Identifying these stemming topics is essential for selecting and establishing state-of-the-art research projects and business entrepreneurships that would be attractive.

Probabilistic topic modeling is a relatively new approach that is being successfully applied to explore and predict the underlying structure of discrete data, such as text. A topic model, such as the Probabilistic Latent Semantic Indexing (PLSI) proposed by Hofmann [9], is a statistical generative model that relates documents and words through latent variables which represent the topics [14]. By considering a document as a mixture of topics, the model is able to generate the words in a document given the small set of latent variables (or topics). Inverting this process, i.e. fitting the generative model to the observed data (words in documents), corresponds to inferring the latent variables and, hence, learning the distributions of underlying topics.

Latent Dirichlet Allocation (LDA) [2] extends the generative model to achieve the capacity of generalizing the topic distributions so that the model can be used to generate unseen documents as well. LDA considers the topics to be multinomial distributions over the words, and assumes the documents to be sampled from a random mixtures of these topics. To complete its generative process for the documents, LDA considers Dirichlet priors for the document distributions over topics and the topic distributions over words.

This paper presents an online version of LDA that automatically captures the thematic patterns and identifies topics of text streams and their changes over time. Our approach allows LDA model to work in an online fashion such that it incrementally builds an up-to-date model (mixture of topics per document and mixture of words per topic) when a new document (or a set of documents) appears. A solution based on the Empirical Bayes method is proposed. The idea is to incrementally adjust the learned topics according to the dynamical changes in the data with no need to access the previously processed documents. This is achieved by sampling words in the newly arrived documents according to the distribution represented so far by the current model. The count of words in topics, resulted from running LDA at a time instance, is used to construct (weighted) priors at the following time instance. Thus, in our method, the new topic distributions will correspond to the previous realistic text structures.

Most of the related work either processes archives in an off-line fashion (e.g. [16]), post-discretizes the time ([17, 13]), or uses unconjugated priors to multinomial distributions and trained on all the previous data (e.g. [3, 15]). Our online topic model, however, makes use of the conjugacy property of the Dirichlet distribution to keep the model's structure simple, and to enable sequential inference. In addition, OLDA processes small subsets of data at a time which improve its memory usage and time complexity. The dynamics of our proposed approach provide a natural mean to solve the task of detecting emerging trends in text streams and tracking their drift over time. The idea is to use the inferred topic description to compute the similarities between the aligned topics across time and detect the topics that appear to be outliers. This approach has the added advantage that one could compute in real time when the topic emerges and when it ceases to be an outlier.

Our method is evaluated both qualitatively and quantitatively using benchmark datasets. The results are compared to the original LDA. We have found meaningful patterns in the discovered topics within the application domain. In addition, the OLDA model is able to align the topics across the epochs and, eventually, captures the evolution of the topics over time easily. The OLDA is also comparable to, and sometimes better than, the original LDA in predicting unseen documents as measured using perplexity.

The rest of the paper is organized as follows. Our Online LDA approach is introduced in Section 3, following a short review of the most related work in the literature (Section 2). In Section 4, we present the experiments we performed on NIPS and Reuters-21578 datasets and the results we obtained. Our final conclusions and suggestions for future work are discussed in Section 5.

2 Related Work

Considering time information for the task of identifying and tracking topics in time-stamped text data is the focus of recent studies (e.g. [4, 7, 10, 11]). Among other approaches, statistical modeling using versions of PLSI (e.g. [5]) and LDA (e.g. [16, 3, 6, 13, 12, 15]) have been deployed to solve this task.

In the probabilistic topic modeling that is based on LDA, the studies have examined latent topics and their changes across time in three main fold. The first, as in [16], had jointly modeled time and word co-occurrence with no Markov dependencies such that it treated time as an observed continuous variable. This approach, however, works offline, as the whole batch of documents is used once to construct the model. This feature does not suit the online setting where text streams continuously arrive with time.

In addition, many methods use post- or pre-discretized time analysis. The former involves fitting a topic model with no reference of time, and then ordering the documents in time, slicing them into discrete subsets, and examining the topic distributions in each time-slice. The work in [6] is one example of such approach. On the other hand, the pre-discretaized time analysis of topic modeling pre-divides the data into discrete time slices, and fits a separate topic model in each slice. Examples of this type include the experiments with the Group-Topic model [17] and the personal information dissemination behavior model [13]. Although our method discretize the time, it is distinguished by its ability of utilizing the newly acquired knowledge within the learning process and tracking the evolution of topics over time.

The work in [3], and most recently [15], have used a time series analysis to present a dynamic topic model (DTM) which explicitly models the evolution of topics with time by estimating the topic distribution at various time instances. To do so, the authors assume that the parameters are conditionally distributed by normal distributions with mean equal to the corresponding parameter at the previous time instance. However, since the normal distribution is not a conjugate to the multinomial distribution, the model does not yield a simple solution to the problems of inference and estimation. Finally, Multiscale Topic Tomography Model (MTTM) [12] is a sequential topic model which is the most relevant work to our approach. It uses conjugate priors using the Poisson distribution to model the generation of word-counts. Unlike our method, MTTM does assume the document streams to be of equal sizes.

3 Online LDA

First, before defining the online approach, we describe the statistical model of LDA [2] and the Gibbs sampling algorithm for inference in this model [6]. A glossary of notations used in the paper is given in Table 1.

LDA is a hierarchical Bayesian network that relates words and documents through latent topics. Since the words are observed, the document and the topic distributions, θ and ϕ, are conditionally independent. Furthermore, the doc-

Table 1. Notation used in the paper

SYMBOL	DESCRIPTION
D	total number of documents
K	number of topics
V	total number of unique words
δ	size of sliding window
N_d	number of word tokens in document d
S^t	a stream of documents arriving at time t
M^t	number of documents in S^t
V^t	number of unique words in S^t
N^t	number of word tokens in S^t
w^t_{di}	the unique word associated with the i^{th} token in document d at time t
z^t_i	the topic associated with w^t_{di}
θ^t_d	the multinomial distribution of topics specific to the document d at time t
ϕ^t_k	the multinomial distribution of words specific to the topic k at time t
α^t_d	K-vector of priors for document d at time t
β^t_k	V^t-vector of priors for topic k at time t
$\mathbf{B}^t_k$	$V^t \times \delta$ evolution matrix of topic k with columns $= \phi^i_k$, $i \in \{t-\delta, \cdots, t\}$
ω^δ	δ-vector of weights of ϕ^i, $i \in \{t-\delta, \cdots, t\}$

uments are not directly linked to the words. Rather, this relationship is governed by additional latent variables, z, introduced to represent the responsibility of a particular topic in using that word in the document, i.e. the topic(s) that the document is focused on. By introducing the Dirichlet priors α and β over the document and topic distributions, respectively, the generative model of LDA is complete and generalized to process unseen documents. LDA is based on the assumption of *exchangeability* for the words in a document and for the documents in a corpus.

The generative process of the topic model specifies a probabilistic sampling procedure that describe how words in documents can be generated based on the hidden topics. It can be described as follows:

1. Draw K multinomials ϕ_k from a Dirichlet prior β, one for each topic k;

2. Draw D multinomials θ_d from a Dirichlet prior α, one for each document d;

3. For all documents, d, in the corpus, then for all words, w_{di}, in the document:

(a) Draw a topic z_i from multinomial θ_d; $(p(z_i|\alpha))$

(b) Draw a word w_i from multinomial ϕ_z; $(p(w_i|z_i, \beta))$

Because an exact approach to estimate ϕ is intractable, sophisticated approximations are usually used. Griffiths and Steyvers in [6] proposed Gibbs sampling as a simple and effective strategy for estimating ϕ and θ. Under Gibbs sampling, ϕ and θ are not explicitly estimated. Instead, the posterior distribution over the assignments of words to topics, $P(\mathbf{z}|\mathbf{w})$, is approximated by means of the Monte Carlo algorithm which iterates over each word token in the text collection and estimates the probability of assigning the current word token to each topic $(P(z_i = j))$, conditioned on the topic assignments to all other word tokens $(\mathbf{z}_{\neg i})$ as follows [6]:

$$P(z_i = j|\mathbf{z}_{\neg i}, w_{di}, \boldsymbol{\alpha}, \boldsymbol{\beta}) \propto \frac{C^{VK}_{w_{\neg i},j} + \beta_{w_i,j}}{\sum_{v=1}^{V}(C^{VK}_{v_{\neg i},j} + \beta_{v,j})} \times \frac{C^{DK}_{d_{\neg i},j} + \alpha_{d,j}}{\sum_{k=1}^{K}(C^{DK}_{d_{\neg i},k} + \alpha_{d,k})}$$

where $C^{VK}_{w_{\neg i},j}$ is the number of times word w is assigned to topic j, not including the current token instance i; and $C^{DK}_{d_{\neg i},j}$ is the number of times topic j is assigned to some word token in document d, not including the current instance i. From this distribution, a topic is sampled and stored as the new topic assignment for this word token. After a sufficient number of sampling iterations, the approximated posterior can be used to get estimates of ϕ and θ by examining the counts of word assignments to topics and topic occurrences in documents .

To enable LDA to work in an on-line fashion on data streams, OLDA model considers the temporal ordering information and assumes that the documents are divided in time slices. At each time slice, a topic model with K components is used to model the newly arrived documents. The generated model, at a given time, is used as a prior for LDA at the successive time slice, when a new data stream is available for processing. The hyper-parameters β can be interpreted as the prior observation counts on the number of times words are sampled from a topic before any word from the corpus is observed ([14], [1]). So, the count of words in topics, resulted from running LDA on documents received at time t, can be used as the priors for the $t+1$ stream.

Our approach allows many alternatives for keeping track of history at any time t, ranging from a full memory that keeps track of the complete history to a short memory that keeps the counts of the model associated with time $t-1$ only. Such variety of solutions suits the structure of text repositories, since the flow and nature of document streams differ according to the type of the corpus and, consequently, the role of history would be different too. While the current experiments will demonstrate some of these differences, it is part of our future work to investigate the role of history in inferring future semantics.

3.1 Generative Process and Approximate Inference

To formulate the problem, we first assume that documents arrive in ascending order of their publication date.

After each time slice, t, of a predetermined size ε, e.g. an hour, a day, or a year, a stream of documents, $S^t = \{d_1,\cdots,d_{M^t}\}$, of variable size, M^t, is received and ready to be processed. The size of the time slice, ε, depends on the nature of the corpus on which the model is applied, and on how fine or coarse the resulted description of data is expected to be. The indices of the documents within a stream, S^t, preserve the order by which the documents were received during the time slice t, i.e. d_1 is the first document to arrive and d_{M^t} is the latest document in the stream. A document d received at time t is represented as a vector of word tokens, $\mathbf{w}_d^t = \{w_{d1}^t, \cdots, w_{dN_d}^t\}$. It is naturally the case that stream S^t introduces new word(s) in the vocabulary. These words are assumed to have 0 count in ϕ for all topics in previous streams. This assumption is important to simplify the definition of matrix $\mathbf{B}$ and the related computation.

Let $\mathbf{B}_k^{t-1}$ denotes an *evolutionary matrix* of topic k in which the columns are the word-topic counts ϕ_k^j, generated for streams received within the time specified by the sliding window, i.e. $j \in \{t-\delta-1, \cdots, t-1\}$. Let ω^δ be a vector of δ weights each of which is associated with a time slice from the past to determine its contribution in computing the priors for stream S^t. We assume that the weights in ω^{t-1} sum to one. Hence, the parameters of a topic k at time t are determined by a weighted mixture of the topic's past distributions as follows:

$$\beta_k^t = \mathbf{B}_k^{t-1}\omega^\delta \qquad (1)$$

Computing the β's in this manner ties the topic distributions in the consecutive models and captures the evolution of topics in a sequential corpus. Thus, the generative model for time slice t of the proposed online LDA model is given as follows:

```
1. For each topic $k = 1, \cdots, K$
  2. Compute $\beta_k^t = \mathbf{B}_k^{t-1}\omega^\delta$
  3. Draw $\phi_k^t \sim \text{Dir}(\cdot|\beta_k^t)$
4. For each document, $d$,
  5. Draw $\theta_d^t \sim \text{Dir}(\cdot|\alpha^t)$
  6. For each word token, $w_i$, in document $d$
    7. Draw $z_i$ from multinomial $\theta_d^t$; $(p(z_i|\alpha^t))$
    8. Draw $w_i$ from multinomial $\phi_{z_i}$; $p(w_i|z_i, \beta_{z_i}^t)$
```

At time slice $= 1$, the topic parameters, ϕ_k^1, are drawn from a Dirichlet prior, $\text{Dir}(\cdot|\beta_k^1)$, where β_k^1 is initialized to some constant, b, as done in the original LDA modeling, e.g. [6].

Maintaining the models' priors as Dirichlet is essentially useful to simplify the inference problem by making use of the conjugancy property of Dirichlet and multinomial distributions. In fact, by tracking the history as prior patterns, the data likelihood and, hence, the posterior inference in the static LDA are left the same, and applying them to our proposed model is a straightforward. The main difference between the two approaches in this regard is that the inference problem in our online approach is solved by using chunks of the data instead of the whole set. This makes the time complexity and memory usage of OLDA efficient and applicable for genuine applications. Our model uses Gibbs sampling as an approximate inference method to estimate the word-topic assignments. The conjugacy property of our priors makes the application of the sampling method in our approach very easy.

3.2 Topic Detection and Tracking

The dynamics of our proposed approach provide a natural mean to capture the topics and their evolution over time. By constructing the priors as a weighted combination of the history, the topics are tied and automatically aligned across time. The matrix $\mathbf{B}_k^t$ can be considered as the evolution of topic k in which the topic development over time is captured. Furthermore, novel concepts or topics can also be identified. We define a novel topic as the one which, when appears, is "different" from the previous (or current) concepts, i.e. is an outlier, and with time it becomes "mainstream" and, hence, ceases to be an outlier.

After applying the topic modeling, a topic is represented as a vector of probabilities over the space of words. The dissimilarity between two topic distributions, p and q, can be computed in such a space using the Kullback Leibler (KL) divergence. The KL divergence $\text{KL}(p \parallel q)$ represents the average additional amount of bits required to encode samples from p with a code based on q [1], and is given by

$$\text{KL}(p \parallel q) = \sum_i p(i) \log \frac{p(i)}{q(i)}$$

KL divergence is not a real metric, since it is not symmetric. Thus, in our work, we compute the average of $\text{KL}(p \parallel q)$ and $\text{KL}(q \parallel p)$ and denote it KL distance or D_{KL} in the rest of the paper.

An emerging topic can be viewed as the one that is different from its peers in the same stream, or from all the topics seen so far. To quantify the difference, we define a $\delta \times K$ distance matrix $Dist$ where each entry, $Dist(t, k)$, is the D_{KL} between the distributions of topic k at time t and $t+1$. Let CL be a confidence level, and $perc^t$ be the percentile, the value below which a CL percent of distances computed at time t fall. The identification of emerging topics can be modeled by considering different approaches to compute the percentile at time t: either to consider the K topic distances computed at time t (current percentile - $perc^t$) , or to use all the $\delta \times K$ distances computed so far (historic percentile - $percALL^t$). Then, if the KL distance of a topic, ϕ_k^t, from the one that immediately precedes it, ϕ_k^{t-1}, exceeds the percentile value, $perc^t$ ($percALL^t$, respectively), the topic is flagged as a *nominated emerging* *topic*. Thus, given the evolution matrices, $\mathbf{B}^t$, the emerging topic detection algorithm (Edetect) at time t can be formulated as follows:

```
1. $Etopics = \emptyset$; $EtopicsALL = \emptyset$;
2. For each previous time slice, $j = 2$ to $\delta$
  3. For each topic, $k = 1$ to $K$
    4.Compute KL distance,
    $Dist(j-1,k) = D_{\mathrm{KL}}(\mathbf{B}_k^t(:,j) \parallel \mathbf{B}_k^t(:,j-1))$
5. Compute $perc^t$ = percentile($Dist(\delta-1,:)$, $CL$);
6. Compute $percALL^t$ = percentile($Dist, CL$);
7. For each topic, $k = 1$ to $K$
  8. If $Dist(\delta-1,k) > perc^t$
    9. $Etopics = Etopics \cup k$
  10. If $Dist(\delta-1,k) > percALL^t$
    11. $EtopicsALL = EtopicsALL \cup k$
```

Thus, the algorithm returns the topics that are flagged as emerging topics in stream S^t. Note that the distances in $Dist$ need not to be recomputed at every time slice and can be constructed incrementally to reduce time complexity.

3.3 OLDA Algorithm

An overview of the proposed Online LDA algorithm is shown in Algorithm 1. In addition to the text streams, S^t, the algorithm takes as input the CL confidence level, the weight vector ω, and fixed Dirichlet values, a and b, for initializing the priors α and β, respectively, at time slice 1. Note that b is also used to set the priors of new words that appear for the first time in any time slice. If N_{stream} denotes the number of streams processed, the output of the algorithm will be: N_{stream} generative models, the evolution matrices $\mathbf{B}_k$ for all topics, and lists of nominated emerging topics, one for each stream.

Algorithm 1 Online LDA

```
INPUT: $b; a; CL; \omega^\delta; S^t, t \in \{1, \cdots, N_{stream}\}$
for $t = 1$ to $N_{stream}$ do
    if $t = 1$ then
        $\beta_k^t = b, k \in \{1, \cdots, K\}$
    else
        $\beta_k^t = \mathbf{B}_k^{t-1}\omega^\delta, k \in \{1, \cdots, K\}$
    end if
    $\alpha_d^t = a, d = 1, \cdots, M^t$
    initialize $\phi^t$ and $\theta^t$ to zeros
    initialize topic assignment, $\mathbf{z}^t$, randomly for all word tokens in $S^t$
    [$\phi^t$,$\theta^t$,$\mathbf{z}^t$] = GibbsSampling($S^t, \beta^t, \alpha^t$)
    $\mathbf{B}_k^t = \mathbf{B}_k^{t-1} \cup \phi_k^t, k \in \{1, \cdots, K\}$
    if $t > 1$ then
        $[Etopics(t), EtopicsA(t)] = Edetect(CL)$
    end if
end for
```

4 Experimental Results

Online LDA (OLDA) is evaluated in three problem domains: document modeling, document classification, and emerging topic detection. The performance of the proposed method is compared to the standard version of LDA. OLDA is trained on the individual stream arriving at each time t, while the original LDA, named LDA-upto, is trained on all the streams received up to time t. Both models were run for 500 iterations and the last sample of the Gibbs sampler was used for evaluation. The number of topics, K, is fixed across all the streams. Following the settings in [2, 6], K, a, and b are set to 50, $50/K$, and 0.1. For now, β_k^t depends on the topic distribution of the previous stream only, i.e. $\delta = 1$. Using different weight settings for ω, three variants of OLDA are considered specifically for the document modeling problem. The standard version of our approach, which we call OLDA, uses the actual counts of the previous model to compute the priors.The second model, namely OLDA-fixed, ignores the history and processes the text stream using fixed symmetric Dirichlet prior. In the last version, named OLDA-norm, the counts are normalized between zero and one before being used. All experiments are run on 2GHz Pentium(R) M-processor laptop using "Matlab Topic Modeling Toolbox", authored by Mark Steyvers and Tom Griffiths[1].

4.1 Datasets

The following is a short description of the datasets used in our experiments.

Reuters-21578[2]. The corpus consists of newswire articles classified by topic and ordered by their date of issue. There are 90 categories with some articles classified in multiple topics. The ApteMod version of this database has been used in many papers. This version consists of 12,902 documents, with approximately 27,000 features in total.

For our experiments, only articles with at least one topic were kept for processing. For data preprocessing, words were only down-cased and stemmed to their root source. The resulting dataset consists of 10337 documents, 12112 unique words, and a total of 793936 word tokens. For simplicity, we partitioned the data into 30 slices and considered each slice as a stream.

NIPS dataset[3]. The NIPS set consists of the full text of the 13 years of proceedings from 1988 to 2000 Neural Information Processing Systems (NIPS) Conferences.

[1] The Topic Modeling Toolbox is available at: psiexp.ss.uci.edu/research/programs_data/toolbox.htm

[2] The original dataset is available to download from the UCI Knowledge Discovery in Databases Archive. http://archive.ics.uci.edu/ml/

[3] The original dataset is available at the NIPS Online Repository. http://nips.djvuzone.org/txt.html.

The data was preprocessed for down-casing, removing stop-words and numbers, and removing the words appearing less than five times in the corpus. The data set contains 1,740 research papers, 13,649 unique words, and 2,301,375 word tokens in total. Each document has a timestamp that is determined by the year of the proceedings. Thus, the set consisted of 13 streams in total. The size of the streams, M^t, varies from 90 to 250 documents.

4.2 Document Modeling

The objective of document modeling is a density estimation that describes the underlying structure of data. One common approach to measure this is by evaluating the model's generalization performance on previously unseen documents. *Perplexity* is a canonical measure of goodness that is used in language modeling to measure the likelihood of a held-out test data to be generated from the underlying (learned) distributions of the model [8]. The higher the likelihood is, the lower the perplexity will be, and, hence, better generalization performance can be achieved. Formally, for a test set of M documents, the perplexity is [2]:

$$perplexity(D_{test}) = \exp\left\{-\frac{\sum_{d=1}^{M} \log p(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d}\right\} \qquad (2)$$

To compute $p(\mathbf{w}_d)$, several iterations of "query sampling" must be performed to get the document-topic counts of the unseen document which are required to compute the likelihood (refer to [8] for details).

We trained the three versions of OLDA and the LDA-upto topic models on the NIPS and Reuters datasets. At every time slice, we compare their perplexity performance. Figures 1 and 2 illustrate the perplexity of the models trained on Reuters and NIPS, respectively. OLDA improved the document modeling in Reuters with respect to the LDA baseline. As for Online models with normalized or fixed priors, the performance is reversed. This shows that information propagated from the past is very useful to predict future streams in Reuters.

However, testing with NIPS showed a different behavior. OLDA with normalized priors performed better on the test data. LDA framework, in general, is a statistically data-dependent approach. So, the role of history would, eventually, vary according to the homogeneity of the domain. This justifies the importance of the weight matrix ω.

In addition, when we tested our model on the training data for NIPS, the perplexity noticeably decreased. Because our approach has more parameters and they are set according to the information propagated from previous streams, the online model results in better fitting of the data processed so far rather than predicting future documents. This result matches similar findings in the literature [12] and satisfies the objective for which our model is applicable.

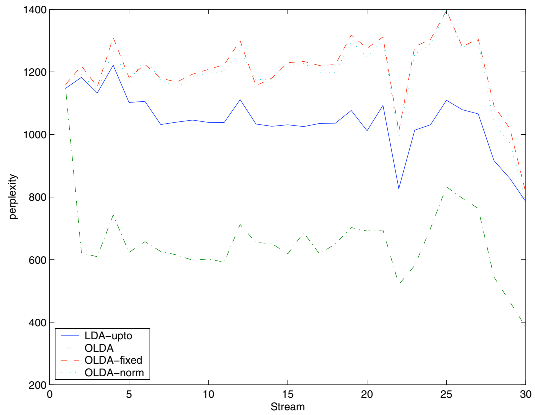

Figure 1. Comparisons of Perplexities of OLDA and LDAupto trained on Reuters

To verify the ability of our model of visualizing the data, we analyzed the posteriors of OLDA estimated from Reuters and NIPS corpora. Due to lack of space, only two examples from NIPS and Reuters are listed in Table 2 and 3 respectively.

Like the standard LDA, our method is able to identify meaningful topics in NIPS such as classification, speech recognition, Bayesian learning, and regression. The topics discovered in Reuters at every stream fit well with the categories that the articles belong to. Yet, OLDA is able to find these topics with no access to the entire data. Rather, the model is generated from a small fraction of documents, which makes our model superior in terms of time and memory efficiency. Figure 3 compares the execution time required for OLDA and the standard LDA to generate the topic model at every time instance for Reuters. It can be seen that OLDA requires approximately a constant time, depending on the size of each stream, while the run time required by LDA-upto to analyze the data is accumulative. In addition, LDA requires the whole data to be stored for future processing, however, our model stores only a metadata of the data in terms of a small number of generative models.

Furthermore, OLDA is able to identify more fine topics that may be represented by a small number of documents at a certain point of time. For example, in NIPS, the topic "Support Vector Machine" (SVM) appeared in three documents in year 1995, in two documents in year 1998, in six documents in 1999, and in 9 documents in 2000.

Table 4 lists the "SVM" documents that appeared in 1995 and the "SVM" topic distribution over words generated by LDAupto, OLDA, and OLDA-norm. The table also shows the weight of the topic "SVM" in the document distribution generated by the models. The number between brackets represents the rank of the topic "SVM" in the doc-

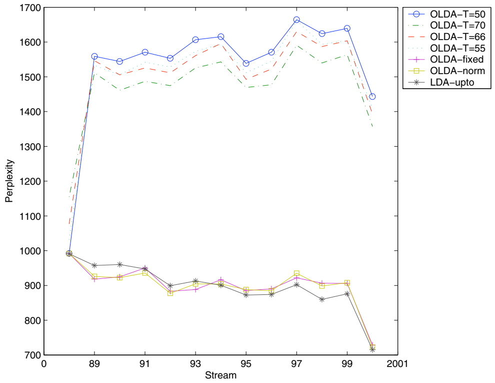

Figure 2. Comparisons of Perplexities of OLDA run on different settings of K and LDAupto trained on NIPS

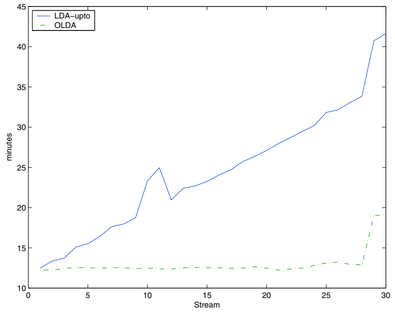

Figure 3. Comparisons of run time required to train OLDA and LDAupto models on NIPS

ument, e.g. rank (1) means the topic has the highest weight in the corresponding document. LDA-upto was not able to detect "SVM" as a distinguished topic, so we report four topics that had the highest weights in "SVM" documents. On the other hand, both the OLDA models were able to detect the topic and assign high weight for it in the documents' distributions. The same observation is found in the 1998 and 1999 models. LDAupto was only able to detect "SVM" in the year 2000.

Many of the topics discovered, like "reinforcement learning" in Table 2, have a strong and constant identity over the years, while other topics were a mixture of either meaningful themes, like topic 14 which is a mixture of "SVM" and "character recognition", or "junk" topics that are holding words like "abstract", "figure", "introduction" and so on. Our intuition is that the number and size of relevant documents is an important factor. By examining the distribution of topic SVM from year 1998 to 2000, it can be clearly seen how SVM related words are dominating over character recognition terms as the number of SVM articles increases. In addition, the setting of the number of components, K, has a major impact too. On inspection, we tested OLDA with different settings for K (see Figure 2). However, detailed analysis of the effect of the number of component is part of our future work.

Table 2. Examples of topics estimated by OLDA from NIPS corpus and its evolution over 13 years

Topic 12: Reinforcement Learning
88: state learning system states time cycles recurrence failure weight algorithm
89: node system state rule learning nodes tin match transition
90: state learning rule system node algorithm change rules controller dynamic
91: learning state reinforcement system world time adaptive planning controller
92: state learning action task exploration tasks sutton elemental
93: learning state reinforcement control time action task optimal based
94: learning state optimal control dynamic policy action time adaptive
95: learning state optimal action policy control reinforcement grid dynamic
96: learning state action policy reinforcement algorithm optimal time
97: learning state action reinforcement time policy optimal algorithm dynamic
98: state learning policy reinforcement optimal time action step control
99: state learning policy action reinforcement optimal rl time
2000: state learning policy action reward time reinforcement belief
Topic 14: Character Recognition - SVM
88: input error vector classifier method classification connection problem
89: input vector classification limited feature characters figure error
90: feature vector large number digits input scale cun parameters local
91: feature vector input large category classification characters error recognition
92: recognition risk character digit feature input vectors digits cun
93: character distance recognition characters rate error segmentation large field handwritten
94: distance recognition character address feature handwriting lines text pen
95: style recognition support content vectors distance feature character database
96: distance tangent recognition machine character simard digit prototype vectors
97: recognition character window distance handwritten machine digit dimensionality ocr
98: recognition distance kernels character machine kernel sv segmented support
99: kernel support recognition vector svm digit machines kernels rotation
2000: kernel support vector svm machines svms kernels feature recognition

Table 3. Examples of topics estimated by OLDA from Reuters corpus and its evolution over the first 10 streams

TOPIC 6: Gold
1: pct interest expect hold rmj gold secur ounc plc agenc
2: interest pct gold expect plc secur hold agenc volum western
3: pct gold interest hold expect agenc given british made ounc
4: hold gold pct land mine agenc given state interest expect
5: ton made agenc pct expect interest state mine north gold
6: reserv gold ton ounc mine ltd agenc silver expect averag
7: gold coin reserv ltd ounc properti ventur immedi develop interest
8: gold ventur or reserv copper develop mine western ltd coin
9: gold copper ton averag ounc mine ltd feet assai ventur
10: reserv averag gold ounc ventur mine ltd ton pct earlier
TOPIC 28: Crude
1: reuter export industri mine produc tonn plan quota output tin
2: industri export reuter produc minist countri accord tonn told miner
3: state industri reuter told member minist output mine accord onli
4: oil opec bpd crude state offici accord industri told output
5: oil state barrel crude minist ecuador offici reuter export output
6: oil barrel crude opec energi minist export ecuador output member
7: oil barrel crude bpd refineri opec minist petroleum state output
8: oil opec crude bpd barrel arabia saudi energi nazer ecuador
9: oil crude energi minist barrel dai gas countri petroleum offici
10: oil barrel opec crude relief revenu energi dai field develop

Table 4. The topic “SVM” (distribution & documents) from NIPS in LDA, OLDA, and OLDA-norm at year 1995. The top lists the weight and rank of the topic SVM for each document. The bottom list the distribution of the topic from each model

Document Title	LDA	OLDA	OLDA-norm
Support Vector Regression Machines	-	$0.1(2)$	$0.41(1)$
Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing	-	$0.13(2)$	$0.54(1)$
Improving the Accuracy and Speed of Support Vector Machines	-	$0.26(1)$	$0.39(1)$

Model	**Topic Distribution**
LDA	- problem, space, points, solution, regions, number, solutions, set, find, approach, boundary, large, solve, method, constraints, maximum, dimensional - function, approximation, optimal, basis, linear, order, form, general, case, ai, process, variable, continuous, theory, section, equation, degree - training, error, set, data, test, prediction, performance, sets, number, examples, validation, experiments, problem, size, generalization - data, estimate, regression, method, variance, bias, based, sample, statistical, neural, selection, true, samples, criterion, fit, risk
OLDA	style, recognition, support, content, vectors, distance, feature, character, vector, database, vapnik, error, accuracy, speed, lines, sv, bilinear
OLDA-norm	function, basis, vector, space, support, feature, kernels, regression, set, radial, smoothing, regularization, estimation, method, equivalent, vapnik, dimensional

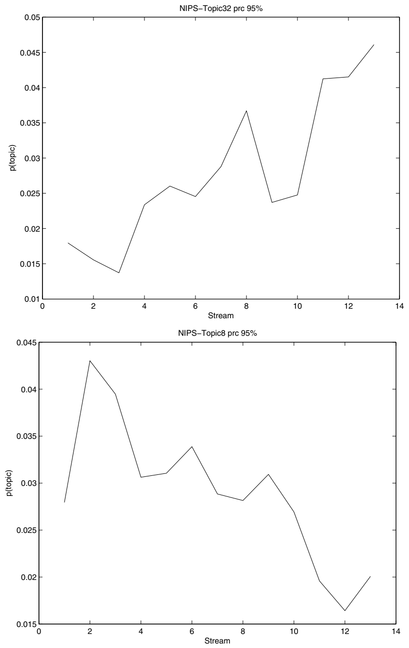

Figure 4. Tracking topics in NIPS over 13 years. Top: topic 32 (Bayesian Learning). Bottom: topic 8 (Multilayer neural networks - supervised learning/gradient descent)

It is also interesting to track the popularity of a topic as a function of time. This can be easily done by examining the topic evolution matrices. Figure 4 illustrates the popularity of two topics, the “Bayesian learning” and “multilayer neural networks” (NN), in terms of topic probability at each year. The first topic is clearly gaining more interest in the literature while the topic “NN” is declining.

4.2.1 Document Classification

The distribution of a document over topics can be considered a reduced description of the document in a new space spanned by the small set of latent variables [2]. So, the performance of the topic model can be evaluated by investigating the amount of discriminative information that is preserved in the document distributions. One way to do this is by solving a classification problem. For evaluation, classification accuracy and F1 measure are common measures.

We conducted a two class-classification problem using the Reuters dataset. At each time slice, OLDA and LDAupto models were trained without using the true class labels with K set to 50, as in [2]. Then, the document distributions, θ_d^t, are used to train a Support Vector Machine (SVM) to classify the “earn” class[4]. SVM was run five times using different $20 - 80\%$ partitions of train-test sets. The average F1 at every time stream for both models is given in Figure 5, and the performance in terms of classification accuracy averaged over all the streams is given in Table 5. While trained on a small subset of the corpus, our approach is able to generate a model that is as descriptive as the one generated using the whole data. In fact, the low F1 obtained with OLDA were due to the random partitioning that resulted in test sets that do not include any positive example.

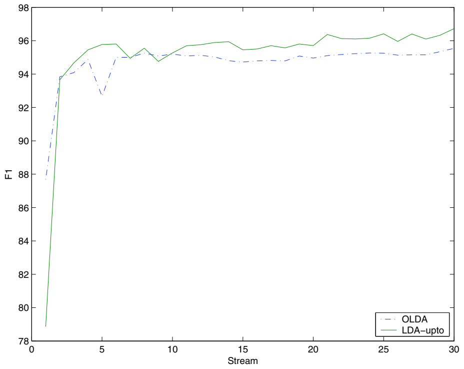

Figure 5. Average F1 of OLDA and LDAupto trained on Reuters

4.2.2 Emerging Topic Detection

The objective of this set of experiments is to test the ability of our method to detect novel topics at the time of their

[4] SVMLight software package is used for our experiments. It is available at: http://svmlight.joachims.org/.

Table 5. The average, minimum, maximum, and standard deviation of the classification accuracy over all streams of Reuters corpus

Model	accuracy	min accuracy	max accuracy	STDEV
OLDA	94.67	87.69	95.54	1.43
LDAupto	95.15	78.86	96.73	3.14

arrival. We test the emerging topic detection at two confidence levels: 90% and 95%. We applied the emerging topic detection method on NIPS and a number of topics were flagged at each year. For example, Topic "SVM" was detected at year 1999 at both confidence levels. Figure 6 illustrates the distance and probability of topic "SVM" with CL set to 90%. The year at which the topic distance exceeds the historic (current) percentile is marked by $\#$ $(*)$. Because the number of components K is fixed, an emerging topic appears first with a small and/or similar topic. For example, when "SVM" first appeared in 1995, it shared topic 14 with the topic "character/digit recognition".

In year 2000, more "SVM" documents are received and, hence, the probability of the topic sharply increased while the distance from the topic distribution at year 1999 decreased. Thus, the topic ceases from being an emerging topic and the algorithm does not consider it novel anymore. The reason why "SVM" was not detected in the year 1995 could be related to the number of documents, i.e. number of tokens, that are associated to the topic compared to other topics. As can be seen in Figure 6, the probability of the topic at that year, stream $=$ 8, is very low. To address this behavior, we are working on a "weighted KL distance" which is invariant with respect to the number of tokens associated to a topic.

Another set of synthetic experiments is performed on Reuters data. The documents of two classes, "crude" and "coffee", were held out for some number of streams. Then, at the forth (seventh) time slice, the documents of "crude" ("coffee") were released. Our emerging topic detection was able to detect both topics as emerging topics at the time of their release for both the current and historic distances.

Table 3 lists the distribution of Topic 28 before and after releasing the "crude" documents while Table 6 illustrates the output of our method for CL= 95% using the historic percentile. The topic 28 (18) in stream 4 (7) clearly corresponds to the "crude" ("coffee") documents that were released at that time. The topic "crude", though, appeared again as a new emerging topic, at stream 10 for example (see Table 6). The words "opec", "relief", "revenue", and "development" are clearly the cause of the flag. These words indicate new news regarding some relief/development efforts of the Opec.

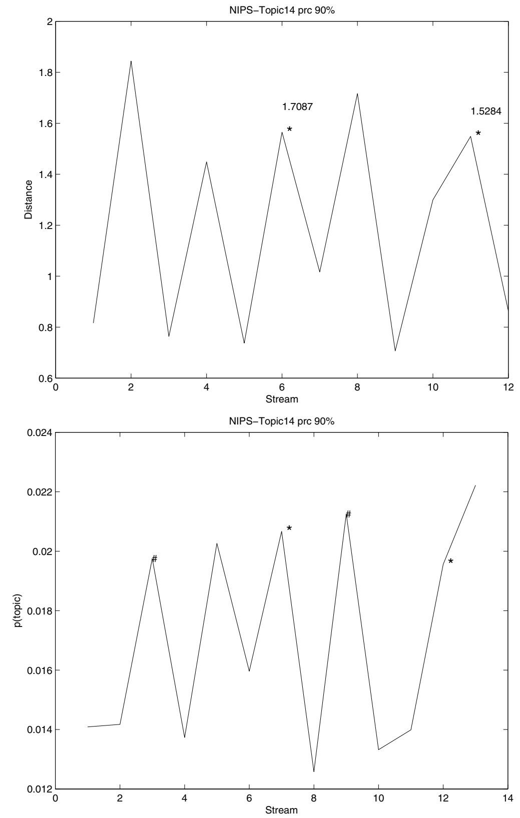

Figure 6. Distance and probability of Topic 14 (Character recognition - SVM) over 13 years. The topic is flagged as emerging topic at years 1990 and 1996 with historic percentile($\#$) and at years 1994 and 1999 with current percentile $(*)$. The confidence level is 90%

Table 6. The output of the Emerging Topic Detection on Reuters corpus. The crude topic (28) is detected at stream 4 and the coffee topic (18) is detected at stream 7. The distributions, probability, percentage of documents of both the past and the new topics are listed

```
STREAM 4 DOCUMENTS 340 WORDS 12112 TOKENS 23603 Perc= 1.8986
TOPIC 28
(Past) %doc 27.3 p(topic) 0.019:
   state industri reuter told member minist output mine accord onli
(Current) %doc 29.412 p(topic) 0.0276:
   oil opec bpd crude state offici accord industri told output
```

```
STREAM 7 DOCUMENTS 339 WORDS 12112 TOKENS 23331 Perc= 1.7182
TOPIC 18
(Past) %doc 35.0 p(topic) 0.02:
   total mai bought between rais harvest reuter accord sinc maiz
(Current) %doc 38.643 p(topic) 0.0234:
   export total quota bag coffe brazil reuter mai bought between
```

```
STREAM 10 DOCUMENTS 335 WORDS 12112 TOKENS 24677 Perc=
1.6763
TOPIC 28
(Past) %doc 25.373 p(topic) 0.0161:
   oil crude energi minist barrel dai ga countri petroleum offici
(Current) %doc 26.866 p(topic) 0.0263:
   oil barrel opec crude relief revenu energi dai field develop
```

5 Conclusions

We have developed an online topic model for discrete data to model the temporal evolution of topics in data streams. Our approach is a non-Markov on-line LDA Gibbs sampler topic model (OLDA) in which the current model, along with the new data, guide the learning of a new generative process that reflects the dynamic changes in the data. This is achieved by using the generated model, at a given time, as a prior for LDA at the successive time slice, when a new data stream becomes available for processing.

The weight of history in the generative process can be controlled by the weight matrix depending on the homogeneity of the domain. Our model results in an evolutionary matrix for each topic in which the evolution of the topic over time is captured. In addition, we proposed an algorithm to detect emerging topics based on the framework of OLDA.

By processing small subsets of documents only, OLDA is able to learn meaningful topics, similar and in some cases better than the LDA baseline. Our method also outperforms LDA in detecting topics represented by a small set of documents at a certain point in time.

The proposed approach can be extended in many directions. Examining different settings for the weight matrix is part of our future work to investigate its effect on the learned models. We are also considering the use of prior-knowledge to learn (or enhance the construction of) the parameters. In addition, different alternatives are considered for the distance metric used to compute the dissimilarities between topic distributions. We plan to construct a weighted distance metric that "normalizes" the document size and distinguishes between inter-topic differences and intra-topic drifts.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[3] D. M. Blei and J. D. Lafferty, "Dynamic topic models," *In International conference on Machine learning*, pp. 113-120, 2006.

[4] B. Cao, D. Shen, J. Sun, X. Wang, Q. Yang, and Z. Chen, "Detect and Track Latent Factors with Online Nonnegative Matrix Factorization," *the 12th International Joint Conference on Artificial Inteligence*, pp. 2689–2694, 2007.

[5] T. Chou and M. Ch. Chen, "Using Incremental PLSI for Threshold-Resilient Online Event Analysis," *the IEEE Transactions On Knowledge And Data Engineering*, vol. 20, no. 3, 2008.

[6] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceeding of the National Academy of Sciences*, pp. 5228–5235, 2004.

[7] R. Guha, R. Kumar, D. Sivakumar, and S. Jose, "Unweaving a Web of Documents," *Proceeding of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2005

[8] G. Heinrich, "Parameter estimation for text analysis," *Arbylon publications*. Retrieved from: http://www.arbylon.net/publications/, Retrieved date: 12/9/2007, Aug 2005.

[9] T. Hofmann, "Probablistic Latent Semantic Indexing," *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, 1999.

[10] J. Kleinberg, "Bursty and hierarchical structure in streams," *Proceeding of the 8th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, 2002.

[11] Q. Mei and C. Zhai, "Discovering evolutionary theme patterns from text: an exploration of temporal text mining," *Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, 2005.

[12] R. Nallapati, S. Ditmore, J.D. Lafferty, and K. Ung, "Multiscale topic tomography," *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery in data mining*, 2007.

[13] X. Song, C.-Y. Lin, B. L. Tseng, and M.-T. Sun, "Modeling and predicting personal information dissemination behavior," *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005.

[14] M. Steyvers and T. L. Griffiths, "Probabilistic Topic Models," In T. Landauer, D McNamara, S. Dennis, and W. Kintsch (ed), *Latent Semantic Analysis: A Road to Meaning*, Laurence Erlbaum, 2005.

[15] C. Wang, D. Blei, and D. Heckerman, "Continuous time dynamic topic models," *The 23rd Conference on Uncertainty in Artificial Intelligence*, 2008.

[16] X. Wang and A. McCallum, "Topics over Time: A Non-Markov Continuous-Time Model of Topical Trends," *ACM SIGKDD international conference on Knowledge discovery in data mining*, 2006.

[17] X. Wang, N. Mohanty, and A. McCallum, "Group and topic discovery from relations and text," *The 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Workshop on Link Discovery: Issues, Approaches and Applications*, pp. 28-35, 2005.