Technical Report CISE, University of Florida (2011) 1-13 Submitted 09/12; ID #520

Topic Learning and Inference Using Dirichlet Allocation Product Partition Models and Hybrid Metropolis Search

Clint P. George CGEORGE@CISE.UFL.EDU
Taylor C. Glenn TCG@CISE.UFL.EDU
Joseph N. Wilson JNW@CISE.UFL.EDU
Paul D. Gader PGADER@CISE.UFL.EDU
Department of Computer and Information Science & Engg
University of Florida
Gainesville, FL 32611

Claudio Fuentes CFUENTES@STAT.UFL.EDU
Vikneshwaran Gopal VIKNESH@STAT.UFL.EDU
George Casella CASELLA@STAT.UFL.EDU
Department of Statistics
University of Florida
Gainesville, FL 32611

Abstract

This paper describes methods for online topic learning and inference based on a product partition model alternative to the Latent Dirichlet Allocation (LDA) topic model. Topic inference for newly encountered documents is performed using a Metropolis search with a learned model without retraining the parameters on the entire training corpus. We also describe how hybrid random walk and simulated annealing methods help the topic search. Our online topic learning method is based on the full Gibbs sampler [1] developed for Dirichlet Allocation using product partition models with a prior on the topic multinomials.

Keywords: Product partition models, online learning, Metropolis search, topic modeling

1. Introduction

Latent Dirichlet Allocation (LDA) [2], a probabilistic topic model, is conventionally used for modeling documents. It assumes that corpus-documents are generated from a mixture of topics, which are defined as probabilistic distributions over the vocabulary of words. Topic models have applications in dimensionality reduction, clustering, and analysis of large document collections [2, 3]. Also, we can apply this to multi-lingual, multi-disciplinary data collections, because it uses statistics of co-occurring words or features rather than their real meaning. The method can be applied to non-linguistic problems such as preference learning and scene analysis in which the properties of interest can be expressed in a similar model.

However, the topic model estimation process can be computationally very expensive and may be inconceivable for huge collections [4]. Also, for systems such as Facebook, blog posts, and Twitter that are constantly updated and may require real-time responses, efficient topic inferencing methods are critical. It is desirable to have a system that can infer the topic structure of newly encountered documents without retraining the estimated topic model. By *topic learning*, this paper means the topic model estimation process for a batch of documents in the corpus (or complete document collection). *Topic inferencing*,

on the other hand, denotes the identification of topic distributions of newly encountered documents.

In this paper, we use a product partition model view of the LDA generative topic model [1] by Claudio et al. to characterize documents. We describe how the newly obtained posterior from the LDA product portion model aids online topic learning and topic inference. We use Markov Chain Monte Carlo (MCMC) techniques such as Gibbs sampling and Metropolis search for the parameter estimation process. We also experimented with several random walk techniques to increase the accuracy of document topic mixtures learned from the Metropolis search. To check the consistency of the model, we tested our method with synthetically generated data based on the LDA generative process. In addition, we compare our method to the use of Gibbs sampling for inferring topic mixtures of newly encountered documents.

The rest of the paper is organized as follows. Section 2 describes state of the art methods to solve the online topic learning and topic inference problems. In section 3, we compare the original LDA model and the product partition view. Section 4 details topic inference and batch learning approaches. In section 5, we analyze our results and discuss the issues of emerging topics and identifiability of topics in both topic learning and inference. Section 6 concludes the discussion and identifies anticipated future work.

2. Related work

Fitting complex Bayesian topic models to huge document collections is an interesting problem in machine learning and applied statistics. One way to tackle this problem is to fit a topic model in an online fashion, i.e., to fit the topic model using streaming document collections. Hoffman et al. [5] explained an online topic learning model, which is based on a variational Bayes inference strategy, for large document collections. This model ignores already visited documents during the online stochastic estimation process.

AlSumait et al. [6] discussed an online version of the LDA model that incrementally builds topic models when it observes a set of documents. Their solution is based on an empirical Bayes method and can track topics during a time period. It uses the counts of words in specific topics (i.e., the β matrix in the Blei et al. LDA model [2]), which is learned from the current batch of documents, as a prior for the future batch of documents. Additionally, it ignores current documents for the future learning process. Our approach to batch topic learning of the document collections is similar to this, however, we use the prior knowledge from the learned model for running a full batch Gibbs sampler of LDA. This sampler, based on product partition models, has been shown to mix better than the original LDA collapsed Gibbs sampler [7, 1].

Yao et al. [4] explain several heuristics to infer topic-mixtures from streaming documents based on methods such as Gibbs sampling, variational inference, and classification. However, their Gibbs sampling based inference is based on Griffiths' [7] collapsed Gibbs sampler.

Our topic inference is based on the new posterior of the per-word topic assignment given by a product partition model [1], and we perform a Metropolis search in the topic space for finding the topic structure of the newly encountered documents. Metropolis search has the advantage of seeking to optimize the posterior probability rather than attempting to characterize its distributions as would a Gibbs sampling approach. Furthermore, with

an appropriate choice of candidate distributions, the Hastings correction can be avoided, yielding a highly efficient technique.

3. LDA and the Product Partition Model

Latent Dirichlet Allocation (LDA) is a probabilistic, graphical model for representing latent topics or hidden variables of the documents in a text corpus [2]. LDA represents a document in the corpus by distributions over topics and a topic itself as a distribution over all terms in the corpus. LDA assumes that a document is a bag-of-words exhibiting exchangeability. We use the following notation and terminology [1, 2] in this paper:

- The corpus vocabulary contains V words and k topics (multinomial distributions) are defined over the vocabulary.
- D represents the total number of documents in the corpus, and each document d has n_d words.
- $\mathbf{w}_d = (w_{1,d}, w_{2,d}, ..., w_{n_d,d})$ represents a document d in the corpus, where $w_{i,d}$ is the i^{th} word instance in the document d.
- Each $w_{i,d}$ is a 1 of V vector with $w_{i_t,d} = 1$ for the sampled word t from the corpus vocabulary.
- $z_{i,d}$, a 1 of k indicator vector with only one entry equals one, represents the topic assignment for the word instance $w_{i,d}$.
- β, a $k \times V$ matrix, represents topic word probabilities.
- θ, a $k \times D$ matrix, represents topic distributions for all documents.

LDA's generative process [2] is as follows:

- For each topic $j = 1, ..., k, \beta_j \sim Dirichlet(b_1, b_2, ..., b_V)$, where $b_1, b_2, ..., b_V$ are Dirichlet hyper-parameters for the topic distributions spanned over the vocabulary
- For each document $d = 1, ..., D$, $\theta_d \sim Dirichlet(\alpha_1, \alpha_2, ..., \alpha_k)$, a topic mixture, where $\alpha_1, \alpha_2, ..., \alpha_k$ are Dirichlet hyper-parameters
- For each word $w_{i,d}$ in the document d :
  - $z_{i,d} \sim Multinomial(1, \theta_d)$
  - $w_{i,d} \sim Multinomial(1, \beta_{j'})$ where $z_{ij',d} = 1$

  According to Blei et al. the model inference is based on

$$P(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{P(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{P(\mathbf{w}|\alpha, \beta)}, \mathbf{w} = \mathbf{w}_1, ..., \mathbf{w}_D, \mathbf{z} = \mathbf{z}_1, ..., \mathbf{z}_D, \quad (1)$$

  The numerator is

$$P(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = \prod_{d=1}^{D} \prod_{i=1}^{n_d} \sum_{z_d} P(w_{i,d}|z_{i,d}, \beta) P(z_{i,d}|\theta_d) P(\theta_d|\alpha) \tag{2}$$

and there is no z_d on the right side, since the result is formed from a sum over all z_d. Thus the solution cannot be used for classification based on explicit knowledge of z_d.

3.1 Product Partition Model for Latent Dirichlet Allocation

This was fully developed in [1]. In this view, instead of mixing over the z_d, it assumes each $w_{i,d}$ as coming from one topic. At first sight we may say that $w_{i,d}$ loses the ability to spread across topics, however, as it is stochastic, there will be many competing classifications that will allow $w_{i,d}$s to be in different topics. Also, this approach can allow classification on z_d.

For clarity, we reiterate the product partition model [1] here. The joint probability of a corpus can be written as

$$P(\mathbf{z}, \theta, \mathbf{w}|\alpha, b) = \left[\prod_{d=1}^{D} P(\mathbf{w}_d|\mathbf{z}_d, \beta) P(\mathbf{z}_d|\theta_d) P(\theta_d|\alpha_1, ..., \alpha_k) \right] P(\beta|b_1, ..., b_V) \tag{3}$$

Ignoring the normalizing constant, the posterior of $(\mathbf{z}, \theta, \beta)$ is

$$P(\mathbf{z}, \theta, \beta|\mathbf{w}, \alpha, b) \propto \left(\prod_{t=1}^{V} \prod_{j=1}^{k} \beta_{jt}^{\sum_{d=1}^{D} m_{jt,d} + b_t - 1} \right) \left(\prod_{d=1}^{D} \prod_{j=1}^{k} \theta_{j,d}^{n_{j,d} + \alpha_t - 1} \right) \tag{4}$$

where $n_{j,d} = \sum_{i=1}^{n_d} z_{ij,d}$ and $m_{jt,d} = \sum_{i=1}^{n_d} w_{it,d} z_{ij,d}$.

Eq. 4 clearly illustrates the *product partition* model. Here, we can see that the sample $w_{i,d}$ is partitioned according to the values of the $z_{i,d}$.

4. Topic learning and inference methods

In the conventional topic model, LDA, the goal of model inference is to identify θ and β matrices, for a given set of documents $\mathbf{w} = \mathbf{w}_1, ..., \mathbf{w}_D$. However, this estimation is complicated due to the latent variables $\mathbf{z_d}$, the word-topic assignments, existing in the model. Algorithms such as variational inference [2], expectation propagation [8], and collapsed Gibbs sampling [7] have been proposed to solve this problem. In this paper, we focus on MCMC based inference methods. Our methods are based on the product partition model based LDA posterior (Eq. 4).

4.1 The full Gibbs sampler of LDA

In Griffiths' collapsed Gibbs sampler for LDA, the model variables θ and β are integrated out [7], and it only samples $\mathbf{z}_d$, the latent topic variables. Its state space is limited to the set of all possible topic assignments ($\mathbf{z}$) to the corresponding word instances in the corpus. In this paper, we use the full Gibbs sampler [1] for LDA topic learning, which samples θ, β, $\mathbf{z}_d$ in one Gibbs sampling step. We show the LDA 2-stage full Gibbs sampler here for clarity.

1. Initialize $\mathbf{z}_d$ randomly for $d = 1, ..., D$
2. Sample β and θ:

$$\beta_j \sim Dirichlet(\sum_{d=1}^{D} m_{j1,d} + b_1, ..., \sum_{d=1}^{D} m_{jV,d} + b_V), j = 1, ..., k \tag{5}$$

$$\theta_d \sim Dirichlet(n_{1,d} + \alpha_1, ..., n_{k,d} + \alpha_k), d = 1, ..., D \tag{6}$$

3. Given β and θ sample z_d:

$$z_{i,d} \sim Multinomial(1, (p_{i1,d}, ..., p_{ik,d})), i = 1, ..., n_d \tag{7}$$

where

$$p_{ij,d} \propto \prod_{t=1}^{V} (\beta_{jt}\theta_{j,d})^{w_{it,d}}, j = 1, ..., k$$

The paper [1] proves that this sampler dominates the LDA collapsed Gibbs sampler [7] in the covariance ordering. It is also shown that this 2-stage full Gibbs sampler has a faster convergence rate.

4.2 Topic inference using Metropolis-Hastings search

Typically, topic inference algorithms are designed to identify the model distributions on β and θ from the entire document collection. Multiple lookups of the entire collection may be required to get an optimal approximation of the model. For applications in which a huge training corpus defines the topics to be used for inference with single or small sets of documents, batch learning methods [2, 8, 7] are not suitable due to inference speed and storage space requirements. In this section, we discuss a model that is suitable for the aforementioned applications.

We can use the product partition model (Eq. 4) to find optimal topic allocations of document-words in a corpus [1]. In our proposed topic inference method, we assume that we have learned a LDA model ($\hat{\beta}$ and $\hat{\theta}$) using any of the previously mentioned estimation methods [2, 7, 1, 8]. We assume these topic distributions, $\hat{\beta}_j$, represent the corpus's topic structure well. From Eq. 4, and a given $\hat{\beta}$ evidence, we calculate the marginal posterior on $\mathbf{z}_d$ for a single document[1], by integrating out θ as:

$$\begin{aligned} P(\mathbf{z}|\mathbf{w}, \alpha, b) &\propto \int \left(\prod_{t=1}^{V} \prod_{j=1}^{k} \hat{\beta}_{jt}^{m_{jt}+b_t-1} \right) \left(\prod_{j=1}^{k} \theta_j^{n_j+\alpha_t-1} \right) d\theta \\ &= \prod_{j=1}^{k} \left(\prod_{t=1}^{V} \hat{\beta}_{jt}^{m_{jt}+b_t-1} \right) \Gamma(n_j + \alpha_j) \end{aligned} \tag{8}$$

Note: we can ignore the normalization constant here, since we only need to search for the $\mathbf{z}_d$ that has maximal posterior probability. We don't need to know its normalized likelihood.

1. Here, we ignore the document index d for ease of notation.

In the Metropolis search algorithm [9], we sample from a known candidate distribution, and accept the sample with probability

$$a(\mathbf{z}', \mathbf{z}) = min\left(1, \frac{P(\mathbf{z}')}{P(\mathbf{z})}\right) \tag{9}$$

We use Eq. 8 as our objective function $P(\mathbf{z})$. In this basic case, we assume that candidate distribution is symmetric, i.e., $T(\mathbf{z}, \mathbf{z}') = T(\mathbf{z}', \mathbf{z})$, where $T(\mathbf{z}, \mathbf{z}')$ represents the probability of transition from the state $\mathbf{z}$ to $\mathbf{z}'$. If this property does not hold, we must add the Hasting's correction [10] to this basic Metropolis acceptance ratio.

Simulated annealing

Determining the number of steps required by the Markov chain to reach its stationary distribution is a difficult task in any MCMC sampler. Choice of initialization conditions and proposal distribution can make an impact on this task. We initially used a *topical* multinomial, i.e., multinomial probabilities with peaks around specific topics, as our candidate distribution for the topic search. With this proposal distribution, the Metropolis acceptance was not impressive. This led us to employ methods such as simulated annealing to improve Metropolis acceptance, yielding better topic-mixture distributions for documents. Section 5 describes strategies to assess inferred documents.

Simulated annealing [11, 12] is a method for estimating complex distributions with multiple peaks, where standard hill-climbing methods may trap the algorithm at a local minimum. The fundamental idea is that during the initial sampling phase, we accept a reasonable probability of a down-hill move to explore the entire state space. Later on, we decrement the probability of such down-hill moves. This yielded some improvement, leading us to pursue the hybrid Metropolis search explained in the next section.

Choosing a candidate distribution - hybrid random walk

We can also tune up the candidate distribution to improve mixing and especially the acceptance probability. To accomplish this goal, we adopt a hybrid sampling scheme [13, 14], which is a mix of *random walks* and *independent draws*. In a candidate distribution that is based on a random walk chain, the new state z' is calculated as

$$z' = z + y \tag{10}$$

where y represents a random walk from the current state z. Based on the distribution of this random variable y, the transition probability $T(z, z')$ can be taken to be symmetric which will speedup the the Metropolis search. We can include a random walk component in the generative process of partitions or topics, which may help the search algorithms to stay at highly probable areas, maintaining the required stationary distribution [14].

On the other hand, in an *independent draw* chain, the Markov chain jumps to the future state z' independent of the current state z from a candidate distribution. In our hybrid scheme, we combine the advantages of both as described below.

Hybrid random walk algorithm:

In the topic search or inference, we consider a single document in the corpus at a time. This hybrid random walk is formed from the LDA model assumption of a document, i.e.,

a single document is generated from an allocation to partitions or topics. We evaluate these partitions by counting word associations to the corresponding corpus wide topics. Let's assume we sample k topic partitions $\omega = (\omega_1, \omega_2, ..., \omega_k)$ using a Metropolis-Hastings algorithm for a document, by the following steps:

1. Initialization: draw an independent sample $\omega^{\mathbf{0}} \sim g$, the candidate distribution
2. For each step t:
   (a) Given a random walk probability b, do a *random walk* from the current state, ω, otherwise, draw independently from g
   (b) Calculate the Metropolis-Hastings acceptance probability [10]

$$a = min\left(1, \frac{P(z'|.)Q(\omega', \omega)}{P(z|.)Q(\omega, \omega')}\right) \tag{11}$$

   where $P(\mathbf{z}|.)$ from Eq. 8 and

$$Q(\omega, \omega') = b\, T(\omega, \omega')_{random\ walk} + (1 - b)\, T(\omega, \omega')_{independent\ draw} \tag{12}$$

   (c) With probability a, accept or reject the sample ω'

Random walk and the g function:

Selection of g is an open problem. One possible method is – Given the current state of partitions ω, select one observation at random, and reallocate it to one of the other $k - 1$ clusters with uniform probability [14]. Based on the random walk method, we update the transition probability $T(\omega, \omega')_{random\ walk}$. Another tweaking parameter in this model is the random walk probability, b. The function g represents any conventional distribution that is easy to sample and evaluate $T(\omega, \omega')_{independent\ draw}$, e.g., a uniform distribution. Our observations with the hybrid random walk are shown in section 5.

4.3 Online batch learning

Conventional topic learning algorithms [2, 7, 8] are designed to run over entire document collections. These algorithms may be best suited to static databases rather than dynamically growing or evolving document corpora. Topic search discussed in the above section is useful for single documents or small document batches with a learned β, which represents the topic structure of the given corpus. However, one drawback of applying this method to a growing database is that the Metropolis algorithm does not learn β, which makes it unsuitable for applications where the corpus topic structure changes over time. In this section, we address this problem by modifying the LDA full Gibbs sampler discussed in section 4.1. This problem was attacked by AlSumait et al. [6] using an empirical Bayes approach.

Modifications:

- Corpus documents arrive as batches
- For the initial batch, we use the full Gibbs sampler , since we have not seen any documents. Suppose, after the initial Gibbs run, we get initial model parameters β_0 and θ_0.

- We sample z_d (Eq. 7) only for new document batches

- The key idea is to replace the prior $Dirichlet(b_1, ..., b_V)$ distributions, with β_{prior} that is calculated from previous batches as:

$$\beta_{prior} = \left\{\epsilon_0\beta_0 + \epsilon_1\beta_1 + ...\epsilon_{(L-1)}\beta_{(L-1)}\right\}$$

  where L indexes the current batch and $\epsilon_l \in (0, 1]$ are the weights associated with every batch. These weights are model tweaking parameters that influence the sampler to forget or remember information about prior document batches. A typical value of ϵ_l would be one, that is the sampler will remember the all prior batches' topic distributions.

- We update Eq. 5 as

$$\beta_j \sim Dirichlet(\sum_{d=1}^{D} m_{j1,d} + \beta_{j1,prior}, ..., \sum_{d=1}^{D} m_{jV,d} + \beta_{jV,prior}), j = 1, ..., k \qquad (13)$$

5. Experiments

To asses the quality and consistency of the topic inference and online learning algorithms, we first applied them to a synthetic document corpus. This dataset was generated based on the LDA generative process (section 3) and a known β_{true}. Together with β_{true}, we stored each document's topic proportions $\theta_{d,true}$, which is sampled from a Dirichlet distribution with given hyper-parameters.

Figure 1 shows a comparison of β_{true} and the estimated β matrices from the full Gibbs sampler ($\hat{\beta}_{full}$), online batch Gibbs sampler ($\hat{\beta}_{online\,batch}$), online incremental Gibbs sampler ($\hat{\beta}_{incremental}$), simulated annealing ($\hat{\beta}_{anneal}$), and hybrid random walk ($\hat{\beta}_{hybrid}$) algorithms. For the full and online batch Gibbs samplers, $\hat{\beta}_{full}$ and $\hat{\beta}_{online\,batch}$ are computed by finding the mean of β over all sampling iterations after the burn-in period. For the incremental Gibbs sampler, simulated annealing, and hybrid random walk, β is computed from the highest probability $\mathbf{z}$ that was sampled. The key difference between the incremental Gibbs sampler and online batch Gibbs sampler is that for the newly encountered documents, the former runs the online Gibbs sampler (section 4.3) in a document-by-document basis, however, the latter runs the online Gibbs sampler for the entire collection at a time. For the simulated annealing experiment, we started with an annealing temperature of 10, and took 2000 iterations to cool down to 1. The hybrid Metropolis search performed a random walk to other clusters uniformly with probability 0.7, and random sampling from the candidate, topical multinomials, with probability 0.3.

The x axis represents the vocabulary of words and y axis represents the corpus topics or partitions, thus, each row is a multinomial over the vocabulary. The value in each cell (x, y) corresponds to the relative frequency of word x in topic multinomial y showing shades ranging from white for zero and black for the maximal frequency in the table. One can see

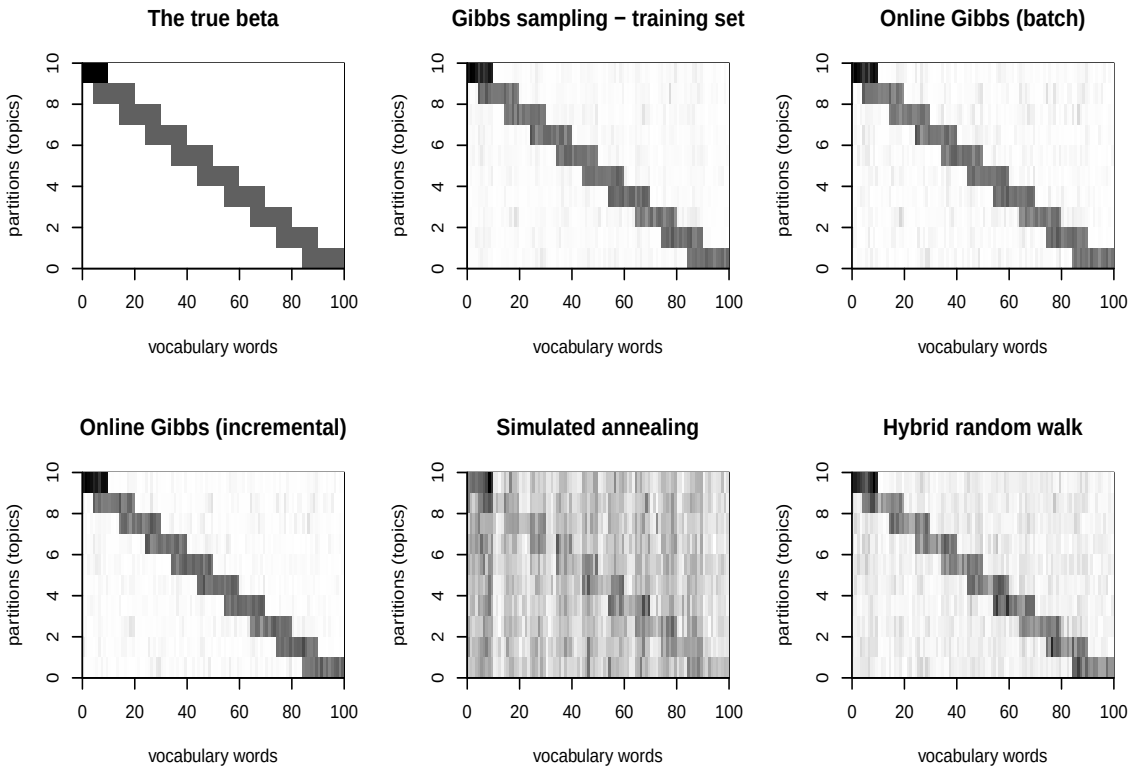

Figure 1: The generating β_{true} and estimated $\hat{\beta}_{full}$ (from 200 training documents) , $\hat{\beta}_{online\,batch}, \hat{\beta}_{incremental}, \hat{\beta}_{anneal}$, and $\hat{\beta}_{hybrid}$ (from 100 test documents)

that the hybrid random walk generates estimates that are comparable to the Gibbs samplers. Furthermore, the hybrid walk algorithm is significantly less computationally expensive [2].

Note that the Gibbs samplers (the full and online batch) do not maintain a-priori orderings of (synthetic data) topics, even though they find topic partitions from document collections. The papers [15, 16] also mentioned the problem of non-identifiability in topic learning. To make comparison easier, we re-ordered the estimated topics from the Gibbs samplers to best match β_{true} and θ_{true}. However, if we use the Metropolis topic search or incremental Gibbs sampling for estimating the LDA model parameters, the topic numbers will match the given β matrix, i.e., the topics are readily identifiable.

we used K-L divergence to measure the difference between a document's generative topic mixture $\theta_{d,true}$ and its inferred topic mixture $\hat{\theta}_d$, as they are distributions. Figure 2 depicts an analysis of the estimated $\hat{\theta}_{online\,batch}$, $\hat{\theta}_{incremental}$, and $\hat{\theta}_{hybrid}$, i.e., document topic mixtures inferred using the previously discussed approaches (section 4.2 and 4.3). We can

2. Our hybrid topic search is 20 times faster than the Gibbs sampler based topic inferences on the synthetic dataset. We implemented the algorithms in C++ using the libraries Armadillo C++ and GSL, and ran them on an Intel Core 2 Duo machine with 4GB DDR2.

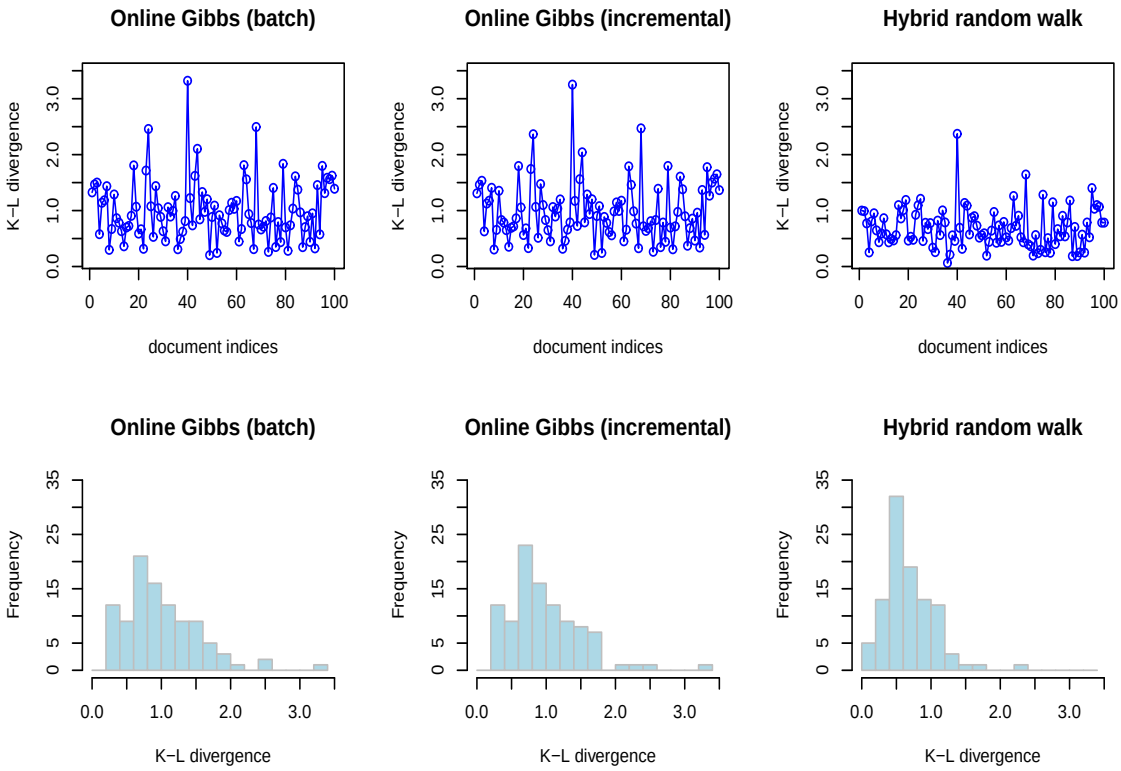

Figure 2: K-L divergence values between the generating $\theta_{d,true}$ and estimated $\hat{\theta}_{online\,batch}$, $\hat{\theta}_{incremental}$, and $\hat{\theta}_{hybrid}$ (where $d = 1, 2, ..., 100$)

see that the online Gibbs samplers result in similar θ estimates, while the hybrid random walk results in slightly lower estimates in the worst case (smaller K-L divergences).

Using Gibbs sampling to estimate topic mixtures for an a-priori set of topic multinomial exhibits the negative side-effect of actually changing the reference topic mixtures as part of its process. For small numbers of topics, this may be negligible, but it can make a difference as the size of documents to be analyzed increases.

6. Summary

In this paper, we proposed two novel topic learning and inference methods for a single document or collection of documents. In our experiments, we found that the topic search algorithm, which is based on the product partition model alternative to Latent Dirichlet Allocation models, works as well as the full Gibbs sampling estimates on a document batch. Moreover, we can do topic search on a per document basis, without updating the given evidence β. Additionally, we proposed an online batch learning algorithm that can be applied on huge document collections, forgetting already learned documents. In summary,

we can use a combination of both algorithms to solve the problem of topic inference of streaming document collections.

Acknowledgments

We would like to acknowledge support for this project from the National Science Foundation (Grants # 0712799, # MMS-1028314, and # DMS-1105127) and the Army Research Office (Grant # W911NF-08-10410). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Office, Army Research Laboratory, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. The authors would like to thank R. Harmon, R. Weaver, P.Howard, and T. Donzelli for their support of this work.

References

[1] Claudio Fuentes, Vikneshwaran Gopal, George Casella, Clint P. George, Taylor C. Glenn, Joseph N. Wilson, and Paul D. Gader. Product partition models for dirichlet allocation. Technical Report 519, September 2011. The Dept of CISE, University of Florida.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *J. Machine Learning Research*, 3:993–1022, 2003.

[3] David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *J. ACM*, 57:7:1–7:30, February 2010.

[4] Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 937–946, New York, NY, USA, 2009. ACM.

[5] Matthew Hoffman, David M. Blei, and Francis Bach. Online learning for latent dirichlet allocation. In *NIPS*, 2010.

[6] Loulwah AlSumait, Daniel Barbará, and Carlotta Domeniconi. On-line LDA: Adaptive Topic Models for Mining Text Streams with Applications to Topic Detection and Tracking. *Data Mining, IEEE International Conference on*, 0:3–12, 2008.

[7] Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences of the United States of America*, 101(Suppl 1):5228–5235, April 2004.

[8] Thomas Minka Department, Thomas Minka, and John Lafferty. Expectation-propagation for the generative aspect model. In *In Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence*, pages 352–359. Morgan Kaufmann, 2002.

[9] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[10] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, April 1970.

[11] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, November 1984.

[12] Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I. Jordan. An Introduction to MCMC for Machine Learning. *Machine Learning*, 50(1):5–43, January 2003.

[13] James G. Booth, George Casella, and James P. Hobert. Clustering using objective functions and stochastic search. *Journal of the Royal Statistical Society - Series B: Statistical Methodology*, 70(1):119–139, 2008.

[14] Claudio Fuentes and George Casella. Testing for the existence of clusters. *SORT*, 33(2):115–146, 2009.

[15] Michal Rosen-Zvi, Thomas Griffiths, Padhraic Smyth, and Mark Steyvers. Learning author topic models from text corpora. Technical report, IBM Research, UC Irvine, and UC Berkeley, November 2005.

[16] Yee Whye Teh, Kenichi Kurihara, and Max Welling. Collapsed variational inference for hdp. In *Advances in Neural Information Processing Systems (NIPS)*, 2008.