Stochastic Bounds for Inference in Topic Models

Xuan Bui[1,2(✉)], Tu Vu[1], and Khoat Than[1]

[1] Hanoi University of Science and Technology,
No.1, Dai Co Viet Road, Hanoi, Vietnam
thanhxuan1581@gmail.com, vutu201130@gmail.com,
khoattq@soict.hust.edu.vn
[2] Thai Nguyen University of Information and Communication Technology,
Quyet Thang, Thai Nguyen, Vietnam

Abstract. Topic models are popular for modeling discrete data (e.g., texts, images, videos, links), and provide an efficient way to discover hidden structures/semantics in massive data. The problem of posterior inference for individual texts is particularly important in streaming environments, but often intractable in the worst case. Some existing methods for posterior inference are approximate but do not have any guarantee on neither quality nor convergence rate. Online Maximum a Posterior Estimation algorithm (OPE) [13] has more attractive properties than existing inference approaches, including theoretical guarantees on quality and fast convergence rate. In this paper, we introduce three new algorithms to improve OPE (so called OPE1, OPE2, OPE3) by using stochastic bounds when doing inference. Our algorithms not only maintain the key advantages of OPE but often outperform OPE and existing algorithms. Our new algorithms have been employed to develop new effective methods for learning topic models from massive/streaming text collections.

Keywords: Topic models · Posterior inference · Online map estimation · Large-scale learning · OPE

1 Introduction

Latent Dirichlet Allocation (LDA) [4] is the class of Bayesian networks. It has found successful applications in a wide range of areas including text modeling [5], bioinformatics [2,9], history [6], politics [12,17], psychology [8]. Estimation of posterior distributions for individual documents is one of the core issues in LDA. Recently, this estimation problem is considered by many researchers, and many methods such as Variational Bayes (VB) [4], Collapsed Variational Bayes (CVB) [19], CVB0 [1], and Collapsed Gibbs Sampling (CGS) [7,18], OPE [13], have been proposed. The quality of LDA in practice is determined by the quality of the inference method being employed. However, except OPE, none of the mentioned methods has a theoretical guarantee on quality or convergence rate.

Our first contribution is the introduction of some new algorithms for doing posterior inference of topic mixture in LDA by improving OPE algorithm.

M. Akagi et al. (eds.), *Advances in Information and Communication Technology*,
Advances in Intelligent Systems and Computing 538, DOI 10.1007/978-3-319-49073-1_62

They are called OPE1, OPE2, OPE3. The posterior inference problem is in fact nonconvex and is NP-hard [3]. OPE is stochastic in nature and theoretically converges to a local maximal/stationary point of the inference problem. In [13], Than and Doan proved that OPE converges at a rate of $O(1/T)$, which surpasses the best rate of existing stochastic algorithms for nonconvex problems [10,16], where T is the number of iterations. One main drawback of OPE is that there is no guarantee for OPE to get rid of saddle points of the inference problems.[1] In this paper, we propose three new variants of OPE, which are called OPE1, OPE2, and OPE3 for doing inference in topic models. Note that OPE does inference by maximizing an objective function of the inference problem. In our algorithms, we use both upper and lower bounds of the objective function. The usage of both bounds is stochastic in nature and help us reduce the possibility of getting stuck at a local stationary point. Therefore, our new variants are more beneficial than the original OPE [13].

Our second contribution in this paper is that we introduce six new methods for learning LDA from massive/streaming text collections. Those methods employ OPE1, OPE2, and OPE3 as their core routines to do inference. From extensive experiments on two large corpora we find that some of our methods can reach state-of-the-art performance in both predictiveness and model quality.

Organization: The rest of this paper is organized as follows. We introduce an overview of posterior inference with LDA, followed by OPE algorithm in Sect. 2. In Sect. 3, three new algorithm for inference are proposed in detail. Section 4 is application of proposed algorithms applied to online learning LDA. In Sect. 5, we give some results test with large datasets and we conclude the paper in Sect. 6.

Notation: Throughout the paper, we use the following conventions and notations. Bold faces denote vectors or matrices. The unit simplex in the n-dimensional Euclidean space is denoted as $\Delta_n = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{x} \geq 0, \sum_{k=1}^{n} x_k = 1\}$, and its interior is denoted as $\overline{\Delta}_n$. We will work with text collections with V dimensions (dictionary size). Each document $\boldsymbol{d}$ will be represented as a frequency vector, $\boldsymbol{d} = (d_1, ..., d_V)^T$ where d_j represents the frequency of term j in $\mathbf{d}$. Denote n_d as the length of $\boldsymbol{d}$, i.e., $n_d = \sum_j d_j$.

2 Background on Topic Models and Posterior Inference

A topic model often assumes that a corpus is composed from K topics, $\beta = (\beta_1, .., \beta_K)$. Each document $\mathbf{d}$ is a mixture of those topics and is assumed to arises from the following generative process. For the n^{th} word of $\mathbf{d}$:

- draw topic index $z_{dn}|\theta_d \sim Multinomial(\theta_d)$
- draw word $w_{dn}|z_{dn}, \beta \sim Multinomial(\beta_{z_{dn}})$

[1] A saddle point is not always a (local) maximal point. Further, the inference might have exponentially large number of saddle points.

Each topic mixture $\theta_d = (\theta_1, \ldots, \theta_K)$ represents the contributions of topics to document $\mathbf{d}$, while β_{kj} shows the contribution of term j to topic k. Note that $\theta \in \Delta_K, \beta_k \in \Delta_V, \forall k$. Both θ and $\mathbf{z}$ are hidden variables and are local for each document.

The generative process above generally describes what probabilistic latent semantic analysis (PLSA) is [4,5]. Latent Dirichlet allocation (LDA) further assumes that θ and β are samples of some Dirichlet distributions. More specifically, $\theta \sim Dirichlet(\alpha)$ and $\beta_k \sim Dirichlet(\eta)$ for any topic. The problem of posterior inference for each document $\mathbf{d}$, given a model $\{\beta, \alpha\}$, is to estimate the full joint distribution $p(\mathbf{z}_d, \theta, \mathbf{d}|\beta, \alpha)$. Direct estimation of this distribution is intractable, i.e., NP-hard in the worst case. Hence, existing inference approaches use different schemes. VB, CVB and CVB0 try to estimate the distribution by maximizing a lower bound of the likelihood $p(\mathbf{d}|\beta, \alpha)$, whereas CGS tries to estimate $p(z|d, \beta, \alpha)$. We consider the MAP estimation of topic mixture for a given document d

$$\theta^* = \mathbf{argmax}_{\theta \in \overline{\Delta}_K} Pr(\theta, \mathbf{d}|\beta, \alpha) = \mathbf{argmax}_{\theta \in \overline{\Delta}_K} Pr(d|\theta, \beta) Pr(\theta|\alpha) \tag{B.1}$$

Remember that the density of the K-dimensional Dirichlet distribution with parameter α is $P(\theta|\alpha) \propto \prod_{k=1}^{K} \theta_k^{\alpha-1}$. Therefore problem (B.1) is equivalent to the following:

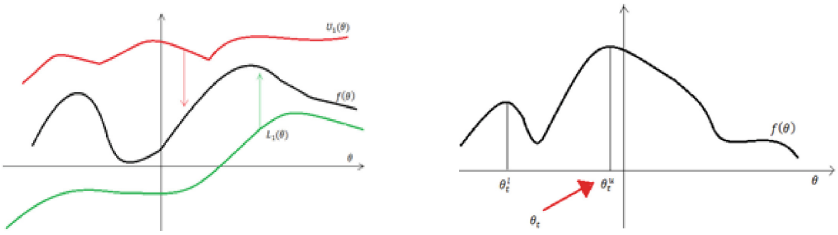
$$\theta^* = \mathbf{argmax}_{\theta \in \overline{\Delta}_K} \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k \tag{B.2}$$

Sontag and Roy [3] showed that this problem is NP-hard in the worst case when $\alpha < 1$. In the case of $\alpha \geq 1$, one can easily show that the problem (B.2) is concave, and therefore it can be solved in polynomial time. Unfortunately, in practice of LDA, the parameter α is often small, says $\alpha < 1$, causing (B.2) to be nonconcave. That is the reason for why (B.2) is intractable in the worst case.

Than and Doan [13] present OPE for doing inference of topic mixtures for documents. The idea of OPE is quite simple. It solves problem (B.2) by iteratively find a vertex of $\overline{\Delta}_K = \{x \in R^K : \sum_{k=1}^{K} x_k \geq \varepsilon > 0\}$ to improve its solution. A good vertex at each iteration is decided by assessing stochastic approximations to the gradient of the objective function. When the number of iterations goes to infinity, OPE will approach to a local maximal/stationary point of problem (B.2).

3 Three New Algorithms for Inference

In this section, we describe important characters of OPE, which were investigated by Than and Doan [13]. Then, we analyze some new perspectives of OPE which lead to our improvements. OPE is a stochastic algorithm. Denote $g_1(\theta) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$ and $g_2(\theta) = (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$. It constructs a sequence of random functions that approximate the objective function of interest



(a) Upper bound and lower bound of objective function

(b) Choose the best point

Fig. 1. Ideas to improve OPE

Algorithm 1. OPE1: The first variant of OPE

Input: document d and model $\{\boldsymbol{\beta}, \alpha\}$
Output: $\boldsymbol{\theta}$ that maximizes
$f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k$.
Initialize $\boldsymbol{\theta_1}$ arbitrary in Δ_K
$f_1^u := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $f_1^l := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 1, 2, ...\infty$ **do**
Pick f_t^u unifromly from $\{\sum_j d_j \log \sum_{k=1}^K \theta_j \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
$U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
$\boldsymbol{e}_t^u := argmax_{\boldsymbol{x} \in \Delta_K} < U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$
$\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$
Pick f_t^l unifromly from $\{\sum_j d_j \log \sum_{k=1}^K \theta_j \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
$L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
$\boldsymbol{e}_t^l := argmax_{\boldsymbol{x} \in \Delta_K} < L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$
$\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$
$\boldsymbol{\theta}_{t+1} :=$ pick uniformly from $\{\boldsymbol{\theta}_{t+1}^u, \boldsymbol{\theta}_{t+1}^l\}$
end for

$f(\theta) = g_1(\theta) + g_2(\theta)$. OPE works by alternatively choosing $g_1(\theta), g_2(\theta)$ in each iteration to create a random function $F_t(\theta)$. Then, $F_t(\theta)$ converges to $f(\theta)$ when $t \to \infty$.

We recognized that $g_1(\theta) < 0, g_2(\theta) > 0$. Hence, in the first iteration, if we choose $g_1(\theta)$ then $F_1(\theta) < f(\theta)$, the sequence of random functions goes from below $f(\theta)$ according to coordinate. In constrast, if we choose $g_2(\theta)$ in the first iteration then $F_1(\theta) > f(\theta)$, the sequence of random functions goes from above $f(\theta)$. We got an idea to create two sequences of random functions that both converging to $f(\theta)$, one begins with $g_1(\theta)$, other begins with $g_2(\theta)$, both converge to $f(\theta)$ (Fig. 1a). Two random sequences give us more information about objective function, so that we can get a better result when seeking maximal point of $f(\theta)$. OPE1 is aimed at increasing randomness of a stochastic algorithm. Getting idea from random forest, which construct a lot of random trees to get the average result of all trees, we use randonness to create a plenty of choices in

our algorithm. We hope that with fully randomness OPE1 can jump over local stationary points to get higher local stationany point. We pick θ_t uniformly from $\{\theta_t^u, \theta_t^l\}$.

Algorithm 2. OPE2: The second variant of OPE

Input: document d and model $\{\boldsymbol{\beta}, \alpha\}$
Output: $\boldsymbol{\theta}$ that maximizes

$$f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^K \log \theta_k.$$

Initialize $\boldsymbol{\theta_1}$ arbitrary in Δ_K
$f_1^u := \sum_j d_j \log \sum_{k=1}^K \theta_k \beta_{kj}$; $f_1^l := (\alpha - 1) \sum_{k=1}^K \log \theta_k$
for $t = 1, 2, ...\infty$ **do**
  Pick f_t^u unifromly from $\{\sum_j d_j \log \sum_{k=1}^K \theta_j \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $U_t := \frac{2}{t} \sum_{h=1}^t f_h^u$
  $\boldsymbol{e}_t^u := argmax_{\boldsymbol{x} \in \Delta_K} < U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$
  $\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$
  Pick f_t^l unifromly from $\{\sum_j d_j \log \sum_{k=1}^K \theta_j \beta_{kj}; (\alpha - 1) \sum_{k=1}^K \log \theta_k\}$
  $L_t := \frac{2}{t} \sum_{h=1}^t f_h^l$
  $\boldsymbol{e}_t^l := argmax_{\boldsymbol{x} \in \Delta_K} < L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$
  $\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$

  $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_{t+1}^u$ with probability $\frac{exp(f(\boldsymbol{\theta}_{t+1}^u))}{exp(f(\boldsymbol{\theta}_{t+1}^u)) + exp(f(\boldsymbol{\theta}_{t+1}^l))}$
and $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_{t+1}^l$ with probability $\frac{exp(f(\boldsymbol{\theta}_{t+1}^l))}{exp(f(\boldsymbol{\theta}_{t+1}^u)) + exp(f(\boldsymbol{\theta}_{t+1}^l))}$
end for

OPE2 continues with ideas of rising randomness, we pick θ_t from $\{\theta_t^u, \theta_t^l\}$ with probabilities which depend on the value $\{f(\theta_t^u), f(\theta_t^l)\}$. This is smoother than probabilities $\{0.5, 0.5\}$ in OPE1.

The last improvement of OPE, we mix idea with greedy algorithm. We are maximizing a function, so in each iteration, we have two choices for a target point. We get the point that makes the value of objective function is higher (Fig. 1b). The idea of the algorithm OPE3 described our idea. OPE3 works differently from OPE. OPE just constructs one sequence of numbers while OPE3 makes three ones depending on each others. Therefore, the structure of sequence $\{\theta_t\}$ is changed. However, OPE3's properties are similar to OPE's.

Theorem 1.[2] *(**Convergence of OPE algorithms**): Consider the objective function $f(\theta)$ in problem (B.2), given fixed $\boldsymbol{d}, \beta, \alpha$. For OPE1, OPE2, OPE3, the followings hold:*

1. *For any $\theta \in \Delta_K$, $U_t(\theta)$ and $L_t(\theta)$ converges to $f(\theta)$ as $t \longrightarrow +\infty$,*
2. *θ_t converges in probability to a local maximal/stationary point of $f(\theta)$ at a rate of $O(1/t)$.*

[2] The detailed proof of this theorem will be presented in another paper.

Algorithm 3. OPE3 : The third variant of OPE

Input: document d and model $\{\boldsymbol{\beta}, \alpha\}$

Output: $\boldsymbol{\theta}$ that maximizes

$$f(\boldsymbol{\theta}) = \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj} + (\alpha - 1) \sum_{k=1}^{K} \log \theta_k.$$

Initialize $\boldsymbol{\theta_1}$ arbitrary in Δ_K

$f_1^u := \sum_j d_j \log \sum_{k=1}^{K} \theta_k \beta_{kj}$; $f_1^l := (\alpha - 1) \sum_{k=1}^{K} \log \theta_k$

for $t = 1, 2, ...\infty$ **do**

Pick f_t^u unifromly from $\{\sum_j d_j \log \sum_{k=1}^{K} \theta_j \beta_{kj}; (\alpha - 1) \sum_{k=1}^{K} \log \theta_k\}$

$U_t := \frac{2}{t} \sum_{h=1}^{t} f_h^u$

$\boldsymbol{e}_t^u := argmax_{\boldsymbol{x} \in \Delta_K} < U_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$

$\boldsymbol{\theta}_{t+1}^u := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^u - \boldsymbol{\theta}_t}{t}$

Pick f_t^l unifromly from $\{\sum_j d_j \log \sum_{k=1}^{K} \theta_j \beta_k j; (\alpha - 1) \sum_{k=1}^{K} \log \theta_k\}$

$L_t := \frac{2}{t} \sum_{h=1}^{t} f_h^l$

$\boldsymbol{e}_t^l := argmax_{\boldsymbol{x} \in \Delta_K} < L_t^{'}(\boldsymbol{\theta}_t), \boldsymbol{x} >$

$\boldsymbol{\theta}_{t+1}^l := \boldsymbol{\theta}_t + \frac{\boldsymbol{e}_t^l - \boldsymbol{\theta}_t}{t}$

$\boldsymbol{\theta}_{t+1} := argmax_{\boldsymbol{\theta} \in \{\boldsymbol{\theta}_{t+1}^u, \boldsymbol{\theta}_{t+1}^l\}} f(\boldsymbol{\theta})$

end for

4 New Approaches for Learning LDA

ML-OPE and Online-OPE are two examples for exploiting OPE algorithm proposed by Than and Doan [13]. These algorithms learn parameters of the LDA model in online schema and streaming data. The core of two algorithms is OPE, which makes ML-OPE and Online-OPE be state-of-the-art algorithms in learning LDA. They are the best algorithm in comparison to others. Their properties were carefully explored in [13].

Based on evolving the inference method of ML-OPE and Online-OPE by OPE1, OPE2 and OPE3, we obtain six new learning algorithms for LDA so called ML-OPE1, ML-OPE2, ML-OPE3, Online-OPE1, Online-OPE2 and Online-OPE3. Thank to rapid convergence of OPE, the result of algorithm is not affected by the number of iterations T [13] (for example, log predictive probability measurements at $T = 20$ and $T = 100$ are asympotic). Therefore, although the number of iteration of methods OPE1, OPE2 and OPE3 is doublely larger than OPE, this does not affect the runtime of inference algorithm. From Theorem 1, we find out that the convergence rate of OPE3 is equal to the one of OPE [13].

ML-OPE1, ML-OPE2, Online-OPE1 and Online-OPE2 algorithms obtain lots of random properties. Inspired by the supervised learning method-Random forest method (By creating many decision trees which have high random nature and taking the average results of these trees, the method is effective in classification). We then make the idea for creating more random natures of these algorithms, and the inference problem (B.2) can jump out of a local maximum to reach close the global one.

ML-OPE3 and Online-OPE3 use the idea of greedy methods, and get better value at each iteration. The greedy algorithms get the best value at every iteration, but the final outcome can not be optimal. OPE3 tries to find maximum by iteration and selects the best optimal point from two ones, so that the value of objective function is higher in each iteration. This is very effective in the case of large-scale machine learning problems. The OPE3 algorithm derived from OPE2 can consider the probability of choosing best point by 1. OPE3 not only archives stable and fast convergence as OPE but also finds better maximum points. So, OPE3 will change the quality of ML-OPE and Online-OPE.

5 Empirical Results

In this section, we investigate the practical performance of our new variants. Since OPE, OPE1, OPE2, OPE3 can play a role in the core subroutine of large-scale learning methods for LDA, we will evaluate the inference algorithms through ML-OPE and Online-OPE based on changing their core inference. By this way, we also see how helpful our new algorithms for posterior inference are. So, thank to changing OPE by our three algorithms in ML-OPE and Online-OPE, we get six new methods for learning LDA. Our simulation results show comparison between the proposed algorithms and the previous ones.

The following two large corpora were used in our experiments: Pubmed consists of 330,000 articles from the Pubmed central; New York Times consists of 300,000 news.[3] To avoid randomness, on each dataset one of learning methods is run for 5 times and reported its average results.

Parameter Settings:

- *Model parameters:* $K = 100, \alpha = \frac{1}{K}, \eta = \frac{1}{K}$
- *Inference parameters:* $T = 20$
- *Learning parameters:* minibatch size $S = |C_t| = 5000, \kappa = 0.9, \tau = 1$

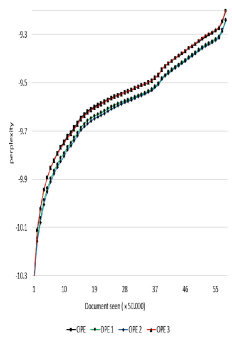
Than and Doan [13] observed that these parameter adapted best for ML-OPE and Online-OPE.

Performance Mesures: We use *NPMI* [11] and *Predictive Probability* [14] to evaluate these learning methods. Log predictive probability measures the predictiveness and generalization of a model to new data, while NPMI evaluates semantics quality of an individual topic in these models.

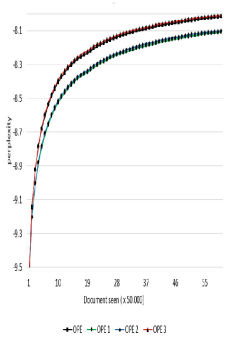
From Figs. 2 and 3, we can see that improved OPE1 and OPE2 making ML-OPE and Online-OPE work worse in both 2 measures, while OPE3 works well with Online-OPE.

Variant of OPE aims to seek θ that maximizes a function $f(\theta)$ on a simplex using stochastic bounds. ML-OPE then updates model parameter β and Online-OPE gets variational element λ. The quality of θ found by OPE directly affects to the quality of $\{\beta, \lambda\}$, then the measures. In practice, OPE performs an fast

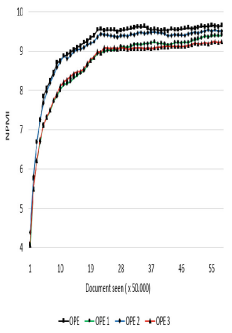
[3] The datasets were taken from http://archive.ics.uci.edu/ml/datasets.



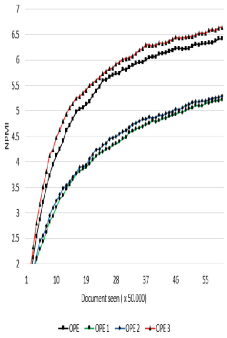
(a) On NYT



(b) On Pubmed



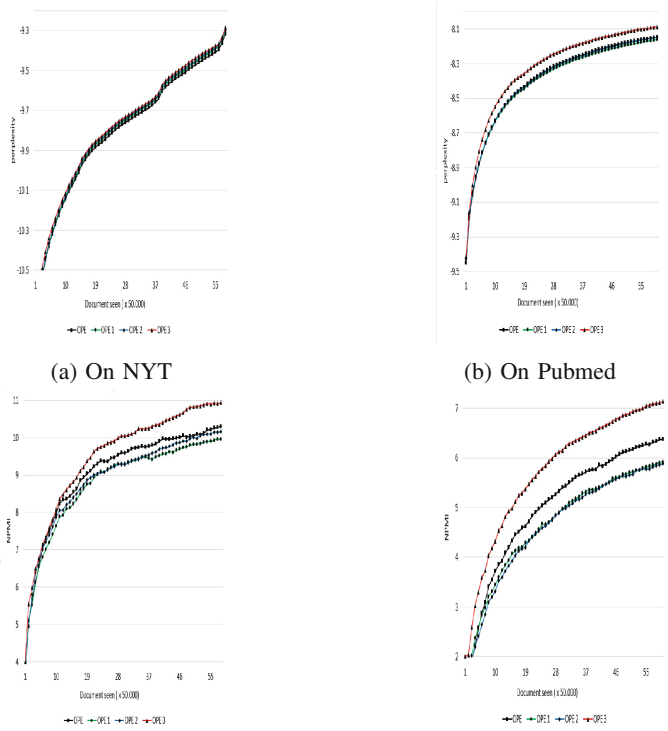
(c) On NYT



(d) On Pubmed

Fig. 2. Performance of ML-OPE when its core inference routine is done by different methods. Note that OPE3 often performs comparably or better than other inference methods.

convergence and stable character. Stability is shown by the number of iterations T. Just $T = 20$ iterations for OPE results in the same predictiveness level as $T = 100$. So, OPE converges very fast. OPE is also stable. Than and Doan did experiments by running 10 times OPE and observed that the results were not too different. OPE1 and OPE2 increase randomness of OPE, but the results of ML-OPE and Online-OPE are not better. With converge property of OPE, it suggests that randomly choose θ_t from $\{\theta_t^u, \theta_t^l\}$ make worse θ from some first iterations, therefore OPE converges fast to worse results. The last improvement makes ML-OPE and Online-OPE work more efficient. This demonstrates our idea of using two random sequences of functions to approximate a objective function. The idea of increasing the randomness and greedy of algorithm are exploited here. Firstly, two random sequences of function are used to rise our participants and information relevant to objective function. Hence in the next iteration we have more choices in θ_t. Secondly, choosing θ_t from $\{\theta_t^u, \theta_t^l\}$ that makes the value of $f(\theta)$ higher in each iteration came from idea of greedy algorithms. There are many ways of choices here, but we designed a best way to create θ_t from $\{\theta_t^u, \theta_t^l\}$.



(a) On NYT (b) On Pubmed

(c) On NYT (d) On Pubmed

Fig. 3. Performance of Online-OPE when its core inference routine is done by different methods. It is easy to see that OPE3 often outperforms the others.

This approach is simple, because it does not need extra parameter. Of course, we can compose θ_t from $\{\theta_t^u, \theta_t^l\}$ by linear combination or something like that. But these approaches make model more complicated. In our experiment, we chose $\theta_t = \alpha\theta_t^u + (1-\alpha)\theta_t^l$ and we get the results better. But we increased the number of parameters in model and we had to choose α empirically. Choosing θ_t in improved OPE3 is more efficient and simpler. OPE can be exploited with general function $f(x)$ if $f(x)$ is a form of $g_1(x) + g_2(x)$. In this paper, we make OPE work better by choose the kind of $g_1(x)$ and $g_2(x) : g_1(x) \leq 0$, $g_2(x) \geq 0$. So for other work using improved OPE3, we can choose the same function $g_1(x), g_2(x)$ and get the better result.

6 Conclusion

We have discussed how posterior inference for individual texts in topic models can be done efficiently. Our novel algorithms (OPE1, OPE2, OPE3) have a theoretical guarantee on quality and convergence rate. In practice, OPE3 can

do inference very fast and effective, and can be easily extended to a wide class of probabilistic models. By exploiting OPE1, OPE2, OPE3 carefully, we have arrived at six efficient methods for learning LDA from data streams or large corpora. As a result, they are good candidates to help us deal with text streams and big data.

Acknowledgments. This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under Grant Number 102.05-2014.28 and by the Air Force Office of Scientific Research (AFOSR), Asian Office of Aerospace Research & Development (AOARD), and US Army International Technology Center, Pacific (ITC-PAC) under Award Number FA2386-15-1-4011.

References

1. Asuncion, A., Welling, M., Smyth, P., Teh, Y.W.: On smoothing and inference for topic models. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 27–34 (2009)
2. Liu, B., Liu, L., Tsykin, A., Goodall, G.J., Green, J.E., Zhu, M., Kim, C.H., Li, J.: Identifying functional miRNA-mRNA regulatory modules with correspondence latent Dirichlet allocation. Bioinformatics **26**(24), 3105 (2010)
3. Sontag, D., Roy, D.M.: Complexity of inference in latent Dirichlet allocation. In: Advances in Neural Information Processing Systems (NIPS) (2011)
4. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**(3), 993–1022 (2003)
5. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (2012)
6. Mimno, D.: Computational historiography: data mining in a century of classics journals. J. Comput. Cult. Heritage **5**(1), 3 (2012)
7. Mimno, D., Hoffman, M.D., Blei, D.M.: Sparse stochastic inference for latent Dirichlet allocation. In: Proceedings of the 29th Annual International Conference on Machine Learning (2012)
8. Schwartz, H.A., Eichstaedt, J.C., Dziurzynski, L., Kern, M.L., Blanco, E., Kosinski, M., Stillwell, D., Seligman, M.E.P., Ungar, L.H.: Toward personality insights from language exploration in social media. In: AAAI Spring Symposium Series (2013)
9. Pritchard, J.K., Stephens, M., Donnelly, P.: Inference of population structure using multilocus genotype data. Genetics **155**(2), 945–959 (2000)
10. Mairal, J.: Stochastic majorization-minimization algorithms for large-scale optimization. In: Advances in Neural Information Processing Systems, pp. 2283–2291 (2013)
11. Lau, J.H., Newman, D., Baldwin, T.: Machine reading tea leaves: automatically evaluating topic coherence and topic model quality. In: Proceedings of the Association for Computational Linguistics, pp. 530–539 (2014)
12. Grimmer, J.: A Bayesian hierarchical topic model for political texts: measuring expressed agendas in senate press releases. Polit. Anal. **18**(1), p1–35 (2010)
13. Than, K., Doan, T.: Guaranteed algorithms for inference in topic models (2015). arXiv preprint arXiv: 1512.03308
14. Hoffman, M.D., Blei, D.M., Wang, C., Paisley, J.: Stochastic variational inference. J. Mach. Learn. Res. **14**(1), 1303–1347 (2013)
15. Aletras, N., Stevenson, M.: Evaluating topic coherence using distributional semantics. In: Proceedings of the 10th International Conference on Computational Semantics, pp. 13–22 (2013)

16. Ghadimi, S., Lan, G.: Stochastic first-and zeroth-order methods for nonconvex stochastic programming. SIAM J. Optim. **23**(4), 2341–2368 (2013)
17. Gerrish, S., Blei, D.: How they vote: issue-adjusted models of legislative behavior. Adv. Neural Inf. Process. Syst. **25**, 2762–2770 (2012)
18. Griffiths, T.L., Steyvers, M.: Finding scientific topics. In: Proceedings of the National Academy of Sciences of the United States of America (2004)
19. Teh, Y.W., Newman, D.M., Welling, A.: Collapsed variational Bayesian inference algorithm for latent Dirichlet allocation. In: Advances in Neural Information Processing Systems, vol. 19, pp. 13–53 (2007)