Online Variational Inference for the Hierarchical Dirichlet Process

Chong Wang **John Paisley** **David M. Blei**
Computer Science Department, Princeton University
{chongw,jpaisley,blei}@cs.princeton.edu

Abstract

The hierarchical Dirichlet process (HDP) is a Bayesian nonparametric model that can be used to model mixed-membership data with a potentially infinite number of components. It has been applied widely in probabilistic topic modeling, where the data are documents and the components are distributions of terms that reflect recurring patterns (or "topics") in the collection. Given a document collection, posterior inference is used to determine the number of topics needed and to characterize their distributions. One limitation of HDP analysis is that existing posterior inference algorithms require multiple passes through all the data—these algorithms are intractable for very large scale applications. We propose an *online* variational inference algorithm for the HDP, an algorithm that is easily applicable to massive and streaming data. Our algorithm is significantly faster than traditional inference algorithms for the HDP, and lets us analyze much larger data sets. We illustrate the approach on two large collections of text, showing improved performance over online LDA, the finite counterpart to the HDP topic model.

1 INTRODUCTION

The hierarchical Dirichlet process (HDP) [1] is a powerful mixed-membership model for the unsupervised analysis of grouped data. Applied to document collections, the HDP provides a nonparametric topic model where documents are viewed as groups of observed words, mixture components (called topics) are distributions over terms, and each document exhibits the topics with different proportions. Given a collection of documents, the HDP topic model finds a low-dimensional latent structure that can be used for tasks like classification, exploration, and summarization. Unlike its finite counterpart, latent Dirichlet allocation [2], the HDP topic model infers the number of topics from the data.

Posterior inference for the HDP is intractable, and much research is dedicated to developing approximate inference algorithms [1, 3, 4]. These methods are limited for massive scale applications, however, because they require multiple passes through the data and are not easily applicable to streaming data.[1] In this paper, we develop a new approximate inference algorithm for the HDP. Our algorithm is designed to analyze much larger data sets than the existing state-of-the-art allows and, further, can be used to analyze *streams* of data. This is particularly apt to the HDP topic model. Topic models promise to help summarize and organize large archives of texts that cannot be easily analyzed by hand and, further, could be better exploited if available on streams of texts such as web APIs or news feeds.

Our method—online variational Bayes for the HDP— was inspired by the recent online variational Bayes algorithm for LDA [7]. Online LDA allows LDA models to be fit to massive and streaming data, and enjoys significant improvements in computation time without sacrificing model quality. Our motivation for extending this algorithm to the HDP is that LDA requires choosing the number of topics in advance. In a traditional setting, where fitting multiple models might be viable, the number of topics can be determined with cross validation or held-out likelihood. However, these techniques become impractical when the data set size is large, and they become impossible when the data are streaming. Online HDP provides the speed of online variational Bayes with the modeling flexibility of the HDP.

The idea behind online variational Bayes in general is to optimize the variational objective function of [8] with stochastic optimization [9]. Optimization proceeds by iteratively taking a random subset of the data, and updating the variational parameters with respect to the subset. Online variational Bayes is particularly efficient when using the natural gradient [10] on models in which traditional variational Bayes

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15. Copyright 2011 by the authors.

[1] One exception that may come to mind is the particle filter [5, 6]. However, this algorithm still requires periodically resampling a variable for every data point. Data cannot be thrown away as in a true streaming algorithm.

can be performed by simple coordinate ascent [11]. (This is the property that allowed [7] to derive an efficient online variational Bayes algorithm for LDA.) In this setting, online variational Bayes is significantly faster than traditional variational Bayes [12], which must make multiple passes through the data.

The challenge we face is that the existing coordinate ascent variational Bayes algorithms for the HDP require complicated approximation methods or numerical optimization [3, 4, 13]. We will begin by reviewing Sethuraman's stick-breaking construction of the HDP [14]. We show that this construction allows for coordinate-ascent variational Bayes without numerical approximation, which is a new and simpler variational inference algorithm for the HDP. We will then use this approach in an online variational Bayes algorithm, allowing the HDP to be applied to massive and streaming data. Finally, on two large archives of scientific articles, we will show that the online HDP topic model provides a significantly better fit than online LDA. Online variational Bayes lets us apply Bayesian nonparametric models at much larger scales.

2 A STICK BREAKING CONSTRUCTION OF THE HDP

We describe the stick-breaking construction of the HDP [14] using the Sethuraman's construction for the DP [15]. This is amenable to simple coordinate-ascent variational inference, and we will use it to develop online variational inference for the HDP.

A two-level hierarchical Dirichlet process (HDP) [1] (the focus of this paper) is a collection of Dirichlet processes (DP) [16] that share a base distribution G_0, which is also drawn from a DP. Mathematically,

$$G_0 \sim \mathrm{DP}(\gamma H) \tag{1}$$

$$G_j \sim \mathrm{DP}(\alpha_0 G_0), \text{ for each } j, \tag{2}$$

where j is an index for each group of data. A notable feature of the HDP is that all DPs G_j share the same set of atoms and only the atom weights differ. This is a result of the almost sure discreteness of the top-level DP.

In the HDP topic model—which is the focus of this paper—we model groups of words organized into documents. The variable w_{jn} is the nth word in the jth document; the base distribution H is a symmetric Dirichlet over the vocabulary simplex; and the atoms of G_0, which are independent draws from H, are called topics.

The HDP topic model contains two additional steps to generate the data. First we generate the topic associated with the nth word in the jth document; then we generate the word from that topic,

$$\theta_{jn} \sim G_j, \quad w_{jn} \sim \mathrm{Mult}(\theta_{jn}). \tag{3}$$

The discreteness of the corpus-level draw G_0 ensures that all documents share the same set of topics. The document-level draw G_j inherits the topics from G_0, but weights them according to document-specific topic proportions.

Teh's Stick-breaking Construction. The definition of the HDP in Eq. 1 is implicit. [1] propose a more constructive representation of the HDP using two stick-breaking representations of a Dirichlet distribution [15]. For the corpus-level DP draw, this representation is

$$\begin{aligned}
\beta'_k &\sim \mathrm{Beta}(1, \gamma), \\
\beta_k &= \beta'_k \textstyle\prod_{l=1}^{k-1}(1-\beta'_l), \\
\phi_k &\sim H, \\
G_0 &= \textstyle\sum_{k=1}^{\infty} \beta_k \delta_{\phi_k}.
\end{aligned} \tag{4}$$

Thus, G_0 is discrete and has support at the atoms $\boldsymbol{\phi} = (\phi_k)_{k=1}^{\infty}$ with weights $\boldsymbol{\beta} = (\beta_k)_{k=1}^{\infty}$. The distribution for $\boldsymbol{\beta}$ is also written as $\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$ [17].

The construction of each document-level G_j is

$$\begin{aligned}
\pi'_{jk} &\sim \mathrm{Beta}\left(\alpha_0 \beta_k, \alpha_0 \left(1 - \textstyle\sum_{l=1}^{k} \beta_l\right)\right), \\
\pi_{jk} &= \pi'_{jk} \textstyle\prod_{l=1}^{k-1}(1-\pi'_{jl}), \\
G_j &= \textstyle\sum_{k=1}^{\infty} \pi_{jk} \delta_{\phi_k},
\end{aligned} \tag{5}$$

where $\boldsymbol{\phi} = (\phi_k)_{k=1}^{\infty}$ are the same atoms as G_0 in Eq. 4.

This construction is difficult to use in an online variational inference algorithm. Online variational inference is particularly efficient when the model is also amenable to coordinate ascent variational inference, and where each update is available in closed form. In the construction above, the stick-breaking weights are tightly coupled between the bottom and top-level DPs. As a consequence, it is not amendable to closed form variational updates [3, 4].

Sethuraman's Stick-breaking Construction. To address this issue, we describe an alternative stick-breaking construction for the HDP that allows for closed-form coordinate-ascent variational inference due to its full conjugacy. (This construction was also briefly described in [14].)

The construction is formed by twice applying Sethuraman's stick-breaking construction of the DP. We again construct the corpus-level base distribution G_0 as in Eq. 4. The difference is in the document-level draws. We use Sethuraman's construction for each G_j,

$$\begin{aligned}
\psi_{jt} &\sim G_0, \\
\pi'_{jt} &\sim \mathrm{Beta}(1, \alpha_0), \\
\pi_{jt} &= \pi'_{jt} \textstyle\prod_{l=1}^{t-1}(1-\pi'_{jl}), \\
G_j &= \textstyle\sum_{t=1}^{\infty} \pi_{jt} \delta_{\psi_{jt}},
\end{aligned} \tag{6}$$

Notice that each document-level atom (i.e., topic) ψ_{jt} maps to a corpus-level atom ϕ_k in G_0 according to the distribution

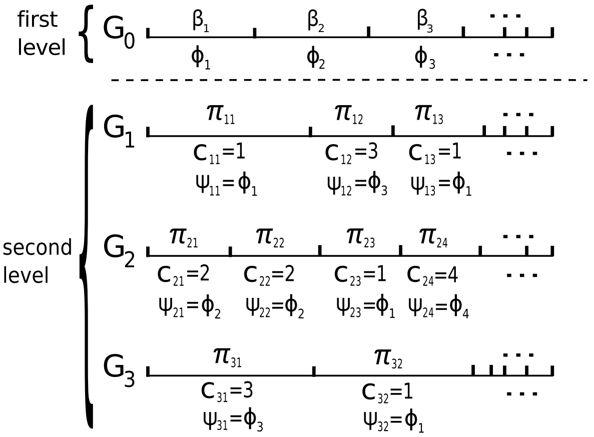

Figure 1: Illustration of the Sethuraman's stick-breaking construction of the two-level HDP. In the first level, $\phi_k \sim H$ and $\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$; in the second level, $\boldsymbol{\pi}_j \sim \mathrm{GEM}(\alpha_0)$, $c_{jt} \sim \mathrm{Mult}(\boldsymbol{\beta})$ and $\psi_{jt} = \phi_{c_{jt}}$.

defined by G_0. Further note there will be multiple document-level atoms ψ_{jt} which map to the same corpus-level atom ϕ_k, but we can verify that G_j contains all of the atoms in G_0 almost surely.

A second way to represent the document-level atoms $\boldsymbol{\psi}_j = (\psi_{jt})_{t=1}^{\infty}$ is to introduce a series of indicator variables, $\boldsymbol{c}_j = (c_{jt})_{t=1}^{\infty}$, which are drawn i.i.d.,

$$c_{jt} \sim \mathrm{Mult}(\boldsymbol{\beta}), \tag{7}$$

where $\boldsymbol{\beta} \sim \mathrm{GEM}(\gamma)$ (as mentioned above). Then let

$$\psi_{jt} = \phi_{c_{jt}}, \tag{8}$$

Thus, we do not need to explicitly represent the document atoms $\boldsymbol{\psi}_j$. This further simplifies online inference.

The property that multiple document-level atoms ψ_{jt} can map to the same corpus-level atom ϕ_k in this representation is similar in spirit to the Chinese restaurant franchise (CRF) [1], where each restaurant can have multiple tables serving the *same* dish ϕ_k. In the CRF representation, a hierarchical Chinese restaurant process allocates dishes to tables. Here, we use a series of random indicator variables $\boldsymbol{c}_j$ to represent this structure. Figure 1 illustrates the concept.

Given the representation in Eq. 6, the generative process for the observed words in jth document, w_{jn}, is as follows,

$$z_{jn} \sim \mathrm{Mult}(\boldsymbol{\pi}_j), \tag{9}$$

$$\theta_{jn} = \psi_{jz_{jn}} = \phi_{c_{jz_{jn}}}, \tag{10}$$

$$w_{jn} \sim \mathrm{Mult}(\theta_{jn}). \tag{11}$$

The indicator z_{jn} selects topic parameter ψ_{jt}, which maps to one topic ϕ_k through the indicators $\boldsymbol{c}_j$. This also provides the mapping from topic θ_{jn} to ϕ_k, which we need in Eq. 3.

3 ONLINE VARIATIONAL INFERENCE FOR THE HDP

With Sethuraman's construction of the HDP in hand, we now turn to our original aim—approximate posterior inference in the HDP for massive and streaming data. Given a large collection of documents, our goal is to approximate the posterior distribution of its latent topic structure.

We will use online variational inference [11]. Traditional variational inference approximates the posterior over the hidden variables by positing a simpler distribution which is optimized to be close in Kullback-Leibler (KL) divergence to the true posterior [8]. This problem is (approximately) solved by optimizing a function equal up to a constant to the KL of interest. In online variational inference, we optimize that function with stochastic approximation.

Online variational inference enjoys a close relationship with *coordinate-ascent variational inference*. Consider a model with latent variables and observations for which the posterior is intractable to compute. One strategy for variational inference is the mean-field approach: posit a distribution where each latent variable is independent and governed by its own parameter, and optimize the variational parameters with coordinate ascent.

Now, suppose that those coordinate ascent updates are available in closed form and consider updating them in parallel. (Note this is no longer coordinate ascent.) It turns out that the vector of parallel coordinate updates is exactly the natural gradient of the variational objective function under conjugate priors [11]. This insight makes stochastic optimization of the variational objective, based on a subset of the data under analysis, a simple and efficient alternative to traditional coordinate-ascent.

Let us now return to the HDP topic model. We will first show that Sethuraman's representation of the HDP above allows for closed-form coordinate-ascent updates for variational inference. Then, we will derive the corresponding online algorithm, which provides a scalable method for HDP posterior inference.

3.1 A New Coordinate-ascent Variational Inference

When applied to Bayesian nonparametric models, variational methods are usually based on stick-breaking representations—these representations provide a concrete set of hidden variables on which to place an approximate posterior [18, 19, 3]. Furthermore, the approximate posterior is usually truncated. The user first sets a truncation on the number of topics to allow, and then relies on variational inference to infer a smaller number that are used in the data. (Two exceptions are found in [20, 21], who developed methods that allow the truncation to grow.) Note that setting a truncation level is different from asserting a number of components in a model. When set large, the

HDP assumptions encourage the approximate posterior to use fewer components.

We use a fully factorized variational distribution and perform mean-field variational inference. The hidden variables that we are interested in are the top-level stick proportions $\boldsymbol{\beta}' = (\beta'_k)_{k=1}^{\infty}$, bottom-level stick proportions $\boldsymbol{\pi}'_j = (\pi'_{jt})_{t=1}^{\infty}$ and the vector of indicators $\boldsymbol{c}_j = (c_{jt})_{t=1}^{\infty}$ for each G_j. We also infer atom/topic distributions $\boldsymbol{\phi} = (\phi_k)_{k=1}^{\infty}$, topic index z_{jn} for each word w_{jn}. Thus our variational distribution has the following form,

$$q(\boldsymbol{\beta}', \boldsymbol{\pi}', \boldsymbol{c}, \boldsymbol{z}, \boldsymbol{\phi}) = q(\boldsymbol{\beta}')q(\boldsymbol{\pi}')q(\boldsymbol{c})q(\boldsymbol{z})q(\boldsymbol{\phi}). \qquad (12)$$

This further factorizes into

$$\begin{aligned} q(\boldsymbol{c}) &= \textstyle\prod_j \prod_t q(c_{jt}|\varphi_{jt}), \\ q(\boldsymbol{z}) &= \textstyle\prod_j \prod_n q(z_{jn}|\zeta_{jn}), \\ q(\boldsymbol{\phi}) &= \textstyle\prod_k q(\phi_k|\lambda_k), \end{aligned}$$

where the variational parameters are φ_{jt} (multinomial), ζ_{jn} (multinomial) and λ_k (Dirichlet). The factorized forms of $q(\boldsymbol{\beta}')$ and $q(\boldsymbol{\pi}')$ are

$$\begin{aligned} q(\boldsymbol{\beta}') &= \textstyle\prod_{k=1}^{K-1} q(\beta'_k|u_k, v_k), \\ q(\boldsymbol{\pi}') &= \textstyle\prod_j \prod_{t=1}^{T-1} q(\pi'_{jt}|a_{jt}, b_{jt}), \end{aligned} \qquad (13)$$

where (u_k, b_k) and (a_{jt}, b_{jt}) are parameters of beta distributions. We set the truncations for the corpus and document levels to K and T. Here, T can be set much smaller than K, because in practice each document G_j requires far fewer topics than those needed for the entire corpus (i.e., the atoms of G_0). With this truncation, our variational distribution has $q(\beta'_K = 1) = 1$ and $q(\pi'_{jT} = 1) = 1$, for all j.

Using standard variational theory [8], we lower bound the marginal log likelihood of the observed data $\mathcal{D} = (\boldsymbol{w}_j)_{j=1}^{D}$ using Jensen's inequality,

$$\begin{aligned} &\log p(\mathcal{D}|\gamma, \alpha_0, \eta) \geq \mathbb{E}_q\left[\log p(\mathcal{D}, \boldsymbol{\beta}', \boldsymbol{\pi}', \boldsymbol{c}, \boldsymbol{z}, \boldsymbol{\phi})\right] + H(q) \\ &= \textstyle\sum_j \left\{\mathbb{E}_q\left[\log\left(p(\boldsymbol{w}_j|\boldsymbol{c}_j, \boldsymbol{z}_j, \boldsymbol{\phi})p(\boldsymbol{c}_j|\boldsymbol{\beta}')p(\boldsymbol{z}_j|\boldsymbol{\pi}'_j)p(\boldsymbol{\pi}'_j|\alpha_0)\right)\right]\right. \\ &\qquad \left. + H(q(\boldsymbol{c}_j)) + H(q(\boldsymbol{z}_j)) + H(q(\boldsymbol{\pi}'_j))\right\} \\ &\quad + \mathbb{E}_q\left[\log p(\boldsymbol{\beta}')p(\boldsymbol{\phi})\right] + H(q(\boldsymbol{\beta}')) + H(q(\boldsymbol{\phi})) \\ &= \mathcal{L}(q), \end{aligned} \qquad (14)$$

where $H(\cdot)$ is the entropy term for the variational distribution. This is the variational objective function, which up to a constant is equivalent to the KL to the true posterior. Taking derivatives of this lower bound with respect to each variational parameter, we can derive the following coordinate ascent updates.

Document-level Updates: At the document level we update the parameters to the per-document stick, the parameters to the per word topic indicators, and the parameters to the per document topic indices,

$$a_{jt} = 1 + \textstyle\sum_n \zeta_{jnt}, \qquad (15)$$

$$b_{jt} = \alpha_0 + \textstyle\sum_n \sum_{s=t+1}^{T} \zeta_{jns}, \qquad (16)$$

$$\varphi_{jtk} \propto \exp\left(\textstyle\sum_n \zeta_{jnt}\mathbb{E}_q\left[\log p(w_{jn}|\phi_k)\right] + \mathbb{E}_q\left[\log \beta_k\right]\right), \qquad (17)$$

$$\zeta_{jnt} \propto \exp\left(\textstyle\sum_{k=1}^{K} \varphi_{jtk}\mathbb{E}_q\left[\log p(w_{jn}|\phi_k)\right] + \mathbb{E}_q\left[\log \pi_{jt}\right]\right). \qquad (18)$$

Corpus-level Updates: At the corpus level, we update the parameters to top-level sticks and the topics,

$$u_k = 1 + \textstyle\sum_j \sum_{t=1}^{T} \varphi_{jtk}, \qquad (19)$$

$$v_k = \gamma + \textstyle\sum_j \sum_{t=1}^{T} \sum_{l=k+1}^{K} \varphi_{jtl}, \qquad (20)$$

$$\lambda_{kw} = \eta + \textstyle\sum_j \sum_{t=1}^{T} \varphi_{jtk}\left(\sum_n \zeta_{jnt} I[w_{jn} = w]\right), \qquad (21)$$

The expectations involved above are taken under the variational distribution q, and are

$$\begin{aligned} &\mathbb{E}_q\left[\log \beta_k\right] = \mathbb{E}_q\left[\log \beta'_k\right] + \textstyle\sum_{l=1}^{k-1} \mathbb{E}_q\left[\log(1 - \beta'_l)\right], \\ &\mathbb{E}_q\left[\log \beta'_k\right] = \Psi(u_k) - \Psi(u_k + v_k), \\ &\mathbb{E}_q\left[\log(1 - \beta'_k)\right] = \Psi(v_k) - \Psi(u_k + v_k), \\ &\mathbb{E}_q\left[\log \pi_{jt}\right] = \mathbb{E}_q\left[\log \pi'_{jt}\right] + \textstyle\sum_{s=1}^{t-1} \mathbb{E}_q\left[\log(1 - \pi'_{js})\right], \\ &\mathbb{E}_q\left[\log \pi'_{jt}\right] = \Psi(a_{jt}) - \Psi(a_{jt} + b_{jt}), \\ &\mathbb{E}_q\left[\log(1 - \pi'_{jt})\right] = \Psi(b_{jt}) - \Psi(a_{jt} + b_{jt}), \\ &\mathbb{E}_q\left[\log p(w_{jn} = w|\phi_k)\right] = \Psi(\lambda_{kw}) - \Psi(\textstyle\sum_w \lambda_{kw}), \end{aligned}$$

where $\Psi(\cdot)$ is the digamma function.

Unlike previous variational inference methods for the HDP [3, 4], this method only contains simple closed-form updates due to the full conjugacy of the stick-breaking construction. (We note that, even in the batch setting, this is a new posterior inference algorithm for the HDP.)

3.2 Online Variational Inference

We now develop online variational inference for an HDP topic model. In online variational inference, we apply stochastic optimization to the variational objective. We subsample the data (in this case, documents), compute an approximation of the gradient based on the subsample, and follow that gradient with a decreasing step-size. The key insight behind efficient online variational inference is that coordinate ascent updates applied in parallel precisely form the *natural gradient* of the variational objective function [11, 7].

Our approach is similar to that described in [7]. Let D be the total number of documents in the corpus, and define the variational lower bound for document j as

$$\begin{aligned} \mathcal{L}_j &= \mathbb{E}_q\left[\log\left(p(\boldsymbol{w}_j|\boldsymbol{c}_j, \boldsymbol{z}_j, \boldsymbol{\phi})p(\boldsymbol{c}_j|\boldsymbol{\beta}')p(\boldsymbol{z}_j|\boldsymbol{\pi}'_j)p(\boldsymbol{\pi}'_j|\alpha_0)\right)\right] \\ &\quad + H(q(\boldsymbol{c}_j)) + H(q(\boldsymbol{z}_j)) + H(q(\boldsymbol{\pi}'_j)) \\ &\quad + \tfrac{1}{D}\left[\mathbb{E}_q\left[\log p(\boldsymbol{\beta}')p(\boldsymbol{\phi})\right] + H(q(\boldsymbol{\beta}')) + H(q(\boldsymbol{\phi}))\right]. \end{aligned}$$

We have taken the corpus-wide terms and multiplied them by $1/D$. With this expression, we can see that the lower bound $\mathcal{L}$ in Eq. 14 can be written as

$$\mathcal{L} = \textstyle\sum_j \mathcal{L}_j = \mathbb{E}_j[D\mathcal{L}_j],$$

where the expectation is taken over the empirical distribution of the data set. The expression $D\mathcal{L}_j$ is the variational lower bound evaluated with D duplicate copies of document j.

With the objective construed as an expectation over our data, online HDP proceeds as follows. Given the existing corpus-level parameters, we first sample a document j and compute its optimal document-level variational parameters $(\boldsymbol{a}_j, \boldsymbol{b}_j, \boldsymbol{\psi}_j, \boldsymbol{\zeta}_j)$ by coordinate ascent (see Eq. 15 to 18.). Then, take the gradient of the corpus-level parameters $(\boldsymbol{\lambda}, \boldsymbol{u}, \boldsymbol{v})$ of $D\mathcal{L}_j$, which is a noisy estimate of the gradient of the expectation above. We follow that gradient according to a decreasing learning rate, and repeat.

Natural Gradients. The gradient of the variational objective contains, as a component, the covariance matrix of the variational distribution. This is a computational problem in topic modeling because each set of topic parameters involves a $V \times V$ covariance matrix, where V is the size of the vocabulary (e.g., 5,000). The *natural gradient* [10]—which is the inverse of the Riemannian metric multiplied by the gradient—has a simple form in the variational setting [11] that allows for fast online inference.

Multiplying the gradient by the inverse of Riemannian metric cancels the covariance matrix of the variational distribution, leaving a natural gradient which is much easier to work with. Specifically, the natural gradient is structurally equivalent to the coordinate updates of Eq 19 to 21 taken in parallel. (And, in stochastic optimization, we treat the sampled document j as though it is the whole corpus.) Let $\partial\boldsymbol{\lambda}(j)$, $\partial\boldsymbol{u}(j)$ and $\partial\boldsymbol{v}(j)$ be the natural gradients for $D\mathcal{L}_j$. Using the analysis in [11, 7], the components of the natural gradients are

$$\partial\lambda_{kw}(j) = -\lambda_{kw} + \eta + D \textstyle\sum_{t=1}^{T} \varphi_{jtk} \left(\sum_n \zeta_{jnt} I[w_{jn} = w]\right), \tag{22}$$

$$\partial u_k(j) = -u_k + 1 + D \textstyle\sum_{t=1}^{T} \varphi_{jtk}, \tag{23}$$

$$\partial v_k(j) = -v_k + \gamma + D \textstyle\sum_{t=1}^{T} \sum_{l=k+1}^{K} \varphi_{jtl}. \tag{24}$$

In online inference, an appropriate learning rate ρ_{t_o} is needed to ensure the parameters to converge to a stationary point [11, 7]. Then the updates of $\boldsymbol{\lambda}$, $\boldsymbol{u}$ and $\boldsymbol{v}$ become

$$\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \rho_{t_o} \partial\boldsymbol{\lambda}(j), \tag{25}$$

$$\boldsymbol{u} \leftarrow \boldsymbol{u} + \rho_{t_o} \partial\boldsymbol{u}(j) \tag{26}$$

$$\boldsymbol{v} \leftarrow \boldsymbol{v} + \rho_{t_o} \partial\boldsymbol{v}(j), \tag{27}$$

where the learning rate ρ_{t_o} should satisfy

$$\textstyle\sum_{t_o=1}^{\infty} \rho_{t_o} = \infty, \quad \sum_{t_o=1}^{\infty} \rho_{t_o}^2 < \infty, \tag{28}$$

```
Initialize $\boldsymbol{\lambda} = (\lambda_k)_{k=1}^{K}$, $\boldsymbol{u} = (u_k)_{k=1}^{K-1}$ and $\boldsymbol{v} = (v_k)_{k=1}^{K-1}$ randomly. Set $t_o = 1$.
while Stopping criterion is not met do
    Fetch a random document $j$ from the corpus.
    Compute $\boldsymbol{a}_j$, $\boldsymbol{b}_j$, $\boldsymbol{\varphi}_j$ and $\boldsymbol{\zeta}_j$ using variational inference using document-level updates, Eq. 15 to 18.
    Compute the natural gradients, $\partial\boldsymbol{\lambda}(j)$, $\partial\boldsymbol{u}(j)$ and $\partial\boldsymbol{v}(j)$ using Eq. 22 to 24.
    Set $\rho_{t_o} = (\tau_0 + t_o)^{-\kappa}$, $t_o \leftarrow t_o + 1$.
    Update $\boldsymbol{\lambda}$, $\boldsymbol{u}$ and $\boldsymbol{v}$ using Eq. 25 to 27.
end while
```

Figure 2: Online variational inference for the HDP

which ensures convergence [9]. In our experiments, we use $\rho_{t_o} = (\tau_0 + t_o)^{-\kappa}$, where $\kappa \in (0.5, 1]$ and $\tau_0 > 0$. Note that the natural gradient is essential to the efficiency of the algorithm. The online variational inference algorithm for the HDP topic model is illustrated in Figure 2.

Mini-batches. To improve stability of the online learning algorithm, practitioners typically use multiple samples to compute gradients at a time—a small set of documents in our case. Let $\mathcal{S}$ be a small set of documents and $S = |\mathcal{S}|$ be its size. In this case, rather than computing the natural gradients using $D\mathcal{L}_j$, we use $(D/S)\sum_{j \in \mathcal{S}} \mathcal{L}_j$. The update equations can then be similarly derived.

4 EXPERIMENTAL RESULTS

In this section, we evaluate the performance of online variational HDP compared with batch variational HDP and online variational LDA.[2]

4.1 Data and Metric

Data Sets. Our experiments are based on two datasets:

- *Nature*: This dataset contains 352,549 documents, with about 58 million tokens and a vocabulary size of 4,253. These articles are from the years 1869 to 2008.
- *PNAS*: The *Proceedings of the National Academy of Sciences* (PNAS) dataset contains 82,519 documents, with about 46 million tokens and a vocabulary size of 6,500. These articles are from the years 1914 to 2004.

Standard stop words and those words that appear too frequently or too rarely are removed.

Evaluation Metric. We use the following evaluation metric to compare performance. For each dataset, we held out 2000 documents as a test set $\mathcal{D}_{\text{test}}$, with the remainder as training data $\mathcal{D}_{\text{train}}$. For testing, we split document $\boldsymbol{w}_j$ in $\mathcal{D}_{\text{test}}$ into two parts, $\boldsymbol{w}_j = (\boldsymbol{w}_{j1}, \boldsymbol{w}_{j2})$, and compute

[2] http://www.cs.princeton.edu/~blei/downloads/onlineldavb.tar

the predictive likelihood of the second part $\boldsymbol{w}_{j2}$ (10% of the words) conditioned on the first part $\boldsymbol{w}_{j1}$ (90% of the words) and on the training data. This is similar to the metrics used in [3, 22], which tries to avoid comparing different hyperparameters. The metric is

$$\text{likelihood}_{\text{pw}} = \frac{\sum_{\boldsymbol{j} \in \mathcal{D}_{\text{test}}} \log p(\boldsymbol{w}_{j2} | \boldsymbol{w}_{j1}, \mathcal{D}_{\text{train}})}{\sum_{\boldsymbol{j} \in \mathcal{D}_{\text{test}}} |\boldsymbol{w}_{j2}|},$$

where $|\boldsymbol{w}_{j2}|$ is the number of tokens in $\boldsymbol{w}_{j2}$ and "pw" means "per-word." Exact computation is intractable, and so we use the following approximation. For all algorithms, let $\bar{\boldsymbol{\phi}}$ be the variational expectation of $\boldsymbol{\phi}$ given $\mathcal{D}_{\text{train}}$. For LDA, let $\bar{\boldsymbol{\pi}}_j$ be the variational expectation given $\boldsymbol{w}_{j2}$ and $\boldsymbol{\alpha}$ be its Dirichlet hyperparameter for topic proportions. The predictive marginal probability of $\boldsymbol{w}_{j1}$ is approximated by

$$p(\boldsymbol{w}_{j2} | \boldsymbol{w}_{j1}, \mathcal{D}_{\text{train}}) \approx \textstyle\prod_{w \in \boldsymbol{w}_{j2}} \sum_k \bar{\pi}_{jk} \bar{\phi}_{kw}.$$

To use this approximation for the HDP, we set the Dirichlet hyperparameter to $\bar{\boldsymbol{\alpha}} = \alpha_0 \bar{\boldsymbol{\beta}}$, where $\bar{\boldsymbol{\beta}}$ is the variational expectation of $\boldsymbol{\beta}$, obtained from the variational expectation of $\boldsymbol{\beta}'$.

4.2 Results

Experimental Settings. For the HDP, we set $\gamma = \alpha_0 = 1$, although using priors is also an option. We set the top-level truncation $K = 150$ and the second level truncation $T = 15$. Here $T \ll K$, since documents usually don't have many topics. For online variational LDA, we set its Dirichlet hyperparameter $\boldsymbol{\alpha} = (1/K, \dots, 1/K)$, where K is the number of topics; we set $K = \{20, 40, 60, 80, 100, 150\}$.[3] We set $\tau_0 = 64$ based on the suggestions in [7], and vary $\kappa = \{0.6, 0.8, 1.0\}$ and the batch size $S = \{16, 64, 256, 1024, 2048\}$. We collected experimental results during runs of 6 hours each.[4]

Nature Corpus. In Figure 3, we plot the per-word log likelihood as a function of computation time for online HDP, online LDA, and batch HDP. (For the online algorithms, we set $\kappa = 0.6$ and the batch size was $S = 256$.) This figure shows that online HDP performs better than online LDA. The HDP uses about 110 topics out of its potential 150. In contrast, online LDA uses almost all the topics and exhibits overfitting at 150 topics. Note that batch HDP is only trained on a subset of 20,000 documents—otherwise it is too slow—and its performance suffers.

In Figure 4, we plot the per-word likelihood after 6 hours of computation, exploring the effect of batch size and values of κ. We see that, overall, online HDP performs better than online LDA. (This matches the reported results in [3], which compares batch variational inference for the HDP and LDA.) Further, we found that small κ favors larger batch sizes. (This matches the results seen for online LDA in [7].)

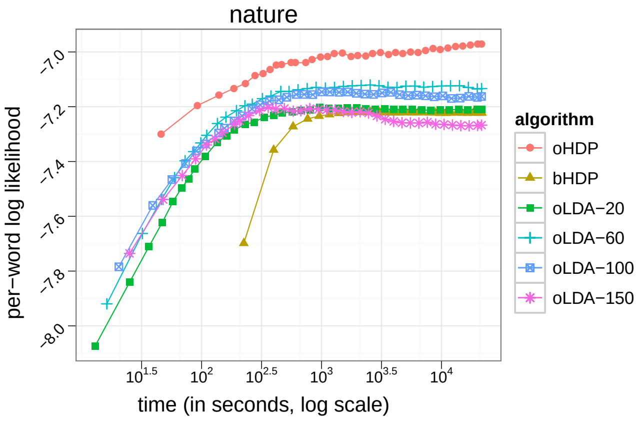

Figure 3: Experimental results on *Nature* with $\kappa = 0.6$ and batch size $S = 256$ (for the online algorithms). Points are sub-sampled for better view. The label "oLDA-20" indicates online LDA with 20 topics. (Not all numbers of topics are shown; see Figure 4 for more details.) Online HDP performs better than online LDA and batch HDP.

We also ran online HDP on the full *Nature* dataset using only one pass (with $\kappa = 0.6$ and a batch size $S = 1024$) by sequentially processing the articles from the year 1869 to 2008. Table 1 tracks the most probable ten words from two topics as we encounter more articles in the collection. Note that the HDP here is not a dynamic topic model [23, 24]; we show these results to demonstrate the online inference process.

These results show that online inference for streaming data finds different topics at different speeds, since the relevant information for each topic does not come at the same time. In this sequential setting, some topics are rarely used until there are documents that can provide enough information to update them (see the top topic in Table 1). Other topics are updated throughout the stream because relevant documents occur throughout the whole collection (see the bottom topic in Table 1).

PNAS Corpus We ran the same experiments on the *PNAS* corpus. Since *PNAS* is smaller than *Nature*, we were able to run batch HDP on the whole data set. Figure 5 shows the result with $\kappa = 0.6$ and batch size $S = 2048$. Online HDP performs better than online LDA. Here batch HDP performs a little better than online HDP, but online HDP is much faster. Figure 6 plots the comparison between online HDP and online LDA across different batch sizes and values of κ.

5 DISCUSSION

We developed an online variational inference algorithm for the hierarchical Dirichlet process topic model. Our algo-

[3] This is different from the top level truncation K in the HDP.

[4] The python package will be available at first author's homepage.

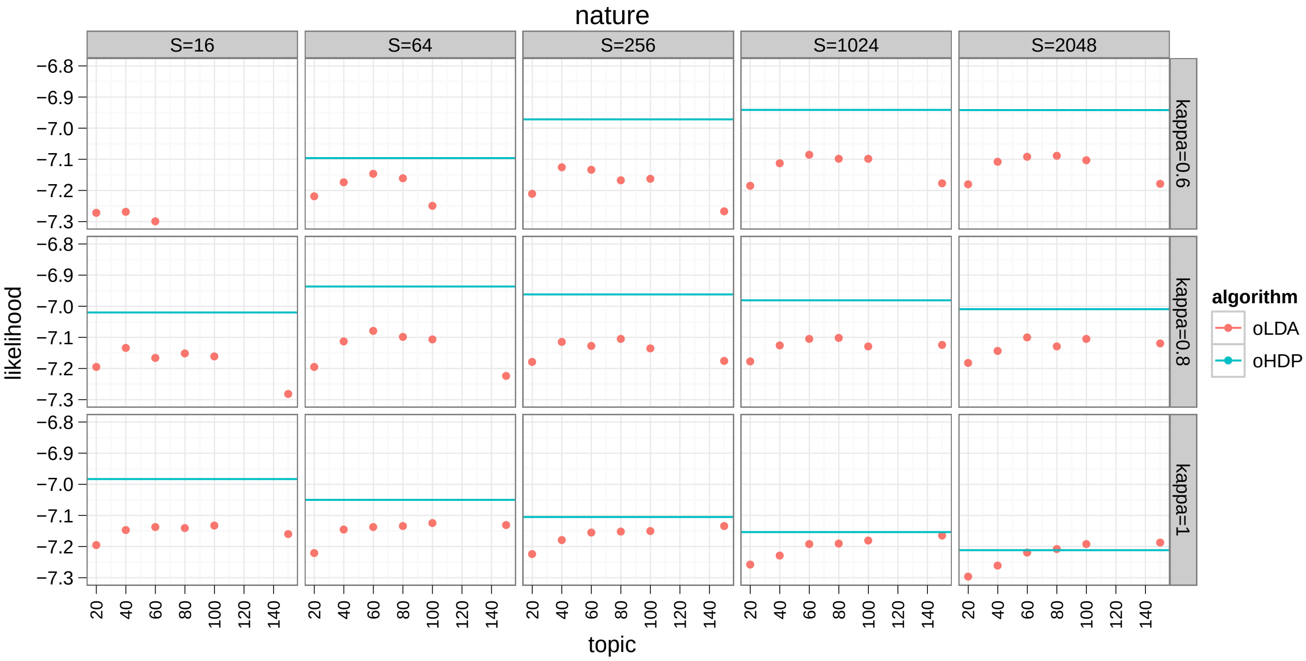

Figure 4: Comparisons of online LDA and online HDP on the *Nature* corpus under various settings of batch size S and parameter κ (kappa), run for 6 hours each. (Some lines for online HDP and points for online LDA do not appear due to figure limits.) The best result among all is achieved by online HDP.

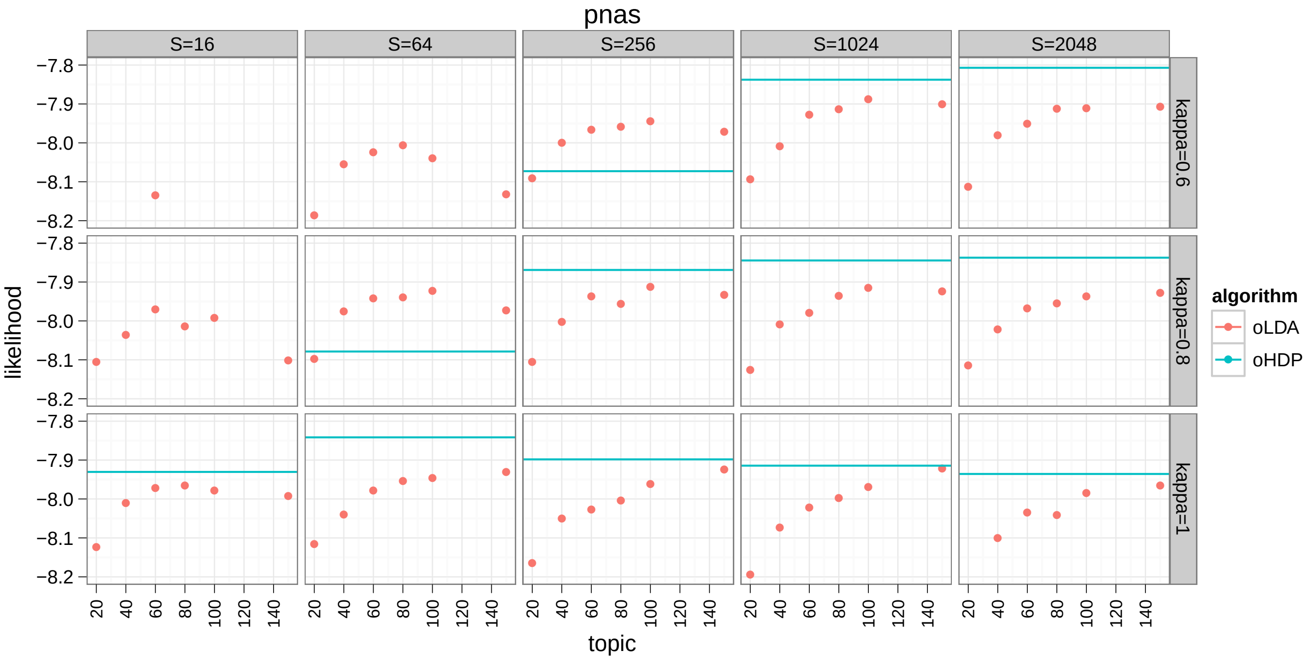

Figure 6: Comparisons of online LDA and online oHDP on the *PNAS* corpus under various settings of batch size S and parameter κ (kappa), run for 6 hours each. (Some lines for online HDP and points for online LDA do not appear due to figure limits.) The best result among all is achieved by online HDP.

40,960	81,920	122,880	163,840	204,800	245,760	286,720	327,680	352,549
author	author	due	weight	rats	rats	rats	rats	neurons
series	series	series	due	response	mice	response	brain	rats
vol	due	distribution	birds	blood	response	dose	memory	memory
due	sea	author	response	sec	dose	saline	dopamine	brain
your	vol	sea	series	weight	drug	injection	mice	dopamine
latter	latter	carried	average	dose	brain	brain	subjects	response
think	hand	statistical	sample	mice	injection	females	neurons	mice
sun	carried	sample	soil	average	food	treated	drug	behavioural
sea	fact	average	population	food	saline	food	induced	training
feet	appear	soil	frequency	controls	females	rat	response	responses
stars	stars	stars	stars	stars	stars	star	stars	galaxies
star	observatory	observatory	observatory	observatory	observatory	arc	galaxy	stars
observatory	star	sun	solar	solar	radio	emission	star	galaxy
sun	sun	star	sun	sun	star	stars	emission	star
magnitude	magnitude	solar	astronomical	astronomical	optical	optical	galaxies	emission
solar	solar	astronomical	star	star	objects	spectrum	optical	optical
comet	motion	greenwich	greenwich	earth	magnitude	image	redshift	redshift
spectrum	comet	earth	eclipse	radio	solar	images	images	spectrum
motion	eclipse	eclipse	instrument	greenwich	positions	ray	image	images
photographs	spectrum	magnitude	royal	motion	plates	magnitude	objects	objects

Table 1: The top ten words from two topics, displayed after different numbers of documents have been processed for inference. The two topics are separated by the dashed line. The first line of the table indicates the number of articles seen so far (beginning from the year 1869). The topic on the top (which could be labeled "neuroscience research on rats") does not have a clear meaning until we have analyzed 204,800 documents. This topic is rarely used in the earlier part of the corpus and few documents provide useful information about it. In contrast, the topic on the bottom (which could be labeled "astronomy research") has a clearer meaning from the beginning. This subject is discussed earlier in *Nature* history.

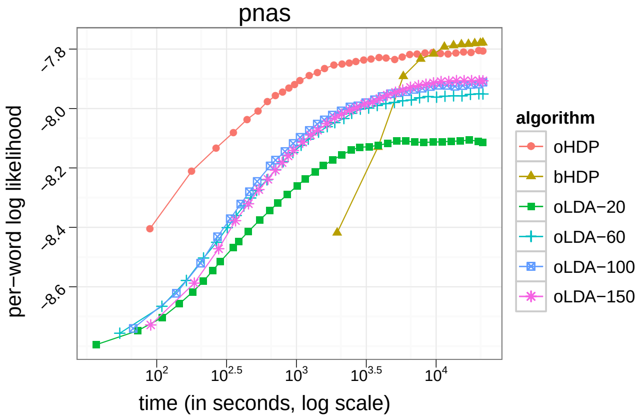

Figure 5: Experimental results on *PNAS* $\kappa = 0.6$ and batch size $S = 2048$ (for the online algorithms). Points are subsampled for better view. (Not all numbers of topics are shown, please see Figure 6 for more details.) Online HDP performs better than online LDA, and slightly worse than batch HDP. Unlike in the *Nature* experiment, batch HDP is trained on the whole training set.

rithm is based on a stick-breaking construction of the HDP that allows for closed-form coordinate ascent variational inference, which is a key factor in developing the online algorithm. Our experimental results show that for large-scale applications, the online variational inference for the HDP can address the model selection problem for LDA and avoid overfitting.

The application of natural gradient learning to online variational inference may be generalized to other Bayesian nonparametric models, as long as we can construct variational inference algorithms with closed form updates under conjugacy. For example, the Indian Buffet process (IBP) [25, 26, 27] might be another model that can use an efficient online variational inference algorithm for large and streaming data sets.

Acknowledgement. Chong Wang is supported by Google PhD fellowship. David M. Blei is supported by ONR 175-6343, NSF CAREER 0745520, AFOSR 09NL202, the Alfred P. Sloan foundation, and a grant from Google.

References

[1] Teh, Y., M. Jordan, M. Beal, et al. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2007.

[2] Blei, D., A. Ng, M. Jordan. Latent Dirichlet allocation.

Journal of Machine Learning Research, 3:993–1022, 2003.

[3] Teh, Y., K. Kurihara, M. Welling. Collapsed variational inference for HDP. In *Neural Information Processing Systems*. 2007.

[4] Liang, P., S. Petrov, D. Klein, et al. The infinite PCFG using hierarchical Dirichlet processes. In *Empirical Methods in Natural Language Processing*. 2007.

[5] Canini, K., L. Shi, T. Griffiths. Online inference of topics with latent Dirichlet allocation. In *Artificial Intelligence and Statistics*. 2009.

[6] Rodriguez, A. On-line learning for the infinite hidden Markov model. Tech. Rep. UCSC-SOE-10-27, Department of Applied Mathematics and Statistics, University of California at Santa Cruz, 2010.

[7] Hoffman, M., D. Blei, F. Bach. Online inference for latent Drichlet allocation. In *NIPS*. 2010.

[8] Jordan, M., Z. Ghahramani, T. Jaakkola, et al. Introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 1999.

[9] Robbins, H., S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):pp. 400–407, 1951.

[10] Amari, S. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[11] Sato, M. Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681, 2005.

[12] Attias, H. A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems 12*. 2000.

[13] Boyd-Graber, J., D. Blei. Syntactic topic models. In *Neural Information Processing Systems*. 2009.

[14] Fox, E., E. Sudderth, M. Jordan, et al. An HDP-HMM for systems with state persistence. In *International Conference on Machine Learning*. 2008.

[15] Sethuraman, J. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

[16] Ferguson, T. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1:209–230, 1973.

[17] Pitman, J. Poisson-Dirichlet and GEM invariant distributions for split-and-merge transformations of an interval partition. *Combinatorics, Probability, and Computing*, 11:501–514, 2002.

[18] Blei, D., M. Jordan. Variational methods for the Dirichlet process. In *21st International Conference on Machine Learning*. 2004.

[19] Kurihara, K., M. Welling, Y. Teh. Collapsed variational Dirichlet process mixture models. In *IJCAI*. 2007.

[20] Kurihara, K., M. Welling, N. Vlassis. Accelerated variational Dirichlet process mixtures. In B. Schölkopf, J. Platt, T. Hoffman, eds., *Advances in Neural Information Processing Systems 19*, pages 761–768. MIT Press, Cambridge, MA, 2007.

[21] Wang, C., D. Blei. Variational inference for the nested Chinese restaurant process. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, A. Culotta, eds., *Advances in Neural Information Processing Systems 22*, pages 1990–1998. 2009.

[22] Asuncion, A., M. Welling, P. Smyth, et al. On smoothing and inference for topic models. In *Uncertainty in Artificial Intelligence*. 2009.

[23] Blei, D., J. Lafferty. Dynamic topic models. In *International Conference on Machine Learning*, pages 113–120. ACM, New York, NY, USA, 2006.

[24] Wang, C., D. Blei, D. Heckerman. Continuous time dynamic topic models. In *Uncertainty in Artificial Intelligence (UAI)*. 2008.

[25] Griffiths, T., Z. Ghahramani. Infinite latent feature models and the Indian buffet process. In Y. Weiss, B. Schölkopf, J. Platt, eds., *Advances in Neural Information Processing Systems 18*, pages 475–482. MIT Press, Cambridge, MA, 2006.

[26] Teh, Y., D. Gorur, Z. Ghahramani. Stick-breaking construction for the Indian buffet process. In *11th Conference on Artificial Intelligence and Statistics*. 2007.

[27] Doshi-Velez, F., K. Miller, J. Van Gael, et al. Variational inference for the Indian buffet process. In *Proceedings of the Intl. Conf. on Artificial Intelligence and Statistics*, pages 137–144. 2009.