Online Latent Dirichlet Allocation with Infinite Vocabulary

Ke Zhai ZHAIKE@CS.UMD.EDU
Department of Computer Science, University of Maryland, College Park, MD USA

Jordan Boyd-Graber JBG@UMIACS.UMD.EDU
iSchool and UMIACS, University of Maryland, College Park, MD USA

Abstract

Topic models based on latent Dirichlet allocation (LDA) assume a predefined vocabulary. This is reasonable in batch settings but not reasonable for streaming and online settings. To address this lacuna, we extend LDA by drawing topics from a Dirichlet process whose base distribution is a distribution over all strings rather than from a finite Dirichlet. We develop inference using online variational inference and—to only consider a finite number of words for each topic—propose heuristics to dynamically order, expand, and contract the set of words we consider in our vocabulary. We show our model can successfully incorporate new words and that it performs better than topic models with finite vocabularies in evaluations of topic quality and classification performance.

1. Introduction

Latent Dirichlet allocation (LDA) is a popular probabilistic approach for exploring topics in collections of documents (Blei et al., 2003). Topic models offer a formalism for exposing a collection's themes and have been used to aid information retrieval (Wei & Croft, 2006), understand academic literature (Dietz et al., 2007; Gerrish & Blei, 2010), and discover political perspectives (Paul & Girju, 2010).

As hackneyed as the term "big data" has become, researchers and industry alike require algorithms that are scalable and efficient. Topic modeling is no different. A common approach to achieve scalability is to convert batch algorithms into streaming algorithms, which only make one pass over the data. In topic modeling, Hoffman et al. (2010) extended LDA to the online setting, and Wang et al. (2011) extended it to the hierarchical Dirichlet process.

However, all of these existing approaches for online inference in topic models make the same limiting assumption. The namesake topics, distributions over words that evince thematic coherence, are always modeled as a multinomial drawn from a finite Dirichlet distribution. This assumption means that the vocabulary is constant and precludes additional words being added over time.

Particularly for streaming algorithms, this is neither reasonable nor appealing. There are many reasons why the assumption of finiteness does not make sense: words can be invented ("crowdsourcing"), words can cross languages ("Gangnam"), or words common one context can become prominent in another context ("vuvuzelas" moving from music to sports in the 2010 World Cup). To be flexible, topic models must be able to capture the addition, invention, and increased prominence of new terms.

Allowing models to expand topics to include additional words requires changing the underlying statistical formalism. Instead of assuming that topics come from a finite Dirichlet distribution, we assume that it comes from a Dirichlet Process (Ferguson, 1973) with a base distribution over all possible words, of which there are an infinite number. Bayesian nonparametric tools like the Dirichlet process allow us to reason about infinite distributions. We review both topic models and Bayesian nonparametrics in Section 2. In Section 3, we present a model that uses Bayesian nonparametrics to discover distributions over words that are not limited to a fixed vocabulary.

In Section 4, we derive approximate posterior inference in our model. Since emerging vocabulary are most important in non-batch settings, in Section 5, we extend inference to streaming settings using online variational inference. We compare the coherence and effectiveness of our infinite vocabulary topic model against models with fixed vocabulary in Section 6.

Figure 1 shows how a topic evolves over time via online variational inference. The algorithm processes documents in groups that we call a *minibatch*; after each minibatch online variational inference updates the parameters of our model. While many of the details will be explained in Section 6.4, words that *were out of the vocabulary* can enter topics and eventually become *high probability words*. For example, "wolverin(e)" first appeared in minibatch 16, was added to

Proceedings of the 30^{th} *International Conference on Machine Learning*, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

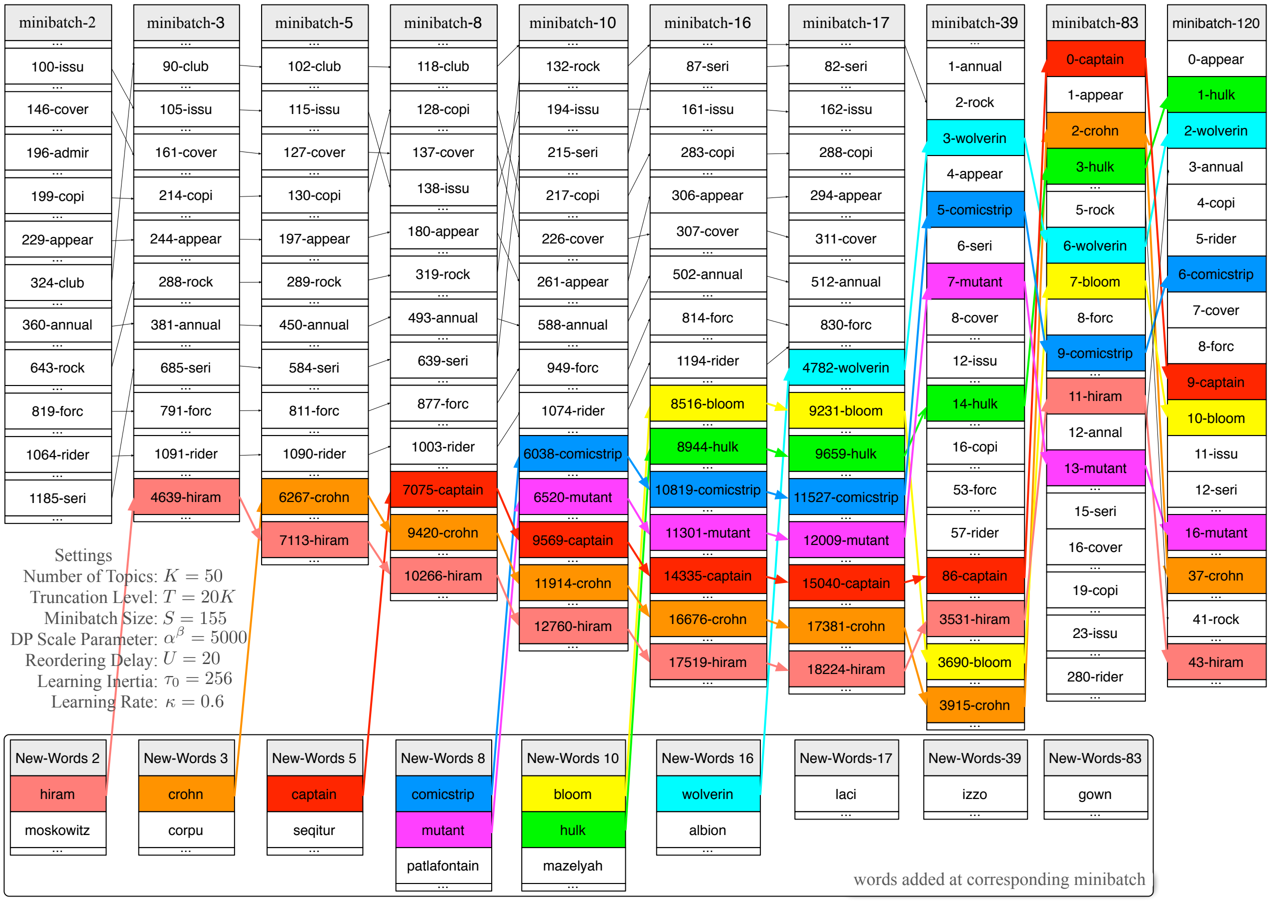


Figure 1. The evolution of a single topic about comic books from the *20 newsgroups* corpus. Each column shows the topic after processing a minibatch. Words within a topic are ordered based on the probability of the word in the topic (descending). The box below the topics shows new words incorporated into the vocabulary in a minibatch. The number in each cell represents the rank of that word in the topic after that minibatch. For example, "hulk" appeared and was introduced to the vocabulary in the minibatch 10. It was ranked at 9659 after minibatch 17 before becoming the 2-nd important word by the last minibatch. Words are colored to help show words' trajectories.

the vocabulary, and became the third important word in this topic by the final minibatch.

2. Background

Latent Dirichlet allocation (Blei et al., 2003) assumes a simple generative process. There are K topics, each drawn from a symmetric Dirichlet distribution, $\boldsymbol{\beta_k} \sim \text{Dir}(\eta), k = \{1, \dots, K\}$ that generate a corpus of observed words:

```
for each document $d$ in a corpus $D$ do
    Choose a distribution $\theta_d$ over topics from a Dirichlet distribution $\boldsymbol{\theta}_d \sim \text{Dir}(\alpha^\theta)$.
    for each of the $n = 1, \dots, N_d$ word indexes do
        Choose a topic $z_n$ from the document's distribution over topics $z_n \sim \text{Mult}(\boldsymbol{\theta}_d)$.
        Choose a word $w_n$ from the appropriate topic's distribution over words $p(w_n | \boldsymbol{\beta}_{z_n})$.
```

However, implicit in this model is that we will only observe a finite number of words, because the support of the Dirichlet distribution $\text{Dir}(\eta)$ is fixed. Moreover, we know *a priori* what words we can observe. This is patently false, as neologisms constantly appear (Algeo, 1980).

2.1. Bayesian Nonparametrics

Bayesian nonparametrics is an appealing solution to this problem. Such approaches allow us to model arbitrary distributions with an unbounded and possibly countably infinite support. While Bayesian nonparametrics is a broad field, we focus on the Dirichlet process (DP, Ferguson 1973).

The Dirichlet process is a two-parameter distribution with a scale parameter α^β and base distribution G_0. A draw G from $\text{DP}(\alpha^\beta, G_0)$ can be modeled as

$$b_1, \dots, b_i, \dots \sim \text{Beta}(1, \alpha^\beta), \quad \rho_1, \dots, \rho_i, \dots \sim G_0.$$

The individual draws from the Beta distribution are the foundation for the stick-breaking construction of the Dirichlet process (Sethuraman, 1994). Each break point b_i models how much of the remaining probability stick we break off.

These break points combine to form an infinite multinomial,

$$\beta_i \equiv b_i \prod_{j=1}^{i-1} (1 - b_j), \qquad G \equiv \sum_i \beta_i \delta_{\rho_i}, \qquad (1)$$

where the weights β_i give the probability of selecting any particular atom ρ_i drawn from the base distribution.

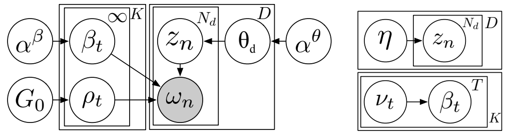
Specifically, the model we develop in Section 3 uses a base distribution G_0 over all possible words, and each topic is a draw from the Dirichlet process. This approach is inspired by unsupervised models that use Bayesian nonparametrics to induce part-of-speech classes.

2.2. Character Models within Latent Variable Models

One of the primary strengths of the probabilistic formalism is its ability to embed specialized models inside more general models. In particular, character-level n-gram models have been used extensively for part-of-speech (POS) induction, which is typically cast as an unsupervised hidden Markov model (HMM) over sequences of text (Goldwater & Griffiths, 2007). Such models use morphological regularity within part of speech classes (e.g., verbs in English often end with "ed") to learn a character n-gram model for specific parts of speech (Clark, 2003). This has been combined within the latent variable HMM model via a Chinese restaurant process (Blunsom & Cohn, 2011) so that the per-POS distribution over words is a non-parametric distribution that has a character n-gram model as its base distribution.

We also view latent clusters of words (topics) as being a nonparametric distribution that has a character n-gram model as its base distribution, but to better support streaming data sets, we use online variational inference, while previous approaches have used Markov chain Monte Carlo (MCMC, Neal 1993). Variational inference is easier to distribute (Zhai et al., 2012), amenable to online updates (Hoffman et al., 2010), and also can be combined with sparse local sampling schemes (Mimno et al., 2012).

Within the topic modeling community, there have been different approaches to how to deal with changing word use. Dynamic topic models (Blei & Lafferty, 2006; Wang et al., 2008), which discover an evolution of a topic over time by viewing word distributions as points in an n-dimensional space undergoing Brownian motion, allow the topic distribution over words to change. These models are compelling because they reveal that topics change radically over time; e.g., physics moving from æther to relativity to quantum mechanics to chromodynamics. However, they assume a **fixed vocabulary**. In datasets that have substantially changing vocabularies, mutable vocabularies can lead to more coherent topics, as we show in Section 6.



(a) LDA with infinite vocabulary (b) Variational

Figure 2. Plate representation for latent Dirichlet allocation with infinite vocabulary (left) and its variational distribution (right).

An elegant solution for large vocabularies is the "hashing trick" (Weinberger et al., 2009). Developed for supervised classification, it maps strings into a restricted set of integers via a hash function. These integers are then considered to be the vocabulary of the topic model. While it is an elegant solution, words are no longer identifiable. However, infinite vocabulary topic models retain identifiable vocabulary and also better model datasets, as we show in Section 6.3.

3. Infinite Vocabulary Topic Model

Our generative process is similar to that of LDA described in Section 2 except that the topics are no longer drawn from a finite Dirichlet. Instead, topics are drawn from a DP with a base distribution G_0 and scale parameter α^β, which assigns non-zero probability to every word:

```
for each topic $k$ do
    Draw words $\rho_{kt}, (t = \{1, 2, ...\})$ from $G_0$.
    Draw $b_{kt} \sim \mathrm{Beta}(1, \alpha^\beta), (t = \{1, 2, \dots\})$.
    Set the stick weights to be $\beta_{kt} = b_{kt} \prod_{s<t} (1 - b_{ks})$.
```

The rest of it is identical to LDA. The DP allows the topics to be distinct distributions, just as in the finite Dirichlet, but are now distributions with support over all possible words. The graphical model and its variational distribution are illustrated in Figure 2.

3.1. A Distribution over Words

An intuitive choice for G_0 is a conventional character language model. However, such a naïve approach is unrealistic and is biased to shorter words; preliminary experiments with this prior yielded poor results. Instead, we define G_0 as the following distribution over strings

```
Choose a length $l \sim \mathrm{Mult}(\boldsymbol{\lambda})$.
Generate character $c_i \sim p(c_i | \boldsymbol{c}_{1,\dots,i-1})$.
```

We assume the conditional distribution follows a n-order Markov model over characters, i.e. $p(c_i | \boldsymbol{c}_{1,\dots,i-1}) = p(c_i | \boldsymbol{c}_{i-n,\dots,i-1})$. This is similar to the classic n-gram language model, except that the length is first chosen from a multinomial distribution over all lengths. Estimating the conditional n-gram probabilities is a well-studied problem in natural language processing (Jelinek & Mercer, 1985).

The full expression for the probability of a word ρ consisting of the characters $c_1, c_2, \dots$ under G_0 is thus

$$G_0(\rho) \equiv p(l = |\rho| \,|\, \boldsymbol{\lambda}) \prod_{i=1}^{|\rho|} p(c_i | \boldsymbol{c}_{i-n,\dots,i-1})$$

where $|\rho|$ represents the length of the word. To avoid the length bias, we choose $\boldsymbol{\lambda}$ as the multinomial that minimizes the average discrepancy between word probabilities in a reference corpus $\mathcal{C}$ and the probability in our language model

$$\boldsymbol{\lambda} \equiv \arg\min_{\boldsymbol{\lambda}} \sum_{\rho} |p_{\mathcal{C}}(\rho) - p_{\mathsf{WM}}(\rho|\,\boldsymbol{\lambda})|^2, \text{s.t.} \sum_l \lambda_l = 1.$$

The n-gram statistics are estimated from an English dictionary which need not be very large, since it is a language model over characters, not words.

4. Variational Approximation

Inference in probabilistic inference uncovers the latent variables that best reconstruct observed data. The quality of this reconstruction is measured by log likelihood. For a corpus of D documents where the d-th document contains N_d words, the joint distribution of the observed data is

$$p(\boldsymbol{W}, \boldsymbol{\rho}, \boldsymbol{\beta}, \boldsymbol{\theta}, \boldsymbol{z}) = \prod_{k=1}^{K} \left[\prod_{t=1}^{\infty} p(\rho_{kt}|G_0) \cdot p(\beta_{kt}|\alpha^{\beta})\right] \left[\prod_{d=1}^{D} p(\boldsymbol{\theta}_d|\alpha^{\theta}) \prod_{n=1}^{N_d} p(z_{dn}|\boldsymbol{\theta}_d) p(\omega_{dn}|z_{dn}, \boldsymbol{\beta}_{z_{dn}})\right].$$

Directly optimizing the joint for latent variables $\boldsymbol{Z} \equiv \{$corpus-level stick proportions $\boldsymbol{\beta}$, document-level topic distributions $\boldsymbol{\theta}$ and word-level topic distribution $\boldsymbol{z}\}$ is intractable, so we use variational inference (Blei et al., 2003).

To use variational inference, we select a simpler family of distributions over the latent variables $\mathbf{Z}$. We call these distributions q. This family of distributions allows us to optimize a lower bound of the likelihood called the *evidence lower bound* (ELBO) $\mathcal{L}$,

$$\log p(\boldsymbol{W}) \geq \mathbb{E}_{q(\boldsymbol{Z})}\left[\log p(\boldsymbol{W}, \boldsymbol{Z})\right] - \mathbb{E}_{q(\boldsymbol{Z})}\left[q\right] = \mathcal{L}. \quad (2)$$

Maximizing $\mathcal{L}$ is equivalent to minimizing the *Kullback-Leibler* (KL) divergence between the true distribution and the variational distribution.

Unlike mean-field approaches (Blei et al., 2003), which assume q is a fully factorized distribution, we integrate out the word-level topic distribution vector $\boldsymbol{\theta}$: $q(\boldsymbol{z}_d\,|\,\eta)$ is a single distribution over K^{N_d} possible topic configurations rather than a product of N_d multinomial distributions over K topics. Combined with a beta distribution $q(b_{kt}|\nu_{kt}^1, \nu_{kt}^2)$ for stick break points, the variational distribution q is

$$q(\boldsymbol{Z}) \equiv q(\boldsymbol{\beta}, \boldsymbol{z}) = \prod_D q(\boldsymbol{z}_d\,|\,\eta) \prod_K q(\boldsymbol{b}_k\,|\,\boldsymbol{\nu}_k^1, \boldsymbol{\nu}_k^2). \quad (3)$$

However, we cannot explicitly represent a distribution over all possible strings, so we truncate our variational distribution for the stick breaking weights $q(\boldsymbol{b}\,|\,\boldsymbol{\nu})$ to a finite set.

4.1. Truncation Ordered Set

Variational methods typically cope with infinite dimensionality of nonparametric models by *truncating* the distribution to a finite subset of all possible atoms that nonparametric distributions consider (Blei & Jordan, 2005; Kurihara et al., 2006; Boyd-Graber & Blei, 2009). This is done by selecting a relatively large truncation index T_k, and then stipulating that the variational distribution uses the rest of the available stick at that index, i.e., $q(b_{T_k} = 1) \equiv 1$. As a consequence, β is zero in expectation under q beyond that index.

However, directly applying such a technique is not feasible here, as truncation is not just a search over dimensionality but also over atom strings and their ordering. This is often a problem in for nonparametric models, and the truncation that solves the problem matches the underlying probabilistic model: for mixture models, it is the number of components (Blei & Jordan, 2005); for hierarchical topic models, it is a tree (Wang & Blei, 2009); for natural language grammars, it is grammatons (Cohen et al., 2010). Similarly, our truncation is not just a fixed vocabulary size; it is a **truncation ordered set** (TOS). The ordering is important because the Dirichlet process is a size-biased distribution; words with lower indices are likely to have a higher probability than words with higher indices.

Each topic has a unique TOS $\mathcal{T}_k$ of limited size that maps every word type w to an integer t; thus $t = \mathcal{T}_k(w)$ is the index of the atom ρ_{kt} that corresponds to w. We defer how we choose this mapping until Section 4.3. More pressing is how we compute the two variational distributions of interest. For $q(z\,|\,\eta)$, we use local collapsed MCMC sampling (Mimno et al., 2012) and for $q(\boldsymbol{b}\,|\,\nu)$ we use stochastic variational inference (Hoffman et al., 2010). We describe both in turn.

4.2. Stochastic Inference

Recall that the variational distribution $q(\boldsymbol{z}_d\,|\,\eta)$ is a single distribution over the N_d vectors of length K. While this removes the tight coupling between θ and z that often complicates mean-field variational inference, it is no longer as simple to determine the variational distribution $q(\boldsymbol{z}_d\,|\,\eta)$ that optimizes Eqn. (2). However, Mimno et al. (2012) showed that Gibbs sampling instantiations of z_{dn}^* from the distribution conditioned on other topic assignments results in a sparse, efficient empirical estimate of the variation distribution. In our model, the conditional distribution of a topic assignment of a word with TOS index $t = \mathcal{T}_k(w_{dn})$ is

$$q(z_{dn} = k|\boldsymbol{z}_{-dn}, t = \mathcal{T}_k(w_{dn})) \quad (4)$$
$$\propto \left(\sum_{\substack{m=1 \\ m \neq n}}^{N_d} \mathbb{I}_{z_{dm}=k} + \alpha_k^{\theta}\right) \exp\left\{\mathbb{E}_{q(\boldsymbol{\nu})}\left[\log \beta_{kt}\right]\right\}.$$

We iteratively sample from this conditional distribution to obtain the empirical distribution $\phi_{dn} \equiv \hat{q}(z_{dn})$ for latent variable z_{dn}, which is fundamentally different from mean-field approach (Blei et al., 2003).

There are two cases to consider for computing Eqn. (4)—whether a word w_{dn} is in the TOS for topic k or not. First, we look up the word's index $t = \mathcal{T}_k(w_{dn})$. If this word is in the TOS, i.e., $t \leq T_k$, the expectations are straightforward (Blei & Jordan, 2005; Mimno et al., 2012)

$$q(z_{dn} = k) \propto \left(\textstyle\sum_{\substack{m=1 \\ m \neq n}}^{N_d} \phi_{dmk} + \alpha_k^\theta \right) \cdot \exp\{\Psi(\nu_{kt}^1) \quad (5)$$
$$+ \textstyle\sum_{s=1}^{s<t} \Psi(\nu_{ks}^2) - \sum_{s=1}^{s \leq t} \Psi(\nu_{ks}^1 + \nu_{ks}^2)\}$$

It is more complicated when a word is not in the TOS. Wang & Blei (2012) proposed a truncation-free stochastic variational approach for nonparametric models. It provides more flexible truncation schemes than split-merge techniques (Wang & Blei, 2009). The algorithm resembles a collapsed Gibbs sampler; it does not represent all mixture components explicitly. For our infinite vocabulary topic model, we do not want to ignore *out-of-vocabulary* (OOV) words; we assign these unseen words probability $1 - \sum_{t \leq T_k} \exp\left\{\mathbb{E}_{q(\nu)} \left[\log \beta_{kt}\right]\right\}$. The conditional distribution of an unseen word ($t > T_k$) then becomes

$$q(z_{dn} = k) \propto \left(\textstyle\sum_{\substack{m=1 \\ m \neq n}}^{N_d} \phi_{dmk} + \alpha_k^\theta \right) \quad (6)$$
$$\cdot \exp\{\textstyle\sum_{s=1}^{s \leq t} \left(\Psi(\nu_{ks}^2) - \Psi(\nu_{ks}^1 + \nu_{ks}^2) \right)\}.$$

This is different from finite vocabulary topic models, where vocabulary is chosen *a priori* and all OOV words are ignored. Our infinite vocabulary topic model assigns non-zero probability mass to the OOV words and allows new words to be included.

4.3. Refining the Truncation Ordered Set

Choosing the TOS is a combinatorial problem. While there exists approaches for variational inference over combinatorial structures (Bouchard-Cote & Jordan, 2010), we cannot employ them directly since the vocabulary is so large and the ELBO has no obvious recursive property we can exploit. Instead, we describe heuristics inspired by MCMC conditional equations, a common practice for updating truncations.

One component of a good TOS is that more frequent words should come first in the ordering. This is reasonable because the stick-breaking prior induces a size-biased ordering of the clusters. This has previously been used for truncation optimization for Dirichlet process mixtures and admixtures (Kurihara et al., 2007; Wang & Blei, 2012).

Another component of a good TOS is that words consistent with the underlying base distribution should be ranked higher than those not consistent with the base distribution. This intuition is also consistent with the conditional sampling equations for MCMC inference (Neal, 1993); the probability of creating a new table with dish ρ is proportional to $\alpha^\beta G_0(\rho)$ in the Chinese restaurant process.

Thus, to update the TOS, we define the ranking score of word t in topic k as

$$R(\rho_{kt}) = p(\rho_{kt}|G_0) \sum_{d=1}^{D} \sum_{n=1}^{N_d} \phi_{dnk} \delta_{\omega_{dn}=\rho_{kt}}, \quad (7)$$

sort all words by the scores within that topic, and then use those positions as the new TOS. In Section 5, we organize TOS based on ranking score in online settings.

5. Online Inference

Online variational inference seeks to optimize the ELBO according to Eqn. (2) by stochastic gradient optimization. Because gradients estimated from a single observation are exceedingly noisy, stochastic inference for topic models typically uses "minibatches" of S documents out of a total of D documents (Hoffman et al., 2010).

An approximation of the natural gradient of $\mathcal{L}$ with respect to ν can be obtained by multiplying the inverse of the Fisher information matrix and the first derivative (Sato, 2001)

$$\Delta\nu_{kt}^1 = 1 + \tfrac{D}{|\mathcal{S}|} \textstyle\sum_{d \in \mathcal{S}} \sum_{n=1}^{N_d} \phi_{dnk} \delta_{\omega_{dn}=\rho_{kt}} - \nu_{kt}^1 \quad (8)$$
$$\Delta\nu_{kt}^2 = \alpha^\beta + \tfrac{D}{|\mathcal{S}|} \textstyle\sum_{d \in S} \sum_{n=1}^{N_d} \phi_{dnk} \delta_{\omega_{dn}>\rho_{kt}} - \nu_{kt}^2,$$

which leads to an update of ν,

$$\nu_{kt}^1 = \nu_{kt}^1 + \epsilon \cdot \Delta\nu_{kt}^1, \qquad \nu_{kt}^2 = \nu_{kt}^2 + \epsilon \cdot \Delta\nu_{kt}^2 \quad (9)$$

where $\epsilon_i = (\tau_0 + i)^{-\kappa}$ defines the step size of the algorithm in minibatch i. The **learning rate** κ controls how quickly new parameter estimates replace the old; $\kappa \in (0.5, 1]$ is required to guarantee convergence. The **learning inertia** τ_0 prevents early minibatches from converging too quickly. We recover the batch setting if $\mathcal{S} = \mathcal{D}$ and $\kappa = 0$.

5.1. Updating the Truncation Ordered Set

A nonparametric streaming model should allow the vocabulary to dynamically expand as new words appear (e.g., introducing "vuvuzelas" for the 2010 World Cup), and contract as needed to best model the data (e.g., removing "vuvuzelas" after the craze passes). We describe three components of this process, expanding the truncation, refining the ordering of TOS, and contracting the vocabulary.

Determining the TOS Ordering This process depends on the ranking score of a word in topic k at minibatch i, $R_{i,k}(\rho)$. Ideally, we would want to compute R from all data. However, only single minibatch is accessible. Thus, we have a per-minibatch estimate of the rank

$$r_{i,k}(\rho) = p(\rho|G_0) \cdot \tfrac{D}{|\mathcal{S}_i|} \textstyle\sum_{d \in \mathcal{S}_i} \sum_{n=1}^{N_d} \phi_{dnk} \delta_{\omega_{dn}=\rho}$$

which we interpolate with our previous ranking

$$R_{ik}(\rho) = (1 - \epsilon) \cdot R_{i-1,k}(\rho) + \epsilon \cdot r_{ik}(\rho) \quad (10)$$

We recover the exact ranking score when $\mathcal{S} = \mathcal{D}$ and $\kappa = 0$ as in the batch setting.

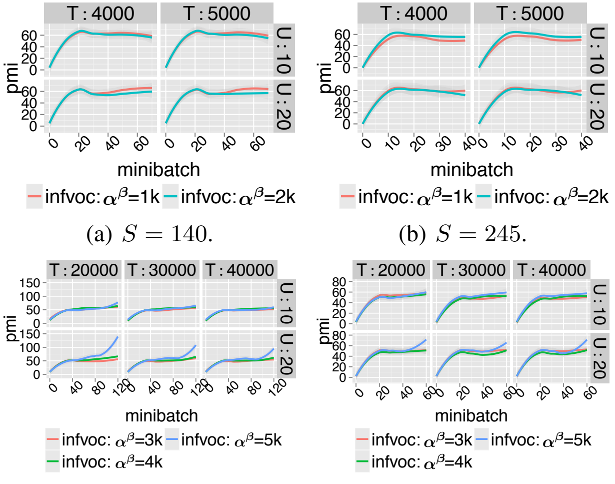
We introduce an additional algorithm parameter, the **reordering delay** U. We found that reordering after every minibatch ($U = 1$) was not effective; we explore the role of reordering delay in Section 6. After U minibatches have been observed, we reorder the TOS for each topic according to the words' ranking score R in Eqn. (10); $\mathcal{T}_k(w)$ becomes the rank position of w according to the latest R_{ik}.

Expanding the Vocabulary Each minibatch contains words we have not seen before. When we see them, we must determine their relative rank position in the TOS, their rank scores, and their associated variational parameters. The latter two issues are relevant for online inference because both are computed via interpolations from previous values in Eqn. (10) and (9). For an unseen word ω, previous values are undefined. Thus, we set $R_{i-1,k}$ for unobserved words to be 0, $\boldsymbol{\nu}$ to be 1, and $\mathcal{T}_k(\omega)$ is $T_k + 1$ (i.e., increase truncation and append to the TOS).

Contracting the Vocabulary To ensure tractability we must periodically prune the words in the TOS. When we reorder the TOS (after every U minibatches), we only keep the top T terms, where T is a user-defined integer. A word type ρ will be removed from $\mathcal{T}_k$ if its index $\mathcal{T}_k(\rho) > T$ and its previous information (e.g., rank and variational parameters) is discarded. In a later minibatch, if a previously discarded word reappears, it is treated as a new word.

6. Experimental Evaluation

In this section, we evaluate the performance of our infinite vocabulary topic model (*infvoc*) on two corpora: *de-news*[1] and *20 newsgroups*.[2] Both corpora were parsed by the same tokenizer and stemmer with a common English stopword list (Bird et al., 2009). First, we examine its sensitivity to both model parameters and online learning rates. Having chosen those parameters, we then compare our model with other topic models with fixed vocabularies.



(a) $S = 140$. (b) $S = 245$.

(c) $S = 155$. (d) $S = 310$.

Figure 3. PMI score on *de-news* (upper, $K = 10$) and *20 newsgroups* (lower, $K = 50$) dataset against different settings of DP scale parameter α^β, truncation level T and reordering delay U, under learning rate $\kappa = 0.8$ and learning inertia $\tau_0 = 64$. Our model seems more sensitive to α^β and less sensitive to T. In addition, a larger setting of U may be more desirable for larger corpora.

Evaluation Metric Typical evaluation of topic models is based on held-out likelihood or perplexity. However, creating a strictly fair comparison for our model against existing topic model algorithms is difficult, as traditional topic model algorithms must discard words that have not previously been observed. Moreover, held-out likelihood is a flawed proxy for how topic models are used in the real world (Chang et al., 2009). Instead, we use two evaluation metrics: topic coherence and classification accuracy.

Pointwise mutual information (PMI), which correlates with human perceptions of topic coherence, computes how well the words in a topic fit together. Following Newman et al. (2009), we extract document co-occurence statistics from Wikipedia, and score a topic's coherence by averaging the pairwise PMI (wrt Wikipedia co-occurence) of the topic's ten highest ranked words. The higher the average pairwise PMI, the more coherent the topic is.

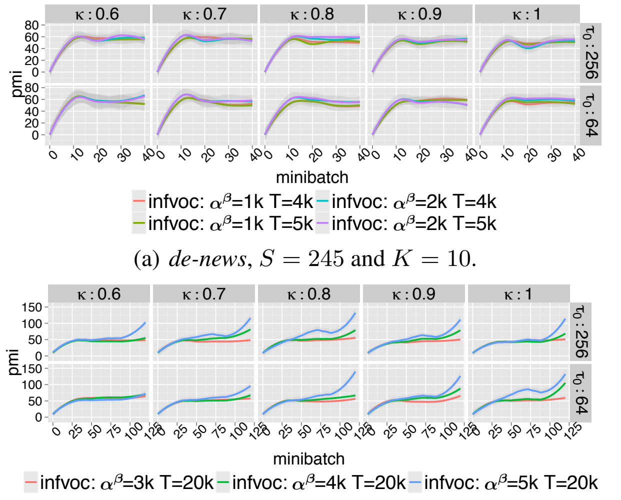
Classification accuracy is the accuracy of a classifier learned from the topic distribution of training documents applied to test documents (the topic model sees both sets). A higher accuracy means the unsupervised topic model better captures the underlying structure of the corpus. To better simulate real-world situations, 20-news's test/train split is by date (test documents appeared after training documents).

Comparisons We evaluate the performance of our model (*infvoc*) against three other models with fixed vocabularies: online variational Bayes LDA (*fixvoc-vb*, Hoffman et al. 2010), online hybrid LDA (*fixvoc-hybrid*, Mimno et al. 2012), and dynamic topic models (*dtm*, Blei & Lafferty 2006). Including dynamic topic models is not a fair comparison, as its inferences requires access to all of the documents in the dataset; unlike the other algorithms, it is not online.

Vocabulary For these fixed vocabulary models, we must decide on a vocabulary *a priori*. We consider three different

[1] A collection of daily news items between 1996 to 2000 in English. It contains 9,756 documents, 1,175,526 word tokens, and 20,000 distinct word types. Available at homepages.inf.ed.ac.uk/pkoehn/publications/de-news.

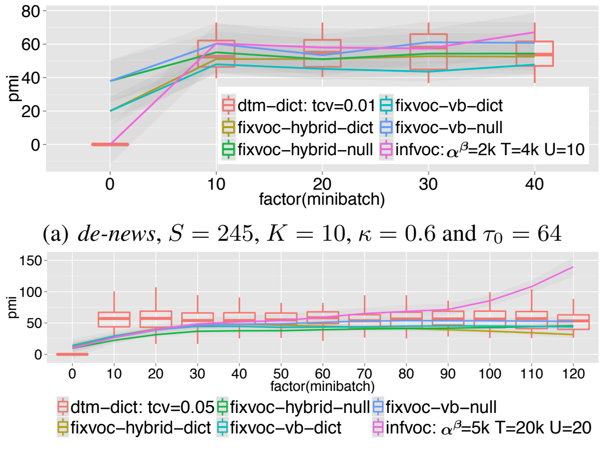
[2] A collection of discussions in 20 different newsgroups. It contains 18,846 documents and 100,000 distinct word types. It is sorted by date into roughly 60% training and 40% testing data. Available at qwone.com/~jason/20Newsgroups.



(a) *de-news*, $S = 245$ and $K = 10$.

(b) *20 newsgroups*, $S = 155$ and $K = 50$.

Figure 4. PMI score on two datasets with reordering delay $U = 20$ against different settings of decay factor κ and τ_0. A suitable choice of DP scale parameter α^β increases the performance significantly. Learning parameters κ and τ_0 jointly define the step decay. Larger step sizes promote better topic evolution.



(a) *de-news*, $S = 245$, $K = 10$, $\kappa = 0.6$ and $\tau_0 = 64$

(b) *20 newsgroups*, $S = 155$, $K = 50$, $\kappa = 0.8$ and $\tau_0 = 64$

Figure 5. PMI score on two datasets against different models. Our model *infvoc* yields a better PMI score against *fixvoc* and *dtm*; gains are more marked in later minibatches as more and more proper names have been added to the topics. Because *dtm* is not an online algorithm, we do not have detailed per-minibatch coherence statistics and thus show topic coherence as a box plot per epoch.

ways of constructing the vocabulary: use no outside information but instead use the first minibatch to define a vocabulary (*null*), or use a comprehensive dictionary[3] (*dict*). We use the same dictionary to train *infvoc*'s base distribution.

Experiment Configuration For all models, we use the same symmetric document Dirichlet prior with $\alpha^\theta = 1/K$, where K is the number of topics. Online models see exactly the same minibatches. For *dtm*, which is not an online algorithm but instead partitions its input into "epochs", we combine documents in ten consecutive minibatches into an epoch (longer epochs tended to have worse performance; this was the shortest epoch that had reasonable runtime).

For online hybrid approaches (*infvoc* and *fixvoc-hybrid*), we collect 10 samples empirically from the variational distribution in E-step with 5 burn-in sweeps. For *fixvoc-vb* approach, we run 50 iterations for local variational parameter updates.

6.1. Sensitivity to Parameters

Figure 3 shows how the PMI score is affected by the DP scale parameter α^β, the truncation level T, and the reordering delay U. The relatively high values of α^β may be surprising to readers used to seeing a DP that instantiates dozens of atoms, but when vocabularies are in tens of thousands, such scale parameters are necessary to support the long tail. Although we did not investigate such approaches, this suggests that more advanced nonparametric distributions (Teh, 2006) or explicitly optimizing α^β may be useful. Relatively large values of U suggest that accurate estimates of the rank order are important for maintaining coherent topics.

While *infvoc* is sensitive to parameters related to the vocabulary, once suitable values of those parameters are chosen, it is no more sensitive to learning-specific parameters than other online LDA algorithms (Figure 4), and values used for other online topic models also work well here.

6.2. Comparing Algorithms: Coherence

Now that we have some idea of how we should set parameters for *infvoc*, we compare it against other topic modeling techniques. We used grid search to select parameters for each of the models[4] and plotted the topic coherence averaged over all topics in Figure 5.

While *infvoc* initially holds its own against other models, it does better and better in later minibatches, since it has managed to gain a good estimate of the vocabulary and the topic distributions have stabilized. Most of the gains in topic coherence come from highly specific proper nouns which are missing from vocabularies of the fixed-vocabulary topic models. This advantage holds even against *dtm*, a batch learning algorithm.

6.3. Comparing Algorithms: Classification

For the classification comparison, we can consider additional topic models. This is because while we needed the

[3] sil.org/linguistics/wordlists/english/

[4] For the *de-news* dataset, we select (*20 newsgroups* parameters in parentheses) minibatch size $S \in \{140, 245\}$ ($S \in \{155, 310\}$), DP scale parameter $\alpha^\beta \in \{1k, 2k\}$ ($\alpha^\beta \in \{3k, 4k, 5k\}$), truncation size $T \in \{3k, 4k\}$ ($T \in \{20k, 30k, 40k\}$), reordering delay $U \in \{10, 20\}$ for *infvoc*; and topic chain variable tcv $\in \{0.001, 0.005, 0.01, 0.05\}$ for *dtm*.

model settings				accuracy %
$S = 155$	$\tau_0 = 64$ $\kappa = 0.6$	*infvoc*	$\alpha^\beta = 3k$ $T = 40k$ $U = 10$	52.683
		fixvoc	vb-dict	45.514
		fixvoc	vb-null	49.390
		fixvoc	hybrid-dict	46.720
		fixvoc	hybrid-null	50.474
		fixvoc	vb dict-hash	52.525
		fixvoc	vb full-hash $T = 30k$	51.653
		fixvoc	hybrid dict-hash	50.948
		fixvoc	hybrid full-hash $T = 30k$	50.948
	dtm-dict $tcv = 0.001$			**62.845**
$S = 310$	$\tau_0 = 64$ $\kappa = 0.6$	*infvoc*	$\alpha^\beta = 3k$ $T = 40k$ $U = 20$	52.317
		fixvoc	vb-dict	44.701
		fixvoc	vb-null	51.815
		fixvoc	hybrid-dict	46.368
		fixvoc	hybrid-null	50.569
		fixvoc	vb dict-hash	48.130
		fixvoc	vb full-hash $T = 30k$	47.276
		fixvoc	hybrid dict-hash	51.558
		fixvoc	hybrid full-hash $T = 30k$	43.008
	dtm-dict $tcv = 0.001$			**64.186**

Table 1. Classification accuracy based on 50 topic features extracted from *20 newsgroups* data. Our model (*infvoc*) out-performs algorithms with a fixed or hashed vocabulary but not *dtm*, a batch algorithm that has access to all documents.

most probable *strings* in a topic for the PMI calculation, the classification experiment only needs a topic-proportion vector for each document. Thus, we can include hashed vocabularies. We consider two strategies for using a hash function for previously unseen words. The first, which we call *dict-hashing*, uses a dictionary for the known words and hashes any other words into the same set of integers. The second, *full-hash*, used in the Vowpal Wabbit package,[5] hashes *all* words into a set of T integers.

We train 50 topics for all models on the entire dataset which is ordered by date, and collect the document level topic distribution for every article. We treat such statistics as features and train a SVM classifier on all training data using Weka (Hall et al., 2009) with default parameters to classify all testing documents into one of the 20 newsgroup labels. A higher accuracy means the model is better capturing the underlying content.

Our model *infvoc* captures better topic features than online LDA *fixvoc* (Table 1) under all settings.[6] This suggests that in settings where data are arriving in a streaming setting *infvoc* can better categorize documents. However, the batch algorithm *dtm*, which has access to the entire dataset performs better because it can use later documents to retrospectively improve its understanding of earlier documents.

6.4. Qualitative Example

Figure 1 shows the evolution of a topic in *20 newsgroups* about *comics* as new vocabulary words enter from new minibatches. While topics improve over time (e.g., relevant words like "seri(es)", "issu(e)", "forc(e)" are ranked higher), interesting words are being added throughout training and obtain a significant weights in later minibatches (e.g., "captain", "comicstrip", "mutant"). This is not the case for standard online LDA—these words are ignored and the model does not capture such information. In addition, only about 40% of the word types appeared in the SIL English dictionary. Even with a comprehensive English dictionary, online LDA could not capture all the word types in the corpus, especially named entities.

7. Conclusion and Future Work

We proposed an online topic model that, instead of assuming vocabulary is known *a priori*, adds and sheds words over time. While our model is better able to create coherent topics, it does not outperform dynamic topic models (Blei & Lafferty, 2006; Wang et al., 2008) that explicitly model how topics change. It would be interesting to allow such models to—in addition to modeling the *change* of topics—also change the underlying *dimensionality* of the vocabulary.

In addition to explicitly modeling the change of topics over time, it is also possible to model additional structure within topic. Rather than a fixed, immutable base distribution, modeling each topic with a hierarchical character n-gram model would capture regularities in the corpus that would, for example, allow certain topics to favor different orthographies (e.g., a technology topic might prefer words that start with "i"). While some topic models have attempted to capture orthography for multilingual applications (Boyd-Graber & Blei, 2009), our approach is more robust and incorporating the our approach with models of transliteration (Knight & Graehl, 1997) might allow concepts expressed in one language better capture concepts in another, further improving the ability of algorithms to capture the evolving themes and topics in large, streaming datasets.

Acknowledgments

The authors thank Chong Wang, Dave Blei, and Matt Hoffman for publishing code and graciously answering questions. We thank Jimmy Lin and the anonymous reviewers for helpful suggestions. This research was supported by NSF grant #1018625. Any opinions, findings, conclusions, or recommendations expressed are the authors' and do not necessarily reflect those of the sponsors.

[5] `hunch.net/~vw/`

[6] Parameters were chosen via cross-validation on a $30\% - 70\%$ dev-test split from the following parameter settings: DP scale parameter $\alpha \in \{2k, 3k, 4k\}$, reordering delay $U \in \{10, 20\}$ (for *infvoc* only); truncation level $T \in \{20k, 30k, 40k\}$ (for *infvoc* and *fixvoc full-hash* models); step decay factors $\tau_0 \in \{64, 256\}$ and $\kappa \in \{0.6, 0.7, 0.8, 0.9, 1.0\}$ (for all online models); and topic chain variable $tcv \in \{0.01, 0.05, 0.1, 0.5\}$ (for *dtm* only).

References

Algeo, John. Where do all the new words come from? *American Speech*, 55(4):264–277, 1980.

Bird, Steven, Klein, Ewan, and Loper, Edward. *Natural Language Processing with Python*. O'Reilly Media, 2009.

Blei, David M. and Jordan, Michael I. Variational inference for Dirichlet process mixtures. *Journal of Bayesian Analysis*, 1(1): 121–144, 2005.

Blei, David M. and Lafferty, John D. Dynamic topic models. In *Proceedings of the International Conference of Machine Learning*, 2006.

Blei, David M., Ng, Andrew, and Jordan, Michael. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

Blunsom, Phil and Cohn, Trevor. A hierarchical Pitman-Yor process HMM for unsupervised part of speech induction. In *Proceedings of the Association for Computational Linguistics*, 2011.

Bouchard-Cote, Alexandre and Jordan, Michael. Variational inference over combinatorial spaces. In *Proceedings of Advances in Neural Information Processing Systems*. 2010.

Boyd-Graber, Jordan and Blei, David M. Multilingual topic models for unaligned text. In *Proceedings of Uncertainty in Artificial Intelligence*, 2009.

Chang, Jonathan, Boyd-Graber, Jordan, and Blei, David M. Connections between the lines: Augmenting social networks with text. In *Knowledge Discovery and Data Mining*, 2009.

Clark, Alexander. Combining distributional and morphological information for part of speech induction. 2003.

Cohen, Shay B., Blei, David M., and Smith, Noah A. Variational inference for adaptor grammars. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

Dietz, Laura, Bickel, Steffen, and Scheffer, Tobias. Unsupervised prediction of citation influences. In *Proceedings of the International Conference of Machine Learning*, 2007.

Ferguson, Thomas S. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973.

Gerrish, Sean and Blei, David M. A language-based approach to measuring scholarly impact. In *Proceedings of the International Conference of Machine Learning*, 2010.

Goldwater, Sharon and Griffiths, Thomas L. A fully Bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of the Association for Computational Linguistics*, 2007.

Hall, Mark, Frank, Eibe, Holmes, Geoffrey, Pfahringer, Bernhard, Reutemann, Peter, and Witten, Ian H. The WEKA data mining software: An update. *SIGKDD Explorations*, 11, 2009.

Hoffman, Matthew, Blei, David M., and Bach, Francis. Online learning for latent Dirichlet allocation. In *NIPS*, 2010.

Jelinek, F. and Mercer, R. Probability distribution estimation from sparse data. *IBM Technical Disclosure Bulletin*, 28:2591–2594, 1985.

Knight, Kevin and Graehl, Jonathan. Machine transliteration. In *Proceedings of the Association for Computational Linguistics*, 1997.

Kurihara, Kenichi, Welling, Max, and Vlassis, Nikos. Accelerated variational Dirichlet process mixtures. In *Proceedings of Advances in Neural Information Processing Systems*, 2006.

Kurihara, Kenichi, Welling, Max, and Teh, Yee Whye. Collapsed variational Dirichlet process mixture models. In *International Joint Conference on Artificial Intelligence*. 2007.

Mimno, David, Hoffman, Matthew, and Blei, David. Sparse stochastic inference for latent Dirichlet allocation. In *Proceedings of the International Conference of Machine Learning*, 2012.

Neal, Radford M. Probabilistic inference using Markov chain Monte Carlo methods. Technical Report CRG-TR-93-1, University of Toronto, 1993.

Newman, David, Karimi, Sarvnaz, and Cavedon, Lawrence. External evaluation of topic models. In *Proceedings of the Aurstralasian Document Computing Symposium*, 2009.

Paul, Michael and Girju, Roxana. A two-dimensional topic-aspect model for discovering multi-faceted topics. In *Association for the Advancement of Artificial Intelligence*, 2010.

Sato, Masa-Aki. Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681, July 2001.

Sethuraman, Jayaram. A constructive definition of Dirichlet priors. *Statistica Sinica*, 4:639–650, 1994.

Teh, Yee Whye. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the Association for Computational Linguistics*, 2006.

Wang, Chong and Blei, David. Variational inference for the nested Chinese restaurant process. In *Proceedings of Advances in Neural Information Processing Systems*, 2009.

Wang, Chong and Blei, David M. Truncation-free online variational inference for bayesian nonparametric models. In *Proceedings of Advances in Neural Information Processing Systems*, 2012.

Wang, Chong, Blei, David M., and Heckerman, David. Continuous time dynamic topic models. In *Proceedings of Uncertainty in Artificial Intelligence*, 2008.

Wang, Chong, Paisley, John, and Blei, David. Online variational inference for the hierarchical Dirichlet process. In *Proceedings of Artificial Intelligence and Statistics*, 2011.

Wei, Xing and Croft, Bruce. LDA-based document models for ad-hoc retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2006.

Weinberger, K.Q., Dasgupta, A., Langford, J., Smola, A., and Attenberg, J. Feature hashing for large scale multitask learning. In *Proceedings of the International Conference of Machine Learning*, pp. 1113–1120. ACM, 2009.

Zhai, Ke, Boyd-Graber, Jordan, Asadi, Nima, and Alkhouja, Mohamad. Mr. LDA: A flexible large scale topic modeling package using variational inference in mapreduce. In *Proceedings of World Wide Web Conference*, 2012.