

# Bayesian Networks on Dirichlet Distributed Vectors

Wray Buntine and Lan Du  
 NICTA and Australian National University  
 {wray.buntine, lan.du}@nicta.com.au

Petteri Nurmi  
 University of Helsinki  
 ptnurmi@cs.helsinki.fi

## Abstract

Exact Bayesian network inference exists for Gaussian and multinomial distributions. For other kinds of distributions, approximations or restrictions on the kind of inference done are needed. In this paper we present generalized networks of Dirichlet distributions, and show how, using the two-parameter Poisson-Dirichlet distribution and Gibbs sampling, one can do approximate inference over them. This involves integrating out the probability vectors but leaving auxiliary discrete count vectors in their place. We illustrate the technique by extending standard topic models to “structured” documents, where the document structure is given by a Bayesian network of Dirichlets.

## 1 Introduction

The Gaussian and the multinomial distributions are the only ones allowing exact Bayesian inference over arbitrary Bayesian networks. Other distributions can be included as long as they stay at particular nodes and restrictions are placed on the direction of inference. For instance, mixing multinomial and Gaussian works as long as in all cases of inference the multinomials are strictly non-descendents of the Gaussian variables (Lauritzen, 1989). Extending inference to Monte Carlo or Gibbs sampling, and allowing general purpose samplers dramatically broadens the range of distributions one can allow (Thomas et al., 1992).

In this paper we show how to perform approximate inference over networks of probability vectors related via (approximate) Dirichlet distributions. So for instance, one could have a hidden Markov model where the hidden states are probabilities vectors  $\mathbf{p}_i$  related in a chain by:

$$\mathbf{p}_{i-1} \sim \text{Dirichlet}(\beta\mathbf{p}_i), \quad \mathbf{p}_i \sim \text{Dirichlet}(\beta\mathbf{p}_{i+1}).$$

Nested Dirichlet distributions like this would make an ideal tool for Bayesian modelling of

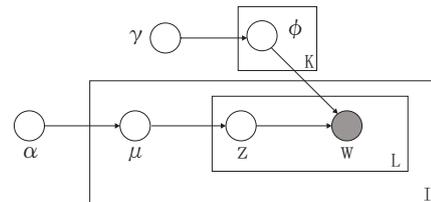


Figure 1: Standard Latent Dirichlet Allocation (LDA) Model.  $\alpha$  and  $\gamma$  are Dirichlet priors,  $\mu$  is a distribution over topics,  $\phi$  is a distribution over words, and  $z$  is a topic associated with a word  $w$ .  $I$  indicates the number of documents, and  $L$  denotes the number of words in document  $i$ .  $K$  is the number of topics.

hierarchical clustering, for instance, where previous methods have hierarchically partitioned the features (Hanson et al., 1991). They can also have a significant role in various places in extending standard topic models (Buntine and Jakulin, 2006; Blei et al., 2003), for instance, making documents, topics or components hierarchical. The standard model is in Figure 1. Both the links  $\alpha \rightarrow \mu$  and  $\gamma \rightarrow \phi$  are Dirichlet but  $\alpha$  and  $\gamma$  are really unknown so should be modelled by Dirichlets themselves, yielding a nested Dirichlet.

While MAP solutions for nested Dirichlets do exist, and an MCMC sampler on the real-valued probability vectors could also be applied, the

nested Dirichlets are often used as latent variables, so these solutions are inaccurate and inefficient respectively. This paper develops a more efficient and exact Gibbs sampler exists that integrates out the real-valued probability vectors by introducing small-valued integer vectors instead.

Bayesian hierarchical methods often use the *two-parameter Poisson-Dirichlet process* (PDP), also known as the Pitman-Yor process (Ishwaran and James, 2001). In Section 2, we discuss these models from our perspective and how they can be used in nested Dirichlet modelling. The basic theory comes from (Buntine and Hutter, 2010), some borrowed from (Teh, 2006a). Using these tools, in Section 3 networks of probability vectors distributed as Dirichlet and using PDPs, first presented in (Wood and Teh, 2009), are shown along with techniques for their statistical analysis. Their analysis is the main contribution of this paper. Some examples of these networks are embedded in new versions of topic models. These are presented, and some empirical results given in Section 4.

## 2 The Two-parameter Poisson-Dirichlet Process

The major tool used here is the *two-parameter Poisson-Dirichlet process* (PDP), also known as the Pitman-Yor process, which is an extension of the *Dirichlet process* (DP). The PDP is defined as:  $\nu \sim PDP(\mu, a, b)$ , where  $a$  is a discount parameter,  $b$  is a strength parameter, and  $\mu$  is a base distribution for  $\nu$ . These are used as tools for non-parametric and hierarchical Bayesian modelling.

The general theory of PDPs applies them to arbitrary measurable spaces (Ishwaran and James, 2001), for instance real valued spaces, but in many recent applications, such as language and vision applications, the domain is countable (*e.g.*, “English words”) and standard theory requires some modifications. In language domains, PDPs and DPs are proving useful for full probability modelling of various phenomena including n-gram modelling and smoothing (Teh, 2006b; Goldwater et al., 2006; Mochi-

hashi and Sumita, 2008), dependency models for grammar (Johnson et al., 2007; Wallach et al., 2008), and for data compression (Wood et al., 2009). The PDP-based n-gram models correspond well to versions of Kneser-Ney smoothing (Teh, 2006b), the state of the art method in applications. These models are intriguing from the probability perspective, as well as sometimes being competitive with performance based approaches. More generally, PDPs have been applied to clustering (Rasmussen, 2000) and image segmentation (Sudderth and Jordan, 2009).

For our purposes, knowledge of the details of the PDP and DP are not required. The two key results needed, however, follow. The Dirichlet Approximation Lemma is adapted from (Buntine and Hutter, 2010):

**Lemma 1** (Dirichlet Approximation Lemma). *Given a  $K$ -dimensional probability vector  $\mu$ , the following approximations on distributions hold (as  $a \rightarrow 0$ )*

$$\begin{aligned} PDP(0, b, \text{discrete}_K(\mu)) &\approx \text{Dirichlet}_K(b\mu), \\ PDP(a, 0, \text{discrete}_K(\mu)) &\approx \text{Dirichlet}_K(a\mu). \end{aligned}$$

*The first approximation is justified because the means and the first two central moments (orders 2 and 3) of the LHS and RHS distributions are equal. The second approximation is justified because the mean and first two central moments (orders 2 and 3) agree with error  $O(a^2)$ .*

Using this, we can see that one can replace the  $K$ -dimensional distribution  $\text{Dirichlet}_K(b\mu)$  by its approximation  $PDP(0, b, \text{discrete}_K(\mu))$ , and this greatly simplifies reasoning because PDPs turn out to be conjugate to multinomials. This result follows from the Marginalisation Lemma adapted from (Buntine and Hutter, 2010) and originally proven in a different format in a hierarchical context by Teh (Teh, 2006a).

**Lemma 2** (Marginalisation Lemma). *Given a probability vector  $\mu$  of dimension  $K$ , and the following set of priors and likelihoods for  $j = 1, \dots, J$*

$$\begin{aligned} \nu_j &\sim PDP(a, b, \text{discrete}_K(\mu)) \\ \mathbf{n}_j &\sim \text{multinomial}_K(\nu_j, N_j) \end{aligned}$$

where  $N_j = \sum_k n_{j,k}$ . Introduce auxiliary latent variables  $\mathbf{t}_j$  such that  $t_{j,k} \leq n_{j,k}$  and  $t_{j,k} = 0$  if and only if  $n_{j,k} = 0$ , then the following posterior distribution holds which marginalises out the  $\nu_{1:J}$  but introduces auxiliary variables  $\mathbf{t}_{1:J}$ :

$$p(\mathbf{n}_{1:J}, \mathbf{t}_{1:J} | a, b, \boldsymbol{\mu}) = \quad (1)$$

$$\prod_j C_{\mathbf{n}_j}^{N_j} \frac{(b|a)_{\sum_k t_{j,k}}}{(b)_{N_j}} \prod_{j,k} S_{t_{j,k},a}^{n_{j,k}} \prod_k \mu_k^{\sum_j t_{j,k}}.$$

The functions introduced in the lemma are as follows:  $C_{\mathbf{n}_j}^{N_j}$  is the multi-dimensional choose function of a multinomial;  $(x)_N$  is given by  $(x|1)_N$ ,  $(x|y)_N$  denotes the Pochhammer symbol with increment  $y$ , it is defined as

$$(x|y)_N = x(x+y)\dots(x+(N-1)y)$$

$$= \begin{cases} x^N & \text{if } y = 0, \\ y^N \times \frac{\Gamma(x/y+N)}{\Gamma(x/y)} & \text{if } y > 0, \end{cases}$$

where  $\Gamma(\cdot)$  denotes the standard gamma function; and  $S_{M,a}^N$  is a generalised Stirling number given by the linear recursion (Buntine and Hutten, 2010; Teh, 2006a)

$$S_{M,a}^{N+1} = S_{M-1,a}^N + (N-Ma)S_{M,a}^N$$

for  $M \leq N$ . It is 0 otherwise and  $S_{0,a}^N = \delta_{N,0}$ . These rapidly become very large so computation needs to be done in log space using a logarithmic addition.

In summary, we replace the Dirichlet by an approximation, a PDP, and then since this is conjugate to a multinomial (with introduction of suitable auxiliary variables), it allows ready processing in nested contexts. The nesting is proven in the next section. We can apply Lemma 2 without having any clear interpretation of the auxiliary variables  $t_{j,k}$ , which would require a more extensive presentation of PDPs<sup>1</sup>.

### 3 Networks of Dirichlet Distributed Vectors

Assume a directed graph  $G$  composed of nodes indexed by integers  $1, \dots, J$  with probability vectors  $\nu_j$  for  $j = 1, \dots, J$  and directed edges  $(i, j)$

<sup>1</sup>In the Chinese Restaurant Process of the sampling of  $\nu_j$ ,  $t_{j,k}$  represents the number of tables that have the same dish  $k$ . For non-atomic distributions  $t_{j,k}$  would almost surely be 1, so no sampling required.

for  $i \in E_j$  where  $E_j$  is the set of parents of the node  $j$ . Parameters in the model are as follows:

$\nu_j$ : the vector of probabilities at the  $j$ -th node.

$\rho_j$ : the vector of mixing probabilities at each node, gives how parent probability vectors are mixed. Only used when  $|E_j| > 1$ .

Hyperparameters in the model are:

$a, b$ : parameters for the PDP.

$\alpha_j$ : Dirichlet prior parameters for root node  $j$  which has no parents.

$\gamma_j$ : Dirichlet prior parameters for mixing probabilities  $\rho_j$  when  $|E_j| > 1$ .

Denote this set of hyperparameters as  $\mathcal{H}$ .

We consider models of the form:

$$\nu_j \sim \text{Dirichlet}_K(\alpha_j) \quad \text{for } E_j = \emptyset$$

$$\nu_j \sim \text{PDP} \left( a, b, \text{discrete}_K \left( \sum_{i \in E_j} \rho_{i,j} \nu_i \right) \right)$$

$$\quad \text{for } E_j \neq \emptyset.$$

Note, when  $|E_j| = 1$ , then the single hyperparameter  $\rho_{i,j}$  for  $i \in E_j$  is equal to 1, so the sum degenerates to  $\nu_i$ . The mixing parameters  $\rho_j$  can be modelled as  $\rho_j \sim \text{Dirichlet}_{E_j}(\gamma_j)$ , only needed when  $|E_j| > 1$ . Also, data is in the form  $\mathbf{n}_j \sim \text{multinomial}_K(\nu_j, N_j)$ .

These models may be embedded in larger networks, for instance they may be the ‘‘topic structure’’ for a topic model.

#### 3.1 Marginalising Parameters

Applying the Marginalisation Lemma at any leaf node  $j$  marginalises out the parameter  $\nu_j$  and introduces auxiliary variables  $\mathbf{t}_j$ :

$$C_{\mathbf{n}_j}^{N_j} \frac{(b|a)_{\sum_k t_{j,k}}}{(b)_{N_j}} \prod_k S_{t_{j,k},a}^{n_{j,k}} \prod_k \left( \sum_{i \in E_j} \rho_{i,j} \nu_{i,k} \right)^{t_{j,k}}.$$

So expand the sum using the multinomial identity. This implies decomposing  $t_{j,k}$  into parts coming from each of its parents, and call this

total now  $t_{j,k}^p$ , and introduce a complementary sum from their children, called  $t_{j,k}^c$

$$t_{j,k}^p = \sum_{i \in E_j} s_{i,j,k}, \quad t_{j,k}^c = \sum_{l: j \in E_l} s_{j,l,k},$$

and let  $\mathbf{s}_{j,k} = (s_{i,j,k} : i \in E_j)$ . Then integrating out  $\rho_j$ , one gets

$$C_{\mathbf{n}_j}^{N_j} \frac{(b|a)_{\sum_k t_{j,k}^p} \prod_{i \in E_j} \Gamma(\gamma_{i,j} + \sum_k s_{i,j,k})}{(b)_{N_j} \Gamma\left(\sum_i \gamma_{i,j} + \sum_k t_{j,k}^p\right)} \prod_k \left( S_{t_{j,k}^p, a}^{n_{j,k}} C_{\mathbf{s}_{j,k}}^{t_{j,k}^p} \prod_{i \in E_j} \nu_{i,k}^{s_{i,j,k}} \right),$$

and to this we must add the constraints for the given  $j$  and all  $k$

$$t_{j,k}^p \leq n_{j,k}, \quad t_{j,k}^p = 0 \text{ if and only if } n_{j,k} = 0.$$

Due to the terms  $\nu_{i,k}^{s_{i,j,k}}$  occurring in this, the counts  $s_{i,j,k}$  can be added to the data for the node for  $\nu_i$ ,  $n_{i,k}$ , and the procedure applied recursively. One has therefore proven the nested version of the Marginalisation Lemma,

**Lemma 3** (Marginalising a Network of Dirichlets). *Given the network of Dirichlets described in this section, introduce counts  $s_{i,j,k} \geq 0$  for  $i \in E_j$  and all  $k$  when  $E_j \neq \emptyset$ . These have parent and child totals  $t_{j,k}^p$  and  $t_{j,k}^c$  as above. These must satisfy constraints at each node  $j$  on the  $k$ -th value of*

$$t_{j,k}^p \leq n_{j,k} + t_{j,k}^c, \quad (2)$$

$$t_{j,k}^p = 0 \quad \text{iff} \quad n_{j,k} + t_{j,k}^c = 0. \quad (3)$$

Then marginalising out all parameters  $\nu_j$  and  $\rho_j$  yields the posterior ( $\mathcal{H}$  is hyperparameters)

$$p(\mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K} | \mathcal{H}) = \prod_{j: E_j \neq \emptyset} \frac{\prod_k \Gamma(\alpha_{j,k} + n_{j,k} + t_{j,k}^c)}{\Gamma\left(\sum_k \alpha_{j,k} + \sum_k n_{j,k} + \sum_k t_{j,k}^c\right)} \prod_{j: E_j \neq \emptyset} \frac{(b|a)_{\sum_k t_{j,k}^p} \prod_{i \in E_j} \Gamma(\gamma_{i,j} + \sum_k s_{i,j,k})}{(b)_{N_j + \sum_k t_{j,k}^c} \Gamma\left(\sum_i \gamma_{i,j} + \sum_k t_{j,k}^p\right)} \prod_{j: E_j \neq \emptyset} C_{\mathbf{n}_j}^{N_j} \prod_k \left( S_{t_{j,k}^p, a}^{n_{j,k} + t_{j,k}^c} C_{\mathbf{s}_{j,k}}^{t_{j,k}^p} \right), \quad (4)$$

The key challenge in working with these models is now handling the auxiliary variables  $\mathbf{s}_{1:J,1:K}$ . When Gibbs sampling is done, for instance, the constraints need to be maintained, and in our experience this turns out to be the major complexity.

### 3.2 Gibbs Sampling

We consider a single discrete item related to the  $j$ -th node, so its distribution is over  $\{1, \dots, K\}$  and has the probability vector  $\nu_j$ . The previous theory dealt with multinomials, so  $\mathbf{n}_j \sim \text{multinomial}_K(\nu_j, N_j)$ , however, in practice, we may also consider  $N_j$  discrete variables with distribution  $\text{discrete}_K(\nu_j)$ . Their sufficient statistics are also  $\mathbf{n}_j$ , and the difference in the posterior is that the choose term  $C_{\mathbf{n}_j}^{N_j}$  is removed.

In Gibbs sampling, suppose we are sampling the probability of this item related the  $j$ -th node. Then  $n_{j,k}$  will be decreased by 1 for some  $k$ , and increased by one for another. Given the form of the posterior in Equation (4), it is easy to consider the change in the posterior when one  $n_{j,k}$  is increased, lets denote this as

$$p(\text{increment } n_{j,k} | \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K}, \mathcal{H}) = \frac{p(\text{increment } n_{j,k}, \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K} | \mathcal{H})}{p(\mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K} | \mathcal{H})}$$

Many of the terms simplify due to ratios of Gamma functions. However, the problem arises that either increasing or decreasing  $n_{j,k}$  may violate the constraints for node  $j$  when  $E_j \neq \emptyset$ .

#### 3.2.1 Increasing a count

If  $n_{j,k}$  is increased by one, then Constraint (3) may be violated for node  $j$ . To fix this, we need to increase one of the  $s_{i,j,k}$  for  $i \in E_j$ . If  $|E_j| > 1$  then we have a choice and sampling needs to be done. Once the  $i$  is chosen, then  $s_{i,j,k}$  is increased by one, and subsequently Constraint (3) may now be violated for node  $i$  so the process iterates up the network. When considering this, we need to consider the set of ancestors of  $j$  reachable along paths where every element  $j'$  has equality for the Constraint (3) on the  $k$ -th value.

Suppose one is incrementing  $n_{j,k}$  and one's choice is to increment the set of counts  $s_{i_2,i_1,k}, \dots, s_{i_u,i_{u-1},k}$  where for convenience  $i_1 = j$ . Along this path, the equality of Constraint (3) must hold for  $i_1 = j, i_2, \dots, i_{u-1}$ . Then the probability of incrementing  $n_{j,k}$  using this path to balance equality constraints, denoted

$$p(\text{incr } n_{i_1,k}, s_{i_2,i_1,k}, \dots, s_{i_u,i_{u-1},k} \mid \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K}, \mathcal{H})$$

is given by

$$\begin{aligned} & \left( \frac{\alpha_{i_u,k} + n_{i_u,k} + t_{i_u,k}^c}{\sum_k \alpha_{i_u,k} + \sum_k n_{i_u,k} + \sum_k t_{i_u,k}^c} \right)^{\delta_{E_{i_u} = \emptyset}} \\ & \left( \frac{1}{b + N_{i_u} + \sum_k t_{i_u,k}^c} \frac{S_{t_{i_u,k}^p, a}^{n_{i_u,k} + t_{i_u,k}^c + 1}}{S_{t_{i_u,k}^p, a}^{n_{i_u,k} + t_{i_u,k}^c}} \right)^{\delta_{E_{i_u} \neq \emptyset}} \\ & \prod_{n=1}^{u-1} \frac{b + a \sum_k t_{i_n,k}^p}{b + N_u + \sum_k t_{i_n,k}^c} \frac{\gamma_{i_{n+1},i_n} + \sum_k s_{i_{n+1},i_n,k}}{\sum_l \gamma_{l,i_n} + \sum_k t_{i_n,k}^p}. \end{aligned}$$

We need to sum this over all possible paths  $i_1, i_2, \dots, i_{u-1}$  starting at  $i_1 = j$  in order to compute  $p(\text{incr } n_{j,k} \mid \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K}, \mathcal{H})$ . The following recursive computation does this summation for a given  $j, k$ .  $Z_i^+$  is evaluated recursively when  $E_i \neq \emptyset$  and Constraint (3) for node  $i$  at value  $k$  has equality. The recursive computation for  $Z_i^+$  is

$$\sum_{l \in E_i} Z_l^+ \frac{b + a \sum_k t_{i,k}^p}{b + N_i + \sum_k t_{i,k}^c} \frac{\gamma_{l,i} + \sum_k s_{l,i,k}}{\sum_n \gamma_{n,i} + \sum_k t_{i,k}^p}.$$

Otherwise,  $Z_i^+$  is evaluated as

$$\begin{aligned} & \frac{1}{b + N_i + \sum_k t_{i,k}^c} \frac{S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c + 1}}{S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c}} \quad \text{when } E_i \neq \emptyset, \\ & \frac{\alpha_{i,k} + n_{i,k} + t_{i,k}^c}{\sum_k \alpha_{i,k} + \sum_k n_{i,k} + \sum_k t_{i,k}^c} \quad \text{when } E_i = \emptyset. \end{aligned}$$

We evaluate  $p(\text{incr } n_{j,k} \mid \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K}, \mathcal{H}) = Z_j^+$ , and this gives the proportionality for sampling which  $k$  to choose when incrementing a count in  $\mathbf{n}_j$ . Once  $k$  is sampled, then a path  $i_1, i_2, \dots, i_{u-1}$  in the constraint set should be sampled. This can be done using a similar function to the one just covered.

### 3.2.2 Decreasing a count

If  $n_{j,k}$  is decreased by one, then Constraint (2) may be violated if the equality holds initially. This process is similar to the previous, except that now the  $s_{i,j,k}$  are decreased and one needs to consider the set of ancestors of  $j$  reachable along paths where every element  $j'$  has equality for the Constraint (2) on the  $k$ -th value. Using a similar argument to previous, one gets a related recursive computation.

$Z_i^-$  is evaluated recursively when  $E_i \neq \emptyset$  and Constraint (2) for node  $i$  at value  $k$  has equality. The recursive computation for  $Z_i^-$  is

$$\begin{aligned} & \sum_{l \in E_i} Z_l^- \frac{b + N_i + \sum_k t_{i,k}^c - 1}{b + a \sum_k t_{i,k}^p - 1} \frac{S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c - 1}}{S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c}} \\ & \frac{\sum_n \gamma_{n,i} + \sum_k t_{i,k}^p - 1}{\gamma_{l,i} + \sum_k s_{l,i,k} - 1} \frac{s_{l,i,k}}{t_{i,k}^p}. \end{aligned}$$

Otherwise,  $Z_i^-$  is evaluated as

$$\begin{aligned} & \frac{(b + N_i + \sum_k t_{i,k}^c - 1) S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c - 1}}{S_{t_{i,k}^p, a}^{n_{i,k} + t_{i,k}^c}} \quad \text{when } E_i \neq \emptyset, \\ & \frac{\sum_k \alpha_{i,k} + \sum_k n_{i,k} + \sum_k t_{i,k}^c - 1}{\alpha_{i,k} + n_{i,k} + t_{i,k}^c - 1} \quad \text{when } E_i = \emptyset. \end{aligned}$$

As before,  $p(\text{decr } n_{j,k} \mid \mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K}, \mathcal{H}) = Z_j^-$  and this gives the proportionality for sampling which  $k$  to choose when wanting to decrement a count in  $\mathbf{n}_j$ . The path to choose when decrementing is chosen similarly.

### 3.2.3 Sampling Auxiliary Counts

The auxiliary counts  $\mathbf{s}_{1:J,1:K}$  also need to be sampled. For correct Gibbs sampling, the space of moves must allow complete access to the state space of values  $\mathbf{s}_{1:J,1:K}$  legal for a given  $\mathbf{n}_{1:J}$ . We introduce two move operators for sampling.

The first operator changes a single  $s_{i,j,k}$  and possibly its ancestor counts. If Constraint (3) for node  $j$  at value  $k$  has equality, then  $s_{i,j,k}$  must be zero and will not change. Allow sampling at node  $j$  for value  $k$  only when  $n_{j,k} + t_{j,k}^c > 0$ . By Constraint (2) for node  $j$  at value  $k$ ,  $s_{i,j,k}$  will be sampled keeping  $s_{i,j,k} \leq n_{j,k} + t_{j,k}^c - \sum_{l \in E_j - \{i\}} s_{l,j,k}$  and keeping  $t_{j,k}^p > 0$ . Decreasing  $s_{i,j,k}$  according to these constraints becomes

the same task as given in Section 3.2.2. Increasing  $s_{i,j,k}$  within the constraints has no flow on affects so is the same as the standard incrementing formula in Section 3.2.1 where  $u = 1$ .

The second operator moves a count from one  $s_{i,j,k}$  to another  $s_{i',j,k}$ . This can be put together by first doing a decrease as per Section 3.2.2 followed by an increase as per Section 3.2.1. The constraints at node  $j$  at value  $k$  will be unaffected by the combined move so no descendent auxiliary counts will change. A recursive argument shows these two operators make Gibbs sampling correct for the space of auxiliary counts  $\mathbf{s}_{1:J,1:K}$ .

### 3.3 Sampling the Variance Parameter $b$

In using these models for topic modelling, we have found performance is quite sensitive to the parameter  $b$  which controls the variance. For instance, for the distribution  $\text{PDP}(a, b, \text{discrete}_K(\boldsymbol{\mu}))$  the hyperparameter  $b$  can thus roughly be thought of as the prior data count since variance is  $O(1/(b+1))$  (Buntine and Hutter, 2010).

We perform Gibbs sampling over  $b$  using auxiliary variables. First, consider the case where  $a = 0$ , discussed in (Teh et al., 2006). Consider the posterior for  $b$ ,  $p(\mathbf{n}_{1:J}, \mathbf{s}_{1:J,1:K} | \mathcal{H}, a = 0)$ , proportional to

$$\prod_{j:E_j \neq \emptyset} \frac{b^{\sum_{i,k} s_{i,j,k}} \Gamma(b)}{\Gamma(b + N_j + \sum_{i,k} s_{j,i,k})}$$

Introduce  $q_j \sim \text{Beta}(b, N_j + \sum_{i,k} s_{j,i,k})$  as auxiliary variables. Then the joint posterior distribution for  $q_j$  and  $b$  is proportional to

$$b^{\sum_{i,j,k} s_{i,j,k}} \prod_{j:E_j \neq \emptyset} q_j^{b-1} (1 - q_j)^{N_j + \sum_{i,k} s_{j,i,k} - 1}.$$

The auxiliary sampling scheme then becomes:

$$q_j \sim \text{Beta}\left(b, N_j + \sum_{i,k} s_{j,i,k}\right) \quad \text{for each } j,$$

$$b \sim \text{Gamma}\left(\sum_{i,j,k} s_{i,j,k} + 1, \sum_j \log 1/q_j\right).$$

For the case when  $a > 0$  things become a bit more elaborate. Now the posterior is proportional to

$$\prod_{j:E_j \neq \emptyset} \frac{\Gamma(b) \Gamma(b/a + \sum_{i,k} s_{i,j,k})}{\Gamma(b + N_j + \sum_{i,k} s_{j,i,k}) \Gamma(b/a)}$$

Introducing the same auxiliary variables as before yields a joint posterior distribution for  $q_j$  and  $b$  that is easily shown to be log concave, so the second step in the previous case ( $a = 0$ ) is now replaced by an adaptive regression sampling step in  $b$  (Gilks and Wild, 1992).

## 4 Experiments

We extended standard LDA, shown in Figure 1 in two directions, which have been more fully developed and experimented with elsewhere (Du et al., 2010a; Du et al., 2010b). Here we cover basic results to demonstrate the effectiveness of the theory developed. The first model, the Segmented Topic Model (STM) (Du et al., 2010a) is shown in Figure 2, and allows a document to be broken into  $J$  segments. Each segment has its own topic proportions  $\boldsymbol{\nu}_{i,j}$  which are related by our scheme using a PDP to the general proportions for the whole documents proportions  $\boldsymbol{\mu}_i$ . The second model, called the Sequential LDA (SeqLDA) (Du et al., 2010b), models the progressive topical dependencies among segments with a hidden Markov model of topic proportions  $\boldsymbol{\nu}_{i,1}, \dots, \boldsymbol{\nu}_{i,J}$ , shown in Figure 3.

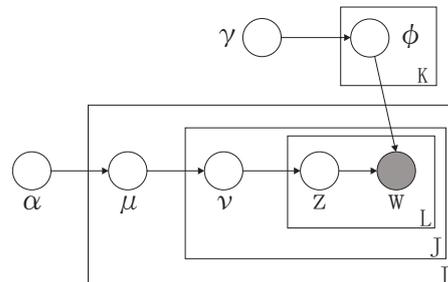


Figure 2: Segmented Topic Model.

A patent dataset was randomly selected from 5000 U.S. patents<sup>2</sup> granted between Jan. and

<sup>2</sup>All patents are from Cambia, <http://www.cambia.org/daisy/cambia/home.html>

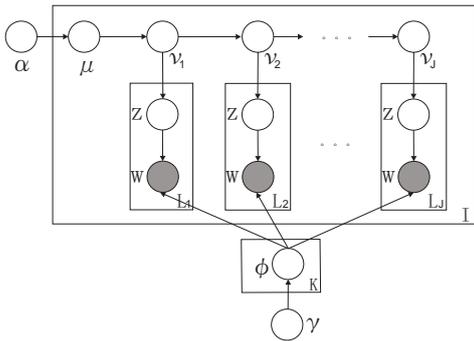


Figure 3: Sequential LDA.

Table 1: Perplexity on datasets.

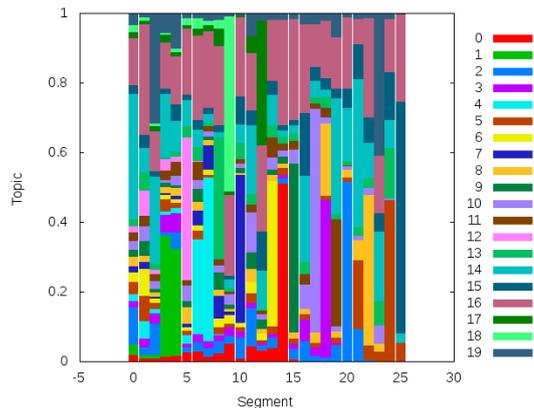
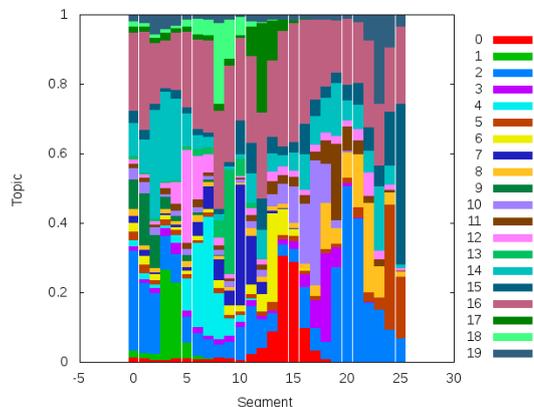
	K	STM	LDA_D	LDA_S
G06-1000	100	1270	1712	1508
	150	1178	1595	1393
NIPS	100	1632	1991	2182
	150	1516	1881	2186

Mar. 2009 under the class “computing; calculating; counting” with international patent classification (IPC) code G06. Patents in this dataset called G06-1000 are split into paragraphs according to the original structure. All stop-words, extremely common words (*e.g.* top 40), and less common words (*i.e.* words appear less than 5 documents) have been removed. This leads to a vocabulary size of 10385 unique words segmented as 60,564 paragraphs, and 2,513,087 words. We also processed the NIPS dataset<sup>3</sup> removing bibliography material (everything after “References”) and header material (everything before “Abstract”) yielding 1,629 documents segmented as 174,474 sentences with 1,773,365 words total.

To evaluate the generalization capability of these models to unseen data, we compute perplexity, a standard measure in language modelling. The perplexity of a collection  $\mathcal{D}$  of  $I$  documents is defined as  $\exp \left\{ - \frac{\sum_{i=1}^I \ln p(\mathbf{w}_i)}{\sum_{i=1}^I N_i} \right\}$  where  $\mathbf{w}_i$  indicates all words in document  $i$ , and  $N_i$  indicates the total number of words in  $i$ . A lower perplexity over unseen documents means better generalization capability. In our experiments, it is computed based on the held-out method introduced in (Rosen-Zvi et al., 2004) with 80% for training and 20% for testing.

<sup>3</sup>It is available at <http://nips.djvuzone.org/txt.html>

Perplexity results appear in the table where LDA has been run twice, once on the full documents (LDA\_D) and once on the segments within documents (LDA\_S). Clearly, the STM model works well. The SeqLDA model was also run and not only gives better results than the LDA, but also reveals a strong sequential structure from segment to segment. To illustrate the sequential behaviour of topics for SeqLDA, Figures 4 and 5 compare topic proportions for aligned topics for each segment (*i.e.* chapter) of the book *The Prince* by Machiavelli. SeqLDA is clearly seen to have topics flow better from one chapter to another than LDA. Refer to (Du et al., 2010a; Du et al., 2010b) for more detailed experimental results.

Figure 4: LDA topics in *The Prince*.Figure 5: SeqLDA topics in *The Prince*.

## 5 Conclusion

We have shown how to perform inference on Bayesian networks of Dirichlet distributed probability vectors using Gibbs sampling over discrete auxiliary variables. Experiments demonstrate the general approach.

## Acknowledgments

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program. This work was supported in part by the IST Programme of the European Community, under the PASCAL2 Network of Excellence, IST-2007-216886. This publication only reflects the authors' views.

## References

- D.M. Blei, A.Y. Ng, and M.I. Jordan. 2003. Latent Dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.
- W. Buntine and M. Hutter. 2010. A Bayesian interpretation of the Poisson-Dirichlet process. *Available at: <http://arxiv.org/abs/1007.0296v1>*.
- W.L. Buntine and A. Jakulin. 2006. Discrete components analysis. In *Subspace, Latent Structure and Feature Selection Techniques*. Springer-Verlag.
- L. Du, W. Buntine, and H. Jin. 2010a. A segmented topic model based on the two-parameter Poisson-Dirichlet process. *Machine Learning (in press)*.
- L. Du, W. Buntine, and H. Jin. 2010b. Sequential latent Dirichlet allocation: Discover underlying topic structures within a document. Technical report, NICTA. In submission.
- W.R. Gilks and P. Wild. 1992. Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, 41:337–348.
- S. Goldwater, T. Griffiths, and M. Johnson. 2006. Interpolating between types and tokens by estimating power-law generators. In *NIPS 18*, pages 459–466. MIT Press.
- R. Hanson, J. Stutz, and P. Cheeseman. 1991. Bayesian classification with correlation and inheritance. In *Proc. of the 12th IJCAI*, pages 692–698.
- H. Ishwaran and L.F. James. 2001. Gibbs sampling methods for stick-breaking priors. *J. ASA*, 96(453):161–173.
- M. Johnson, T.L. Griffiths, and S. Goldwater. 2007. Adaptor grammars: A framework for specifying compositional nonparametric Bayesian models. In *NIPS 19*, pages 641–648. MIT Press.
- S.L. Lauritzen. 1989. Mixed graphical association models. *Scand. Jnl. of Statistics*, 16(4):273–306.
- D. Mochihashi and E. Sumita. 2008. The infinite Markov model. In *NIPS 20*, pages 1017–1024. MIT Press.
- C.E. Rasmussen. 2000. The infinite Gaussian mixture model. In *NIPS 12*, pages 554–560. MIT Press.
- M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. 2004. The author-topic model for authors and documents. In *Proc. of the 20th UAI*, pages 487–49.
- E.B. Sudderth and M. Jordan. 2009. Shared segmentation of natural scenes using dependent Pitman-Yor processes. In *NIPS 21*. MIT Press.
- Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei. 2006. Hierarchical Dirichlet processes. *J. ASA*, 101.
- Y.W. Teh. 2006a. A Bayesian interpretation of interpolated Kneser-Ney. Technical Report TRA2/06, School of Computing, National University of Singapore.
- Y.W. Teh. 2006b. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proc. of the 21st ICCL and the 44th ACL*, pages 985–992.
- A. Thomas, D.J. Spiegelhalter, and W.R. Gilks. 1992. BUGS: A program to perform Bayesian inference using Gibbs sampling. In *Bayesian Statistics 4*, pages 837–42. Clarendon Press.
- H. Wallach, C. Sutton, and A. McCallum. 2008. Bayesian modeling of dependency trees using hierarchical Pitman-Yor priors. In *Proc. of the Workshop on Prior Knowledge for Text and Language (with ICML/UAI/COLT)*, pages 15–20.
- F. Wood and Y. W. Teh. 2009. A hierarchical nonparametric Bayesian approach to statistical language model domain adaptation. In *Proc. of the Int. Conf. on Artificial Intelligence and Statistics*, volume 12.
- F. Wood, C. Archambeau, J. Gasthaus, L.F. James, and Y.W. Teh. 2009. A stochastic memoizer for sequence data. In *Proc. of ICML'09*.