Modelling Sequential Text with an Adaptive Topic Model

Lan Du*
Department of Computing
Macquarie University
Sydney, Australia
lan.du@mq.edu.au

Wray Buntine*
Canberra Research Lab
National ICT Australia
Canberra, Australia
wray.buntine@nicta.com.au

Huidong Jin*
CSIRO Mathematics, Informatics
and Statistics,
Canberra, Australia
warren.jin@csiro.au

Abstract

Topic models are increasingly being used for text analysis tasks, often times replacing earlier semantic techniques such as latent semantic analysis. In this paper, we develop a novel adaptive topic model with the ability to adapt topics from both the previous segment and the parent document. For this proposed model, a Gibbs sampler is developed for doing posterior inference. Experimental results show that with topic adaptation, our model significantly improves over existing approaches in terms of perplexity, and is able to uncover clear sequential structure on, for example, Herman Melville's book "Moby Dick".

1 Introduction

Natural language text usually consists of topically structured and coherent components, such as groups of sentences that form paragraphs and groups of paragraphs that form sections. Topical coherence in documents facilitates readers' comprehension, and reflects the author's intended structure. Capturing this structural topical dependency should lead to improved topic modelling. It also seems reasonable to propose that text analysis tasks that involve the structure of a document, for instance, summarisation and segmentation, should also be improved by topic models that better model that structure.

Recently, topic models are increasingly being used for text analysis tasks such as summarisation (Arora and Ravindran, 2008) and segmentation (Misra et al., 2011; Eisenstein and Barzilay, 2008), often times replacing earlier semantic techniques such as latent semantic analysis (Deerwester et al., 1990). Topic models can be improved by better modelling the semantic aspects of text, for instance integrating collocations into the model (Johnson, 2010; Hardisty et al., 2010) or encouraging topics to be more semantically coherent (Newman et al., 2011) based on lexical coherence models (Newman et al., 2010), modelling the structural aspects of documents, for instance modelling a document as a set of segments (Du et al., 2010; Wang et al., 2011; Chen et al., 2009), or improving the underlying statistical methods (Teh et al., 2006; Wallach et al., 2009). Topic models, like statistical parsing methods, are using more sophisticated latent variable methods in order to model different aspects of these problems.

In this paper, we are interested in developing a new topic model which can take into account the structural topic dependency by following the higher level document subject structure, but we hope to retain the general flavour of topic models, where components (*e.g.*, sentences) can be a mixture of topics. Thus we need to depart from the earlier HMM style models, see, *e.g.*, (Blei and Moreno, 2001; Gruber et al., 2007). Inspired by the idea that documents usually exhibits internal structure (*e.g.*, (Wang et al., 2011)), in which semantically related units are clustered together to form semantically structural segments, we treat documents as sequences of segments (*e.g.*, sentences, paragraphs, sections, or chapters). In this way, we can model the topic correlation be-

*This work was partially done when Du was at College of Engineering & Computer Science, the Australian National University when working together with Buntine and Jin there.

Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pages 535–545, Jeju Island, Korea, 12–14 July 2012.

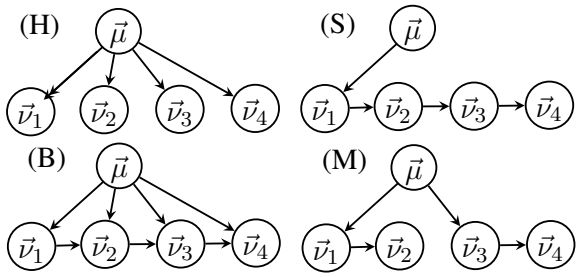


Figure 1: Different structural relationships for topics of sections in a 4-part document, hierarchical (H), sequential (S), both (B) or mixed (M).

tween the segments in a "bag of segments" fashion, *i.e.*, beyond the "bag of words" assumption, and reveal how topics evolve among segments.

Indeed, we were impressed by the improvement in perplexity obtained by the *segmented topic model* (STM) (Du et al., 2010), so we considered the problem of whether one can add sequence information into a structured topic model as well. Figure 1 illustrates the type of structural information being considered, where the vectors are some representation of the content. STM is represented by the hierarchical model. A strictly sequential model would seem unrealistic for some documents, for instance books. A topic model using the strictly sequential model was developed (Du et al., 2012) but it reportedly performs halfway between STM and LDA. In this paper, we develop an adaptive topic model to go beyond a strictly sequential model while allow some hierarchical influence. There are two possible hybrids, one called "mixed" has distinct breaks in the sequence, while the other called "both" overlays both sequence and hierarchy and there could be relative strengths associated with the arrows. We employ the "both" hybrid but use the relative strengths to adaptively allow it to approximate the "mixed" hybrid.

Research in Machine Learning and Natural Language Processing has attempted to model various topical dependencies. Some work considers structure within the sentence level by mixing hidden Markov models (HMMs) and topics on a word by word basis: the aspect HMM (Blei and Moreno, 2001) and the HMM-LDA model (Griffiths et al., 2005) that models both short-range syntactic dependencies and longer semantic dependencies. These models operate at a finer level than we are considering at a segment (like paragraph or section) level. To make a tool like the HMM work at higher levels, one needs to make stronger assumptions, for instance assigning each sentence a single topic and then topic specific word models can be used: the hidden topic Markov model (Gruber et al., 2007) that models the transitional topic structure; a global model based on the generalised Mallows model (Chen et al., 2009), and a HMM based content model (Barzilay and Lee, 2004). Researchers have also considered time-series of topics: various kinds of dynamic topic models, following early work of (Blei and Lafferty, 2006), represent a collection as a sequence of sub-collections in epochs. Here, one is modelling the collections over broad epochs, not the structure of a single document that our model considers.

This paper is organised as follows. We first present background theory in Section 2. Then the new model is presented in Section 3, followed by Gibbs sampling theory and algorithm in Sections 4 and 5 respectively. Experiments are reported in Section 6 with a conclusion in Section 7.

2 Background

The basic topic model is first presented in Section 2.1, as a point of departure. In seeking to develop a general sequential topic model, we hope to go beyond a strictly sequential model and allow some hierarchical influence. This, however, presents two challenges: modelling and statistical inference. Hierarchical inference (and thus sequential inference) over probability vectors can be handled using the theory of hierarchical Poisson-Dirichlet processes (PDPs). This is presented in Section 2.2.

2.1 The LDA model

The benchmark model for topic modelling is latent Dirichlet allocation (LDA) (Blei et al., 2003), a latent variable model of documents. Documents are indexed by i, and words $\vec{w}$ are observed data. The latent variables are $\vec{\mu}_i$ (*the topic distribution* for a document) and $\vec{z}$ (the *topic assignments* for observed words), and the model parameter of $\vec{\phi}_k$'s (*word distributions*). These notation are later extended in Ta-

ble 1. The generative model is as follows:

$$\begin{aligned}
\vec{\phi}_k &\sim \text{Dirichlet}_W\left(\vec{\gamma}\right) && \forall\, k \\
\vec{\mu}_i &\sim \text{Dirichlet}_K\left(\vec{\alpha}\right) && \forall\, i \\
z_{i,l} &\sim \text{Discrete}_K\left(\vec{\mu}_i\right) && \forall\, i, l \\
w_{i,l} &\sim \text{Discrete}_K\left(\vec{\phi}_{z_{i,l}}\right) && \forall\, i, l\,.
\end{aligned}$$

$\text{Dirichlet}_K(\cdot)$ is a K-dimensional Dirichlet distribution. The hyper-parameter $\vec{\gamma}$ is a Dirichlet prior on *word distributions* (*i.e.*, a Dirichlet smoothing on the multinomial parameter $\vec{\phi}_k$ (Blei et al., 2003)) and the Dirichlet prior $\vec{\alpha}$ on topic distributions.

2.2 Hierarchical PDPs

A discrete probability vector $\vec{\mu}$ of finite dimension K is sampled from some distribution $F_\tau(\vec{\mu}_0)$ with a parameter set, say τ, and is also dependent on a parent probability vector $\vec{\mu}_0$ also of finite dimension K. Then a sample of size N is taken according to the probability vector $\vec{\mu}$, represented as $\vec{z} \in \{1, ..., K\}^N$. This data is collected into counts $\vec{n} = (n_1, ..., n_K)$ where n_k is the number of data in $\vec{z}$ with value k and $\sum_k n_k = N$. This situation is represented as follows:

$$\vec{\mu} \sim F_\tau(\vec{\mu}_0); \quad \vec{z}_i \sim \text{Discrete}_K(\vec{\mu}) \text{ for } i = 1, ..., N\,.$$

Commonly in topic modelling, the Dirichlet distribution is used for discrete probability vectors. In this case $F_\tau(\vec{\mu}_0) \equiv \text{Dirichlet}_K(b\vec{\mu}_0)$, $\tau \equiv (K, b)$ where b is the concentration parameter. Bayesian analysis yields a marginalised likelihood, after integrating out $\vec{\mu}$, of

$$p\left(\vec{z}\middle|\tau, \vec{\mu}_0, \text{Dirichlet}\right) = \frac{\text{Beta}\left(\vec{n} + b\vec{\mu}_0\right)}{\text{Beta}\left(b\vec{\mu}_0\right)}\,, \quad (1)$$

where $\text{Beta}(\cdot)$ is the vector valued function normalising the Dirichlet distribution. A problem here is that $p(\vec{z}|b, \vec{\mu}_0)$ is an intractable function of $\vec{\mu}_0$.

Dirichlet processes and Poisson-Dirichlet processes alleviate this problem by using an auxiliary variable trick (Robert and Casella, 2004). That is, we introduce an *auxiliary variable* over which we also sample but do not need to record. The auxiliary variable is the *table count*[1] which is a t_k for each n_k and it represents the number of "tables" over which the n_k "customers" are spread out. Thus the following constraints hold:

$$0 \le t_k \le n_k \quad \text{and} \quad t_k = 0 \text{ iff } n_k = 0\,. \quad (2)$$

When the distribution over probability vectors follows a Poisson-Dirichlet process which has two parameters $\tau \equiv (a, b)$ and the parent distribution $\vec{\mu}_0$, then $F_\tau(\vec{\mu}_0) \equiv \text{PDP}(a, b, \vec{\mu}_0)$. Here a is the *discount parameter*, b the *concentration parameter* and $\vec{\mu}_0$ the base measure. In this case Bayesian analysis yields an augmented marginalised likelihood (Buntine and Hutter, 2012), after integrating out $\vec{\mu}$, of

$$p\left(\vec{z}, \vec{t}\middle|\tau, \vec{\mu}_0, \text{PDP}\right) = \frac{(b|a)_T}{(b)_N} \prod_k S^{n_k}_{t_k,a} \left(\mu_{0,k}\right)^{t_k} \quad (3)$$

where $T = \sum_k t_k$, $(x|y)_N = \prod_{n=0}^{N-1}(x + ny)$ denotes the Pochhammer symbol, $(x)_N = (x|1)_N$, and $S^N_{M,a}$ is a generalized Stirling number that is readily tabulated (Buntine and Hutter, 2012).

There are two fundamental things to notice about Equation (3). Positively, the term in $\vec{\mu}_0$ takes the form of a multinomial likelihood, so we can propagate it up and perform inference on $\vec{\mu}_0$ unencumbered by the functional mess of Equation (1). Thus Poisson-Dirichlet processes allow one to do Bayesian reasoning on hierarchies of probability vectors (Teh, 2006; Teh et al., 2006). Negatively, however, one needs to sample the auxiliary variables $\vec{t}$ leading to some problems: The range of t_k, $\{0, ..., n_k\}$, is broad. Also, contributions from individual data z_i have been lost so the mixing of the MCMC can sometimes be slow. We confirmed these problems on our first implementation of the Adaptive Topic Model presented next in Section 3.

A further improvement on PDP sampling is achieved in (Chen et al., 2011), where another auxiliary variable is introduced, a so-called *table indicator*, that for each datum z_i indicates whether it is the "head of its table" (recall the n_k data are spread over t_k tables, each table has one and only one "head"). Let $r_i = 1$ if z_i is the "head of its table," and zero otherwise. According to this "table" logic, the number of tables for n_k must be the number of data z_i that are also head of table, so $t_k = \sum_{i=1}^N 1_{z_i=k} 1_{r_i=1}$. Moreover, given this definition, the first constraint of Equation (2) on t_k is

[1] Based on the Chinese Restaurant analogy (Teh et al., 2006), each table has a dish, a data value, while data, the customer, is assigned to tables, and multiple tables can serve the same dish.

automatically satisfied. Finally, with t_k tables then there must be exactly t_k heads of table, and we are indifferent about which data are heads of table, thus

$$p\left(\vec{z},\vec{r}\middle|\tau,\vec{\mu}_0,\text{PDP}\right)=p\left(\vec{z},\vec{t}\middle|\tau,\vec{\mu}_0,\text{PDP}\right)\prod_k\binom{n_k}{t_k}^{-1}. \tag{4}$$

When using this marginalised likelihood in a Gibbs sampler, the z_i themselves are usually latent so also sampled, and we develop a blocked Gibbs sampler for (z_i, r_i). Since $\vec{r}$ only appears indirectly through the table counts $\vec{t}$, one does not need to store the $\vec{r}$, instead just resamples an r_i when needed according to the proportion t_w/n_w where $z_i = w$.

3 The proposed Adaptive Topic Model

In this section an adaptive topic model (AdaTM) is developed, a fully structured topic model, by using a PDP to simultaneously model the hierarchical and the sequential topic structures. Documents are assumed to be broken into a sequence of segments. Topic distributions are used to mimic the subjects of documents and subtopics of their segments. The notations and terminologies used in the following sections are given in Table 1.

In AdaTM, the two topic structures are captured by drawing topic distributions from the PDPs with two base distributions as follows. The document topic distribution $\vec{\mu}_i$ and the j^{th} segment topic distribution $\vec{\nu}_{i,j}$ are linearly combined to give a base distribution for the $(j+1)^{th}$ segment's topic distribution $\vec{\nu}_{i,j+1}$. The topic distribution of the first segment, *i.e.*, $\vec{\nu}_{i,1}$, is drawn directly with the base distribution $\vec{\mu}_i$. Call this generative process *topic adaptation*. The graphical representation of AdaTM is shown in Figure 2, and clearly shows the combination of sequence and hierarchy for the topic probabilities. Note the linear combination at each node $\vec{\nu}_{i,j}$ is weighted with latent proportions $\rho_{i,j}$.

Table 1: List of notation for AdaTM

K	number of topics
I	number of documents
J_i	number of segments in document i
$L_{i,j}$	number of words in document i, segment j
W	number of words in dictionary
$\vec{\mu}_i$	document topic probabilities for document i
$\vec{\alpha}$	K-dimensional prior for each $\vec{\mu}_i$
$\vec{\nu}_{i,j}$	segment topic probabilities for document i and segment j
$\rho_{i,j}$	mixture weight associating with the link between $\vec{\nu}_{i.j}$ and $\vec{\nu}_{i,j-1}$
$\vec{\Phi}$	word probability vectors as a $K \times W$ matrix
$\vec{\phi}_k$	word probability vector for topic k, entries in Φ
$\vec{\gamma}$	W-dimensional prior for each $\vec{\phi}_k$
$w_{i,j,l}$	word in document i, segment j, position l
$z_{i,j,l}$	topic for word $w_{i,j,l}$

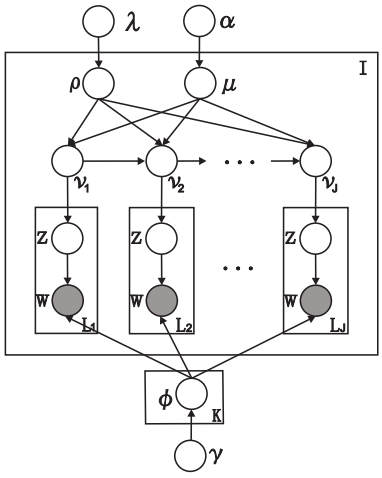


Figure 2: The adaptive topic model: $\vec{\mu}$ is the document topic distribution, $\vec{\nu}_1$, $\vec{\nu}_2$, ..., $\vec{\nu}_J$ are the segment topic distributions, and $\vec{\rho}$ is a set of the mixture weights.

The resultant model for AdaTM is:

$$\begin{aligned}
\vec{\phi}_k &\sim \text{Dirichlet}_W\left(\vec{\gamma}\right) && \forall\, k\\
\vec{\mu}_i &\sim \text{Dirichlet}_K\left(\vec{\alpha}\right) && \forall\, i\\
\rho_{i,j} &\sim \text{Beta}(\lambda_S,\lambda_T) && \forall\, i,j\\
\vec{\nu}_{i,j} &\sim \text{PDP}\left(\rho_{i,j}\vec{\nu}_{i,j-1}+(1-\rho_{i,j})\vec{\mu}_i,a,b\right)\\
z_{i,j,l} &\sim \text{Discrete}_K\left(\vec{\nu}_{i,j}\right) && \forall\, i,j,l\\
w_{i,j,l} &\sim \text{Discrete}_K\left(\vec{\phi}_{z_{i,j,l}}\right) && \forall\, i,j,l\ .
\end{aligned}$$

For notational convenience, let $\vec{\nu}_{i,0} = \vec{\mu}_i$. Assume the dimensionality of the Dirichlet distribution (*i.e.*, the number of topics) is known and fixed, and word probabilities are parameterised with a $K \times W$ matrix $\vec{\Phi} = (\vec{\phi}_1, ..., \vec{\phi}_K)$.

4 Gibbs Sampling Formulation

Given observations and model parameters, computing the posterior distribution of latent variables is infeasible for AdaTM due to the intractable computa-

Table 2: List of statistics for AdaTM

$M_{i,k,w}$	the total number of words in document i with dictionary index w and being assigned to topic k
$M_{k,w}$	total $M_{i,k,w}$ for document i, *i.e.*, $\sum_i M_{i,k,w}$
$\vec{M}_k$	vector of W values $M_{k,w}$
$n_{i,j,k}$	topic count in document i segment j for topic k
$N_{i,j}$	topic total in document i segment j, *i.e.*, $\sum_{k=1}^{K} n_{i,j,k}$
$t_{i,j,k}$	table count in the CPR for document i and paragraph j, for topic k that is inherited back to paragraph $j-1$ and $\vec{\mu}_{i,j-1}$.
$s_{i,j,k}$	table count in the CPR for document i and paragraph j, for topic k that is inherited back to the document and $\vec{\mu}_i$.
$T_{i,j}$	total table count in the CRP for document i and segment j, equal to $\sum_{k=1}^{K} t_{i,j,k}$.
$S_{i,j}$	total table count in the CRP for document i and segment j, equal to $\sum_{k=1}^{K} s_{i,j,k}$.
$\vec{t}_{i,j}$	table count vector of $t_{i,j,k}$'s for segment j.
$\vec{s}_{i,j}$	table count vector of $s_{i,j,k}$'s for segment j.

tion of marginal probabilities. Therefore, we have to use approximate inference techniques. This section proposes a blocked Gibbs sampling algorithm based on methods from Chen et al. (2011). Table 2 lists all statistics needed in the algorithm. Note for easier understanding, terminologies of the Chinese Restaurant Process (Teh et al., 2006) will be used, *i.e.*, customers, dishes and restaurants, correspond to words, topics and segments respectively.

The first major complication, over the use of the hierarchical PDP and Equation (3) and the table indicator trick of Equation (4), is handling the linear combination of $\rho_{i,j}\vec{\nu}_{i,j-1} + (1-\rho_{i,j})\vec{\mu}_i$ used in the PDPs. We manage this as follows: First, Equation (3) shows that a contribution of the form $(\mu_{0,k})^{t_k}$ results. In our case, this becomes

$$\prod_k \left(\rho_{i,j}\nu_{i,j-1,k} + (1-\rho_{i,j})\mu_{i,k}\right)^{t'_{i,j,k}}$$

where $t'_{i,j,k}$ is the corresponding introduced auxiliary variable the table count which is involved with constraints on $n_{i,j,k}+t_{i,j+1,k}$, from Equation (2). To deal with this power of a sum, we break the counts $t'_{i,j,k}$ into two parts, those that contribute to $\vec{\nu}_{i,j-1}$ and those that contribute to $\vec{\mu}_i$. We call these parts $t_{i,j,k}$ and $s_{i,j,k}$ respectively. The product can then be expanded and $\rho_{i,j}$ integrated out. This yields:

$$\text{Beta}\left(S_{i,j}+\lambda_S, T_{i,j}+\lambda_T\right)\prod_k \nu_{i,j-1,k}^{t_{i,j,k}}\mu_{i,k}^{s_{i,j,k}} .$$

The powers $\nu_{i,j-1,k}^{t_{i,j,k}}$ and $\mu_{i,k}^{s_{i,j,k}}$ can then be pushed up to the next nodes in the PDP/Dirichlet hierarchy. Note the standard constraints and table indicators are also needed here.

The precise form of the table indicators needs to be considered as well since there is a hierarchy for them, and this is the second major complication in the model. As discussed in Chen et al. (2011), table indicators are not required to be recorded, instead, randomly sampled in Gibbs cycles. The table indicators when known can be used to reconstruct the table counts $t_{i,j,k}$ and $s_{i,j,k}$, and are reconstructed by sampling from them. For now, denote the table indicators as $u_{i,j,l}$ for word $w_{i,j,l}$.

To complete a formulation suitable for Gibbs sampling, we first compute the marginal distribution of the observations $\vec{w}_{1:I,1:J}$ (words), the topic assignments $\vec{z}_{1:I,1:J}$ and the table indicators $\vec{u}_{1:I,1:J}$. The Dirichlet integral is used to integrate out the document topic distributions $\vec{\mu}_{1:I}$ and the topic-by-words matrix $\vec{\Phi}$, and the joint posterior distribution computed for a PDP is used to recursively marginalise out the segment topic distributions $\vec{\nu}_{1:I,1:J}$. With these variables marginalised out, we derive the following marginal distribution

$$\begin{aligned} p(\vec{z}_{1:I,1:J}, \vec{w}_{1:I,1:J}, \vec{u}_{1:I,1:J} \,|\, \vec{\alpha}, \vec{\gamma}, a, b) \;=\; & \qquad (5)\\ \prod_{i=1}^{I} \frac{\text{Beta}_K\left(\vec{\alpha}+\sum_{j=1}^{J_i}\vec{s}_{i,j}\right)}{\text{Beta}_K\left(\vec{\alpha}\right)} \prod_{k=1}^{K} \frac{\text{Beta}_W\left(\vec{\gamma}+\vec{M}_k\right)}{\text{Beta}_W\left(\vec{\gamma}\right)} & \\ \prod_{i=1}^{I}\prod_{j=1}^{J_i} \text{Beta}\left(S_{i,j}+\lambda_S, T_{i,j}+\lambda_T\right) \frac{(b|a)_{T_{i,j}+S_{i,j}}}{(b)_{N_{i,j}+T_{i,j+1}}} & \\ \prod_{i=1}^{I}\prod_{j=1}^{J_i}\prod_{k=1}^{K} \binom{(n_{i,j,k}+t_{i,j+1,k})}{(t_{i,j,k}+s_{i,j,k})}^{-1} S_{t_{i,j,k}+s_{i,j,k},a}^{n_{i,j,k}+t_{i,j+1,k}} . & \end{aligned}$$

And the following constraints apply:

$$t_{i,j,k} + s_{i,j,k} \le n_{i,j,k} + t_{i,j+1,k}, \qquad (6)$$

$$t_{i,j,k} + s_{i,j,k} = 0 \text{ iff } n_{i,j,k} + t_{i,j+1,k} = 0 \,. \qquad (7)$$

The first constraint falls out naturally when table indicators are used. For convenience of the formulas,

set $t_{i,J_i+1,k} = 0$ (there is no $J_i + 1$ segment) and $t_{i,1,k} = 0$ (the first segment only uses $\vec{\mu}_i$).

Now let us consider again the table indicators $u_{i,j,l}$ for word $w_{i,j,l}$. If this word is in topic k at document i and segment j, then it contributes a count to $n_{i,j,k}$. It also indicates if it contributes a new table, or a count to $t'_{i,j,k}$ for the PDP at this node. However, as we discussed above, this then contributes to either $t_{i,j,k}$ or $s_{i,j,k}$. If it contributes to $t_{i,j,k}$, then it recurses up to contribute a data count to the PDP for document i segment $j-1$. Thus it also needs a table indicator at that node. Consequently, the table indicator $u_{i,j,l}$ for word $w_{i,j,l}$ must specify whether it contributes a table to all PDP nodes reachable by it in the graph.

We define $u_{i,j,l}$ specifically as $u_{i,j,l} = (u_1, u_2)$ such that $u_1 \in [-1, 0, 1]$ and $u_2 \in [1, \cdots, j]$, where u_2 indicates segment denoted by node ν_j up to which $w_{i,j,l}$ contributes a table. Given u_2, $u_1 = -1$ denotes $w_{i,j,l}$ contributes a table count to $s_{i,u_2,k}$ and $t_{i,j',k}$ for $u_2 < j' \leq j$; $u_1 = 0$ denotes $w_{i,j,l}$ does not contribute a table to node u_2, but contributes a table count to $t_{i,j',k}$ for $u_2 < j' \leq j$; and $u_1 = 1$ denotes $w_{i,j,l}$ contributes a table count to each $t_{i,j',k}$ for $u_2 \leq j' \leq j$.

Now, we are ready to compute the conditional probabilities for jointly sampling topics and table indicators from the model posterior of Equation (5).

5 Gibbs Sampling Algorithm

The Gibbs sampler iterates over words, doing a blocked sample of $(z_{i,j,l}, u_{i,j,l})$. The first task is to reconstruct $u_{i,j,l}$ since it is not stored. Since the posterior of Equation (5) does not explicitly mention the $u_{i,j,l}$'s, they occur indirectly through the table counts, and we can randomly reconstruct them by sampling them uniformly from the space of possibilities. Following this, we then remove the values $(z_{i,j,l}, u_{i,j,l})$ from the full set of statistics. Finally, we block sample new values for $(z_{i,j,l}, u_{i,j,l})$ and add them to the statistics. The new $u_{i,j,l}$ is subsequently forgotten and the $z_{i,j,l}$ recorded.

Reconstructing table indicator $u_{i,j,l}$: We start at the node indexed i, j. If $s_{i,j,k}+t_{i,j,k} = 1$ and $n_{i,j,k}+t_{i,j+1,k} > 1$ then no tables can be removed since there is only one table but several customers at the table. Thus $u_{i,j,l} = (u_1, u_2) = (0, j)$ and there is no sampling. Otherwise, by symmetry arguments, we sample u_1 via

$$p(u_1 = -1, 0, 1 | u_2 = j, z_{i,j,l} = k) \propto (s_{i,j,k}, t_{i,j,k}, n_{i,j,k} + t_{i,j+1,k} - s_{i,j,k} - t_{i,j,k}),$$

since there are $n_{i,j,k}+t_{i,j+1,k}$ data distributed across the three possibilities. If after sampling $u_1 = -1$, the data contributes a table count up to $\vec{\mu}_i$ and so $u_{i,j,l} = (u_1, u_2) = (-1, j)$. If $u_1 = 0$, the $u_{i,j,l} = (u_1, u_2) = (0, j)$. Otherwise, the data contributes a table count up to the parent PDP for $\vec{\nu}_{i,j-1}$ and we recurse, repeating the sampling process at the parent node. Note, however, that the table indicator $(0, j')$ for $j' < j$ is equivalent to the table indicator $(1, j' + 1)$ as far as statistics is concerned.

Block sampling $(z_{i,j,l}, u_{i,j,l})$: The full set of possibilities are, for each possible topic $z_{i,j,l} = k$:

- no tables are created, so $u_{i,j,l} = (0, j)$,
- tables are created contributing a table count all the way up to node j' $(\leq j)$ but stop at j' and do not subsequently contribute a count to $\vec{\mu}_i$, so $u_{i,j,l} = (1, j')$,
- tables are created contributing a table count all the way up to node $j' \leq j$ but stop at j' and also subsequently contribute a count to $\vec{\mu}_i$, so $u_{i,j,l} = (-1, j')$.

These three possibilities lead to detailed but fairly straight forward changes to the posterior of Equation (5). Thus a full blocked sampler for $(z_{i,j,l}, u_{i,j,l})$ can be constructed.

Estimates: learnt values of $\vec{\mu}_i$, $\vec{\nu}_{i,j}$, $\vec{\phi}_k$ are needed for evaluation, perplexity calculations, *etc.* These are estimated by taking averages after the Gibbs sampler has burnt in, using the standard posterior means for Dirichlets and Poisson-Dirichlets.

6 Experiments

In the experimental work, we have three objectives: (1) to explore the setting of hyper-parameters, (2) to compare the model with the earlier sequential LDA (SeqLDA) of (Du et al., 2012), STM of (Du et al., 2010) and standard LDA, and (3) to view the results in detail on a number of characteristic problems.

Table 3: Datasets

	#docs	#segs	#words	vocab
Pat-A	500	51,748	2,146,464	16,573
Pat-B	397	9,123	417,631	7,663
Pat-G06	500	11,938	655,694	6,844
Pat-H	500	11,662	562,439	10,114
Pat-F	140	3,181	166,091	4,674
Prince-C	1	26	10,588	3,292
Prince-P	1	192	10.588	3,292
Moby Dick	1	135	88,802	16,223

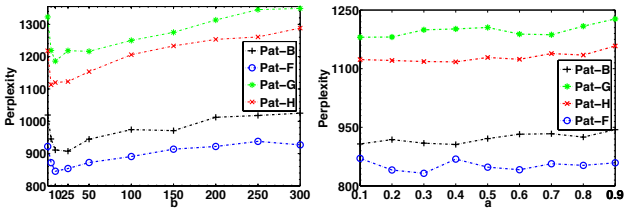
6.1 Datasets

For general testing, five patent datasets are randomly selected from U.S. patents granted in 2009 and 2010. Patents in Pat-A are selected from international patent class (IPC) "A", which is about "HUMAN NECESSITIES"; those in Pat-B are selected from class "B60" about "VEHICLES IN GENERAL"; those in Pat-H are selected from class "H" about "ELECTRICITY"; those in Pat-F are selected from class "F" about "MECHANICAL ENGINEERING; LIGHTING; HEATING; WEAPONS; BLASTING"; and those in Pat-G are selected from class "G06" about "COMPUTING; CALCULATING; COUNTING". All the patents in these five datasets are split into paragraphs that are taken as segments, and the sequence of paragraphs in each patent is reserved in order to maintain the original layout. All the stop words, the top 10 common words, the uncommon words (i.e., words in less than five patents) and numbers have been removed.

Two books used for more detailed investigation are "The Prince" by Niccolò Machiavelli and "Moby Dick" by Herman Melville. They are split into chapters and/or paragraphs which are treated as segments, and only stop-words are removed. Table 3 shows in detail the statistics of these datasets after preprocessing.

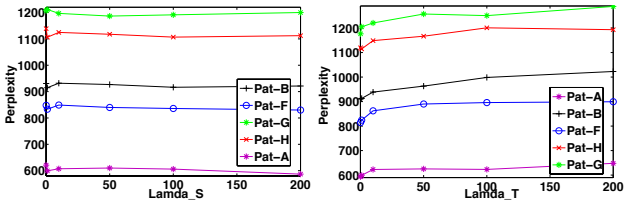
6.2 Design

Perplexity, a standard measure of dictionary-based compressibility, is used for comparison. When reporting test perplexities, the held-out perplexity measure (Rosen-Zvi et al., 2004) is used to evaluate the generalisation capability to the unseen data. This is known to be unbiased. To compute the held-out perplexity, 20% of patents in each data set was randomly held out from training to be used for testing. For this, 1000 Gibbs cycles were done for burn-in followed by 500 cycles with a lag for 100 for parameter estimation.



(a) fix $a = 0$ (b) fix $b = 10$

Figure 3: Analysis of parameters of Poisson-Dirichlet process. (a) shows how perplexity changes with b; (b) shows how it changes with a.



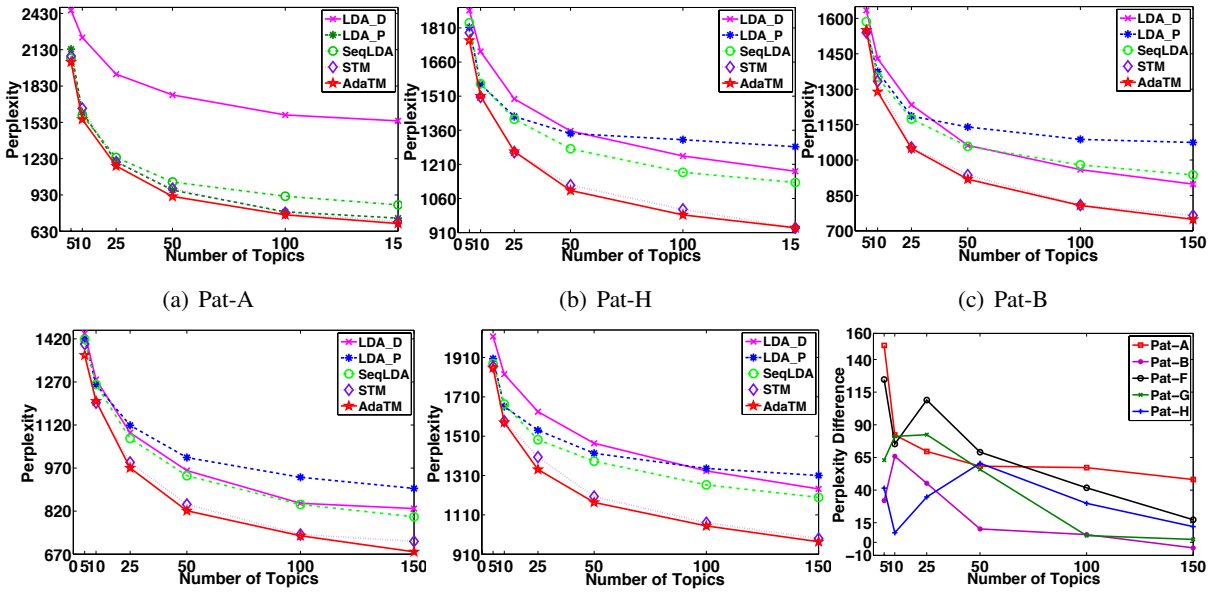
(a) fix $\lambda_T = 1$ (b) fix $\lambda_S = 1$

Figure 4: Analysis of the two parameters for Beta distribution. (a) how perplexity changes with λ_S; (b) how it changes with λ_T.

We implemented all the four models, *e.g.*, LDA, STM, SeqTM and AdaTM in C, and ran them on a desktop with Intel Core i5 CPU (2.8GHz×4), even though our code is not multi-threaded. Perplexity calculations, data input and handling, *etc.*, were the same for all algorithms. We note that the current AdaTM implementation is an order of magnitude slower than regular LDA per major Gibbs cycle.

6.3 Hyper-parameters in AdaTM

Experiments on the impact of the hyper-parameters on the patent data sets were as follows: First, fixing $K = 50$, the Beta parameters $\lambda_T = 1$ and $\lambda_S = 1$, optimise symmetric α, and do two variations *fix-a:* $a = 0.0$, trying $b = 1, 5, 10, 25, ..., 300$, and *fix-b:* $b = 10$, trying $a = 0.1, 0.2, ..., 0.9$. Second, *fix-*λ_T *(fix-*λ_S*)*: fix $a = 0.2$ and $\lambda_T(\lambda_S) = 1$, optimise b and α, change $\lambda_S(\lambda_T) = 0.1, 1, 10, 50, 100, 200$. Figures 3 and 4 show the corresponding plots. Figure 3(b) and Figure 4(a) show that varying the values of a and λ_S does not significantly change the



(a) Pat-A (b) Pat-H (c) Pat-B

(d) Pat-F (e) Pat-G (f) Shuffle

Figure 5: Perplexity comparisons.

perplexity. In contrast, Figure 3(a) shows different b values significantly change perplexity. Therefore, we sought to optimise b. The experiment of fixing $\lambda_S = 1$ and changing λ_T shows a small λ_T is preferred.

6.4 Perplexity Comparison

Perplexity comparisons were done with the default settings $a = 0.2$, $\alpha = 0.1$, $\gamma = 0.01$, $\lambda_S = 1$, $\lambda_T = 1$ and b optimised automatically using the scheme from (Du et al., 2012). Figure 5 shows the results on these five patent datasets for different numbers of topics. LDA_D is LDA run on whole patents, and LDA_P is LDA run on the paragraphs within patents. Table 4 gives the p-values of a one-tail paired t-test for AdaTM versus the others, where lower p-value indicates AdaTM has statistically significant lower perplexity. From this we can see that AdaTM is statistically significantly better than SeqLDA and LDA, and somewhat better than STM.

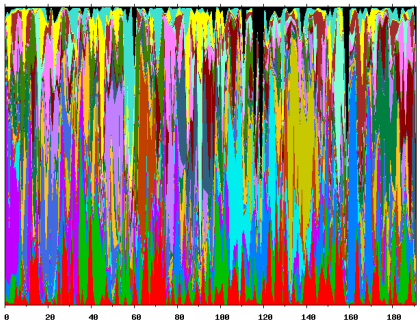
In addition, we ran another set of experiments by randomly shuffling the order of paragraphs in each patent several times before running AdaTM. Then, we calculate the difference between perplexities with and without random shuffle. Figure 5(f) shows the plot of differences in each data sets. The positive difference means randomly shuffling the order of paragraphs indeed increases the perplexity. It can further prove that there does exist sequential topic structure in patents, which confirms the finding in (Du et al., 2012).

6.5 Topic Evolution Comparisons

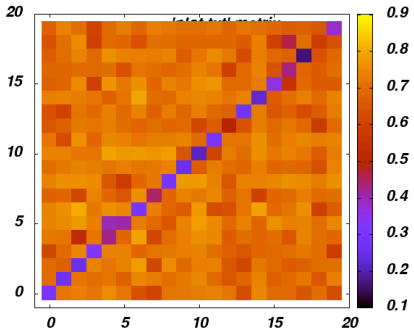
All the comparison experiments reported in this section are run with 20 topics, the upper limit for easy visualisation, and without optimising any parameters. The Dirichlet Priors are fixed as $\alpha_k = 0.1$ and $\gamma_w = 0.01$. For AdaTM, SeqLDA, and STM, $a = 0.0$ and $b = 100$ for "The Prince" and $b = 200$ for "Moby Dick". These settings have proven robust in experiments. To align the topics so visualisations match, the sequential models are initialised using an LDA model built at the chapter level. Moreover, all the models are run at both the chapter and the paragraph level. With the common initialisation, both paragraph level and chapter level models can

Table 4: P-values for one-tail paired t-test on the five patent datasets.

	AdaTM				
	Pat-G	Pat-A	Pat-F	Pat-H	Pat-B
LDA_D	.0001	.0001	.0002	.0001	.0001
LDA_P	.0041	.0030	.0022	.0071	.0096
SeqLDA	.0029	.0047	.0003	.0012	.0023
STM	.0220	.0066	.0210	.0629	.0853



(a) Evolution of paragraph topics for LDA



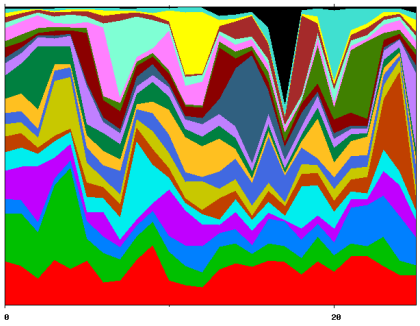
(b) Topic alignment of LDA versus AdaTM topics for chapters

Figure 6: Analysis on "The Prince".

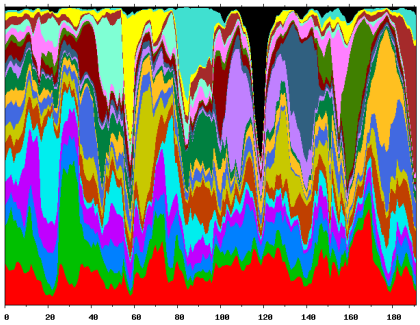
be aligned.

To visualise topic evolution, we use a plot with one colour per topic displayed over the sequence. Figure 6(a) shows this for LDA run on paragraphs of "The Prince". The proportion of 20 topics is the Y-axis, spread across the unit interval. The paragraphs run along the X-axis, so the topic evolution is clearly displayed. One can see there is no sequential structure in this derived by the LDA model, and similar plots result from "Moby Dick" for LDA. Figure 6(b) shows the alignment of topics between the initialising model (LDA+chapters) and AdaTM run on chapters. Each point in the matrix gives the Hellinger distance between the corresponding topics, color coded. The plots for the other models, chapters or paragraphs, are similar so plots like Figure 6(a) for the other models can be meaningfully compared.

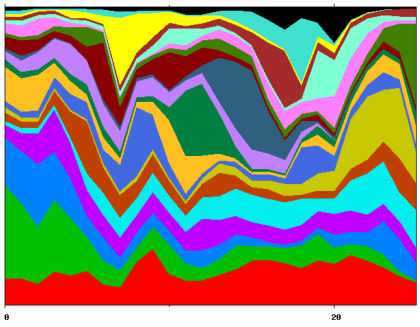
Figure 7 then shows the corresponding evolution plots for AdaTM and SeqLDA on chapters and paragraphs. The contrast of these with LDA is stark. The large improvement in perplexity for AdaTM (see Section 6.4) along with no change in lexical coherence (see Section 6.2) means that the sequential information is actually beneficial statistically. Note that SeqLDA, while exhibiting slightly stronger sequential structure than AdaTM in these



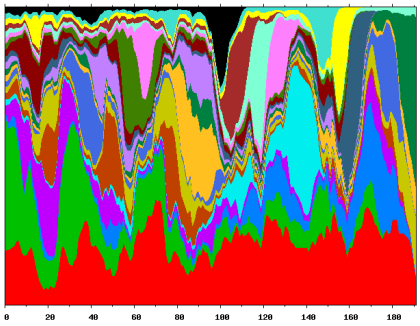
(a) AdaTM on chapters



(b) AdaTM on paragraphs

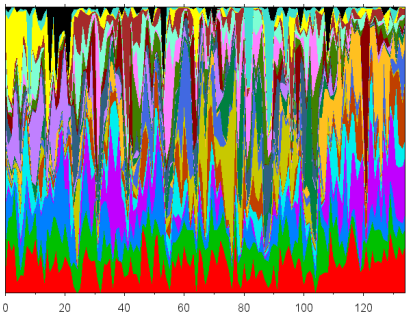


(c) SeqLDA on chapters

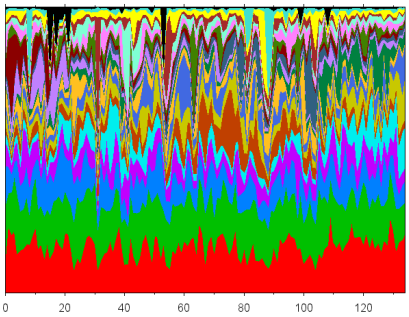


(d) SeqLDA on paragraphs

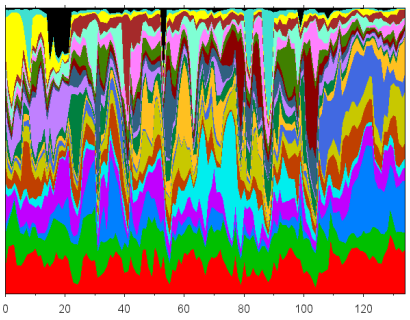
Figure 7: Topic Evolution on "The Prince".



(a) LDA on chapters



(b) STM on Chapters



(c) AdaTM on Chapters

Figure 8: Topic Evolution on "Moby Dick".

figures has significantly worse test perplexity, so its sequential affect is too strong and harming results. Also, note that some topics have different time sequence profiles between AdaTM and SeqLDA. Indeed, inspection of the top words for each show these topics differ somewhat. So while the LDA to AdaTM/SeqLDA topic correspondences are quite good due to the use of LDA initialisation, the correspondences between AdaTM and SeqLDA have degraded. We see that AdaTM has nearly as good sequential characteristics as SeqLDA. Furthermore, segment topic distribution $\nu_{i,j}$ of SeqLDA are gradually deviating from the document topic distribution μ_i, which is not the case for AdaTM.

Results for "Moby Dick" on chapters are comparable. Figure 8 shows similar topic evolution plots for LDA, STM and AdaTM. In contrast, the AdaTM topic evolutions are much clearer for the less frequent topics, as shown in Figure 8(c). Various parts of this are readily interpreted from the storyline. Here we briefly discuss topics by their colour: *black:* Captain Peleg and the business of signing on; *yellow:* inns, housing, bed; *mauve:* Queequeg; *azure:* (around chapters 60-80) details of whales *aqua:* (peaks at 8, 82, 88) pulpit, schools and mythology of whaling.

We see that AdaTM can be used to understand the topics with regards to the sequential structure of a book. In contrast, the sequential nature for LDA and STM is lost in the noise. It can be very interesting to apply the proposed topic models to some text analysis tasks, such as topic segmentation, summarisation, and semantic title evaluation, which are subject to our future work.

7 Conclusion

A model for adaptive sequential topic modelling has been developed to improve over a simple exchangeable segments model STM (Du et al., 2010) and a naive sequential model SeqLDA (Du et al., 2012) in terms of perplexity and its confirmed ability to uncover sequential structure in the topics. One could extract meaningful topics from a book like Herman Melville's "Moby Dick" and concurrently gain their sequential profile. The current Gibbs sampler is slower than regular LDA, so future work is to speed up the algorithm.

Acknowledgments

The authors would like to thank all the anonymous reviewers for their valuable comments. Lan Du was supported under the Australian Research Council's Discovery Projects funding scheme (project numbers DP110102506 and DP110102593). Dr. Huidong Jin was partly supported by CSIRO Mathematics, Informatics and Statistics for this work. 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 Center of Excellence program.

References

R. Arora and B. Ravindran. 2008. Latent Dirichlet allocation and singular value decomposition based multi-document summarization. In *ICDM '08: Proc. of 2008 Eighth IEEE Inter. Conf. on Data Mining*, pages 713–718.

R. Barzilay and L. Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *HLT-NAACL 2004: Main Proceedings*, pages 113–120. Association for Computational Linguistics.

D.M. Blei and J.D. Lafferty. 2006. Dynamic topic models. In *ICML '06: Proc. of 23rd international conference on Machine learning*, pages 113–120.

D.M. Blei and P.J. Moreno. 2001. Topic segmentation with an aspect hidden Markov model. In *Proc. of 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 343–348.

D.M. Blei, A.Y. Ng, and M.I. Jordan. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

W. Buntine and M. Hutter. 2012. A Bayesian view of the Poisson-Dirichlet process. Technical Report arXiv:1007.0296v2, *ArXiv*, Cornell, February.

H. Chen, S.R.K. Branavan, R. Barzilay, and D.R. Karger. 2009. Global models of document structure using latent permutations. In *Proceedings of Human Language Technologies: The 2009 Annual Conf. of the North American Chapter of the Association for Computational Linguistics*, pages 371–379, Stroudsburg, PA, USA. Association for Computational Linguistics.

C. Chen, L. Du, and W. Buntine. 2011. Sampling for the Poisson-Dirichlet process. In *European Conf. on Machine Learning and Principles and Practice of Knowledge Discovery in Database*, pages 296–311.

S.C. Deerwester, S.T. Dumais, T.K. Landauer, G.W. Furnas, and R.A. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407.

L. Du, W. Buntine, and H. Jin. 2010. A segmented topic model based on the two-parameter Poisson-Dirichlet process. *Machine Learning*, 81:5–19.

L. Du, W. Buntine, H. Jin, and C. Chen. 2012. Sequential latent dirichlet allocation. *Knowledge and Information Systems*, 31(3):475–503.

J. Eisenstein and R. Barzilay. 2008. Bayesian unsupervised topic segmentation. In *Proc. of Conf. on Empirical Methods in Natural Language Processing*, pages 334–343. Association for Computational Linguistics.

T.L. Griffiths, M. Steyvers, D.M. Blei, and J.B. Tenenbaum. 2005. Integrating topics and syntax. In *Advances in Neural Information Processing Systems 17*, pages 537–544.

A. Gruber, Y. Weiss, and M. Rosen-Zvi. 2007. Hidden topic markov models. *Journal of Machine Learning Research - Proceedings Track*, 2:163–170.

E.A. Hardisty, J. Boyd-Graber, and P. Resnik. 2010. Modeling perspective using adaptor grammars. In *Proc. of the 2010 Conf. on Empirical Methods in Natural Language Processing*, pages 284–292, Stroudsburg, PA, USA. Association for Computational Linguistics.

M. Johnson. 2010. PCFGs, topic models, adaptor grammars and learning topical collocations and the structure of proper names. In *Proc. of 48th Annual Meeting of the ACL*, pages 1148–1157, Uppsala, Sweden, July. Association for Computational Linguistics.

H. Misra, F. Yvon, O. Capp, and J. Jose. 2011. Text segmentation: A topic modeling perspective. *Information Processing & Management*, 47(4):528–544.

D. Newman, J.H. Lau, K. Grieser, and T. Baldwin. 2010. Automatic evaluation of topic coherence. In *North American Chapter of the Association for Computational Linguistics - Human Language Technologies*, pages 100–108.

D. Newman, E.V. Bonilla, and W. Buntine. 2011. Improving topic coherence with regularized topic models. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 496–504.

C.P. Robert and G. Casella. 2004. *Monte Carlo statistical methods*. Springer. second edition.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. 2004. The author-topic model for authors and documents. In *Proc. of 20th conference on Uncertainty in Artificial Intelligence*, pages 487–494.

Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. 2006. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101:1566–1581.

Y. W. Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proc. of 21st Inter. Conf. on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 985–992.

H. Wallach, D. Mimno, and A. McCallum. 2009. Rethinking LDA: Why priors matter. In *Advances in Neural Information Processing Systems 19*.

H. Wang, D. Zhang, and C. Zhai. 2011. Structural topic model for latent topical structure analysis. In *Proc. of 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, pages 1526–1535, Stroudsburg, PA, USA. Association for Computational Linguistics.