Incorporating Domain Knowledge into Topic Modeling via Dirichlet Forest Priors

David Andrzejewski[*†] ANDRZEJE@CS.WISC.EDU
Xiaojin Zhu[*] JERRYZHU@CS.WISC.EDU
Mark Craven[†*] CRAVEN@BIOSTAT.WISC.EDU

*Department of Computer Sciences, †Department of Biostatistics and Medical Informatics
University of Wisconsin-Madison, Madison, WI 53706 USA

Abstract

Users of topic modeling methods often have knowledge about the composition of words that should have high or low probability in various topics. We incorporate such domain knowledge using a novel Dirichlet Forest prior in a Latent Dirichlet Allocation framework. The prior is a mixture of Dirichlet tree distributions with special structures. We present its construction, and inference via collapsed Gibbs sampling. Experiments on synthetic and real datasets demonstrate our model's ability to follow and generalize beyond user-specified domain knowledge.

1. Introduction

Topic modeling, using approaches such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003), has enjoyed popularity as a way to model hidden topics in data. However, in many applications, a user may have additional knowledge about the composition of words that should have high probability in various topics. For example, in a biological application, one may prefer that the words "termination", "disassembly" and "release" appear with high probability in the same topic, because they all describe the same phase of biological processes. Furthermore, a biologist could automatically extract these preferences from an existing biomedical ontology, such as the Gene Ontology (GO) (The Gene Ontology Consortium, 2000). As another example, an analyst may run topic modeling on a corpus of people's wishes, inspect the resulting topics, and notice that "into, college" and "cure, cancer" all appear with high probability in the same topic. The analyst may want to interactively express the preference that the two sets of words should not appear together, re-run topic modeling, and incorporate additional preferences based on the new results. In both cases, we would like these preferences to guide the recovery of latent topics. Standard LDA lacks a mechanism for incorporating such domain knowledge.

In this paper, we propose a principled approach to the incorporation of such domain knowledge into LDA. We show that many types of knowledge can be expressed with two primitives on word pairs. Borrowing names from the constrained clustering literature (Basu et al., 2008), we call the two primitives Must-Links and Cannot-Links, although there are important differences. We then encode the set of Must-Links and Cannot-Links associated with the domain knowledge using a *Dirichlet Forest prior*, replacing the Dirichlet prior over the topic-word multinomial $p(\text{word}|\text{topic})$. The Dirichlet Forest prior is a mixture of Dirichlet tree distributions with very specific tree structures. Our approach has several advantages: (i) A Dirichlet Forest can encode Must-Links and Cannot-Links, something impossible with Dirichlet distributions. (ii) The user can control the strength of the domain knowledge by setting a parameter η, allowing domain knowledge to be overridden if the data strongly suggest otherwise. (iii) The Dirichlet Forest lends itself to efficient inference via collapsed Gibbs sampling, a property inherited from the conjugacy of Dirichlet trees. We present experiments on several synthetic datasets and two real domains, demonstrating that the resulting topics not only successfully incorporate the specified domain knowledge, but also generalize beyond it by including/excluding other related words not explicitly mentioned in the Must-Links and Cannot-Links.

Appearing in *Proceedings of the 26th International Conference on Machine Learning*, Montreal, Canada, 2009.

2. Related Work

We review LDA using the notation of Griffiths and Steyvers (2004). Let there be T topics. Let $\mathbf{w} = w_1 \dots w_n$ represent a corpus of D documents, with a total of n words. We use d_i to denote the document of word w_i, and z_i the hidden topic from which w_i is generated. Let $\phi_j^{(w)} = p(w|z = j)$, and $\theta_j^{(d)} = p(z = j)$ for document d. The LDA generative model is then:

$$\theta \sim \text{Dirichlet}(\alpha) \quad (1)$$
$$z_i|\theta^{(d_i)} \sim \text{Multinomial}(\theta^{(d_i)}) \quad (2)$$
$$\phi \sim \text{Dirichlet}(\beta) \quad (3)$$
$$w_i|z_i, \phi \sim \text{Multinomial}(\phi_{z_i}) \quad (4)$$

where α and β are hyperparameters for the document-topic and topic-word Dirichlet distributions, respectively. For simplicity we will assume symmetric α and β, but asymmetric hyperparameters are also possible.

Previous work has modeled correlations in the LDA document-topic mixtures using the logistic Normal distribution (Blei & Lafferty, 2006), DAG (Pachinko) structures (Li & McCallum, 2006), or the Dirichlet Tree distribution (Tam & Schultz, 2007). In addition, the concept-topic model (Chemudugunta et al., 2008) employs domain knowledge through special "concept" topics, in which only a particular set of words can be present. Our work complements the previous work by encoding complex domain knowledge on *words* (especially arbitrary Cannot-Links) into a flexible and computationally efficient prior.

3. Topic Modeling with Dirichlet Forest

Our proposed model differs from LDA in the way ϕ is generated. Instead of (3), we have

$$\mathbf{q} \sim \text{DirichletForest}(\beta, \eta)$$
$$\phi \sim \text{DirichletTree}(\mathbf{q})$$

where $\mathbf{q}$ specifies a Dirichlet tree distribution, β plays a role analogous to the topic-word hyperparameter in standard LDA, and $\eta \geq 1$ is the "strength parameter" of the domain knowledge. Before discussing DirichletForest(β, η) and DirichletTree$(\mathbf{q})$, we first explain how knowledge can be expressed using Must-Link and Cannot-Link primitives.

3.1. Must-Links and Cannot-Links

Must-Links and Cannot-Links were originally proposed for constrained clustering to encourage two instances to fall into the same cluster or into separate clusters, respectively. We borrow the notion for topic modeling. Informally, the Must-Link primitive prefers that two words tend to be generated by the same topic, while the Cannot-Link primitive prefers that two words tend to be generated by separate topics. However, since any topic ϕ is a multinomial over words, any two words (in general) always have some probability of being generated by the topic. We therefore propose the following definition:

Must-Link (u, v): Two words u, v have similar probability within any topic, i.e., $\phi_j^{(u)} \approx \phi_j^{(v)}$ for $j = 1 \dots T$. It is important to note that the probabilities can be both large or both small, as long as they are similar. For example, for the earlier biology example we could say Must-Link (termination, disassembly).

Cannot-Link (u, v): Two words u, v should not both have large probability within any topic. It is permissible for one to have a large probability and the other small, or both small. For example, one primitive for the wish example can be Cannot-Link (college, cure).

Many types of domain knowledge can be decomposed into a set of Must-Links and Cannot-Links. We demonstrate three types in our experiments: we can **Split** two or more sets of words from a single topic into different topics by placing Must-Links within the sets and Cannot-Links between them. We can **Merge** two or more sets of words from different topics into one topic by placing Must-Links among the sets. Given a common set of words which appear in multiple topics (such as stopwords in English, which tend to appear in all LDA topics), we can **Isolate** them by placing Must-Links within the common set, and then placing Cannot-Link between the common set and the other high-probability words from all topics. It is important to note that our Must-Links and Cannot-Links are *preferences* instead of hard constraints.

3.2. Encoding Must-Links

It is well-known that the Dirichlet distribution is limited in that all words share a common variance parameter, and are mutually independent except the normalization constraint (Minka, 1999). However, for Must-Link (u, v) it is crucial to control the two words u, v differently than other words.

The Dirichlet tree distribution (Dennis III, 1991) is a generalization of the Dirichlet distribution that allows such control. It is a tree with the words as leaf nodes; see Figure 1(a) for an example. Let $\gamma^{(k)}$ be the Dirichlet tree edge weight *leading into node* k. Let $C(k)$ be the immediate children of node k in the tree, L the leaves of the tree, I the internal nodes, and $L(k)$ the

leaves in the subtree under k. To generate a sample $\phi \sim \text{DirichletTree}(\gamma)$, one first draws a multinomial at each internal node $s \in I$ from $\text{Dirichlet}(\gamma^{C(s)})$, i.e., using the weights from s to its children as the Dirichlet parameters. One can think of it as re-distributing the probability mass reaching s by this multinomial (initially, the mass is 1 at the root). The probability $\phi^{(k)}$ of a word $k \in L$ is then simply the product of the multinomial parameters on the edges from k to the root, as shown in Figure 1(b). It can be shown (Dennis III, 1991) that this procedure gives $\text{DirichletTree}(\gamma) \equiv p(\phi|\gamma) =$

$$\left(\prod_k^L \phi^{(k)^{\gamma^{(k)}-1}}\right)\left(\prod_s^I \frac{\Gamma\left(\sum_k^{C(s)}\gamma^{(k)}\right)}{\prod_k^{C(s)}\Gamma\left(\gamma^{(k)}\right)}\left(\sum_k^{L(s)}\phi^{(k)}\right)^{\Delta(s)}\right)$$

where $\Gamma(\cdot)$ is the standard gamma function, and the notation $\prod_k^L$ means $\prod_{k\in L}$. The function $\Delta(s) \equiv \gamma^{(s)} - \sum_{k\in C(s)}\gamma^{(k)}$ is the difference between the in-degree and out-degree of internal node s. When this difference $\Delta(s) = 0$ for all internal nodes $s \in I$, the Dirichlet tree reduces to a Dirichlet distribution.

Like the Dirichlet, the Dirichlet tree is conjugate to the multinomial. It is possible to integrate out ϕ to get a distribution over word counts directly, similar to the multivariate Pólya distribution: $p(\mathbf{w}|\gamma) =$

$$\prod_s^I\left(\frac{\Gamma\left(\sum_k^{C(s)}\gamma^{(k)}\right)}{\Gamma\left(\sum_k^{C(s)}\left(\gamma^{(k)}+n^{(k)}\right)\right)}\prod_k^{C(s)}\frac{\Gamma\left(\gamma^{(k)}+n^{(k)}\right)}{\Gamma(\gamma^{(k)})}\right) \quad (5)$$

Here $n^{(k)}$ is the number of word tokens in $\mathbf{w}$ that appear in $L(k)$.

We encode Must-Links using a Dirichlet tree. Note that our definition of Must-Link is transitive: Must-Link (u, v) and Must-Link (v, w) imply Must-Link (u, w). We thus first compute the transitive closures of expressed Must-Links. Our Dirichlet tree for Must-Links has a very simple structure: each transitive closure is a subtree, with one internal node and the words in the closure as its leaves. The weights from the internal node to its leaves are $\eta\beta$. The root connects to these internal nodes s with weight $|L(s)|\beta$, where $|\cdot|$ represents the set size. In addition, the root directly connects to other words not in any closure, with weight β. For example, the transitive closure for a Must-Link (A,B) on vocabulary {A,B,C} is simply {A,B}, corresponding to the Dirichlet tree in Figure 1(a).

To understand this encoding of Must-Links, consider first the case when the domain knowledge strength parameter is at its weakest $\eta = 1$. Then in-degree equals out-degree for any internal node s (both are $|L(s)|\beta$), and the tree reduces to a Dirichlet distribution with symmetric prior β: the Must-Links are turned off in this case. As we increase η, the re-distribution of probability mass at s (governed by a Dirichlet under s) has increasing *concentration* $|L(s)|\eta\beta$ but the same uniform base-measure. This tends to redistribute the mass evenly in the transitive closure represented by s. Therefore, the Must-Links are turned on when $\eta > 1$. Furthermore, the mass *reaching* s is independent of η, and can still have a large variance. This properly encodes the fact that we want Must-Linked words to have similar, but not always large, probabilities. Otherwise, Must-Linked words would be forced to appear with large probability in *all* topics, which is clearly undesirable. This is impossible to represent with Dirichlet distributions. For example, the blue dots in Figure 1(c) are ϕ samples from the Dirichlet tree in Figure 1(a), plotted on the probability simplex of dimension three. While it is always true that $p(A) \approx p(B)$, their total probability mass can be anywhere from 0 to 1. The most similar Dirichlet distribution is perhaps the one with parameters (50,50,1), which generates samples close to (0.5, 0.5, 0) (Figure 1(d)).

3.3. Encoding Cannot-Links

Cannot-Links are considerably harder to handle. We first transform them into an alternative form that is amenable to Dirichlet trees. Note that Cannot-Links are not transitive: Cannot-Link (A,B) and Cannot-Link (B,C) does not entail Cannot-Link (A,C). We define a Cannot-Link-graph where the nodes are words[1], and the edges correspond to the Cannot-Links. Then the *connected components* of this graph are independent of each other when encoding Cannot-Links. We will use this property to factor a Dirichlet-tree selection probability later. For example, the two Cannot-Links (A,B) and (B,C) form the graph in Figure 1(e) with a single connected component {A,B,C}.

Consider the subgraph on connected component r. We define its *complement graph* by flipping the edges (on to off, off to on), as shown in Figure 1(f). Let there be $Q^{(r)}$ *maximal cliques* $M_{r1} \ldots M_{rQ^{(r)}}$ in this complement graph. In the following, we simply call them "cliques", but it is important to remember that they are maximal cliques of the complement graph, not the original Cannot-Link-graph. In our example, $Q^{(r)} = 2$ and $M_{r1} = \{A, C\}$, $M_{r2} = \{B\}$. These cliques have the following interpretation: each clique (e.g., $M_{r1} = \{A, C\}$) is the maximal subset of words in the connected component that can "occur together".

[1]When there are Must-Links, all words in a Must-Link transitive closure form a single node in this graph.

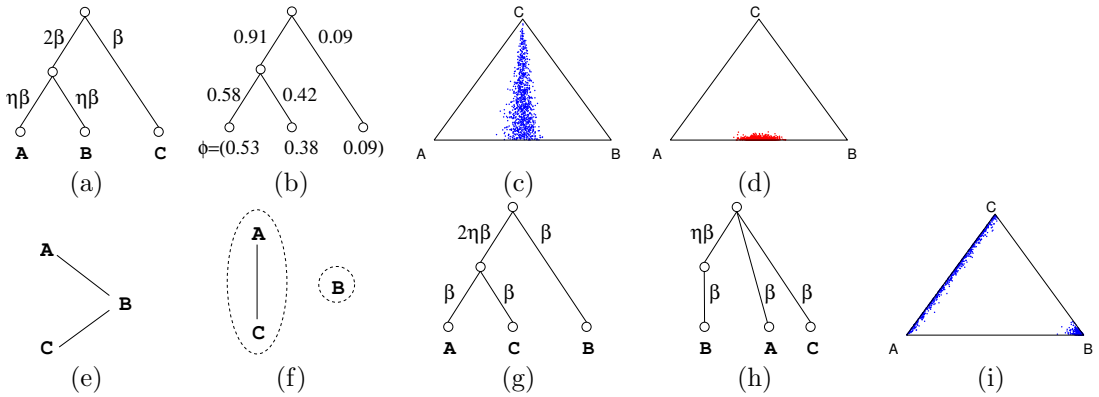

Figure 1. Encoding Must-Links and Cannot-Links with a Dirichlet Forest. (a) A Dirichlet tree encoding Must-Link (A,B) with $\beta = 1, \eta = 50$ on vocabulary {A,B,C}. (b) A sample ϕ from this Dirichlet tree. (c) A large set of samples from the Dirichlet tree, plotted on the 3-simplex. Note $p(A) \approx p(B)$, yet they remain flexible in actual value, which is desirable for a Must-Link. (d) In contrast, samples from a standard Dirichlet with comparable parameters (50,50,1) force $p(A) \approx p(B) \approx 0.5$, and cannot encode a Must-Link. (e) The Cannot-Link-graph for Cannot-Link (A,B) and Cannot-Link (B,C). (f) The complementary graph, with two maximal cliques {A,C} and {B}. (g) The Dirichlet subtree for clique {A,C}. (h) The Dirichlet subtree for clique {B}. (i) Samples from the mixture model on (g,h), encoding both Cannot-Links, again with $\beta = 1, \eta = 50$.

That is, these words are allowed to simultaneously have large probabilities in a given topic without violating any Cannot-Link preferences. By the maximality of these cliques, allowing any word outside the clique (e.g., "B") to also have a large probability will violate at least 1 Cannot-Link (in this example 2).

We discuss the encoding for this single connected component r now, deferring discussion of the complete encoding to section 3.4. We create a mixture model of $Q^{(r)}$ Dirichlet subtrees, one for each clique. Each topic selects exactly one subtree according to probability

$$p(q) \propto |M_{rq}|, \quad q = 1 \dots Q^{(r)}. \tag{6}$$

Conceptually, the selected subtree indexed by q tends to redistribute nearly all probability mass to the words within M_{rq}. Since there is no mass left for other cliques, it is impossible for a word outside clique M_{rq} to have a large probability. Therefore, no Cannot-Link will be violated. In reality, the subtrees are soft rather than hard, because Cannot-Links are only preferences. The Dirichlet subtree for M_{rq} is structured as follows. The subtree's root connects to an internal node s with weight $\eta|M_{rq}|\beta$. The node s connects to words in M_{rq}, with weight β. The subtree's root also directly connects to words not in M_{rq} (but in the connected component r) with weight β. This will send most probability mass down to s, and then flexibly redistribute it among words in M_{rq}. For example, Figures 1(g,h) show the Dirichlet subtrees for $M_{r1} = \{A, C\}$ and $M_{r2} = \{B\}$ respectively. Samples from this mixture model are shown in Figure 1(i), representing multinomials in which no Cannot-Link is violated. Such behavior is not achievable by a Dirichlet distribution, or a single Dirichlet tree[2].

Finally, we mention that although in the worst case the number of maximal cliques $Q^{(r)}$ in a connected component of size $|r|$ can grow exponentially as $O(3^{|r|/3})$ (Griggs et al., 1988), in our experiments $Q^{(r)}$ is no larger than 3, due in part to Must-Linked words "collapsing" to single nodes in the Cannot-Link graph.

3.4. The Dirichlet Forest Prior

In general, our domain knowledge is expressed by a set of Must-Links and Cannot-Links. We first compute the transitive closure of Must-Links. We then form a Cannot-Link-graph, where a node is either a Must-Link closure or a word not present in any Must-Link. Note that the domain knowledge must be "consistent" in that no pairs of words are simultaneously Cannot-Linked and Must-Linked (either explicitly or implicitly through Must-Link transitive closure.) Let R be the number of connected components in the Cannot-Link-graph. Our Dirichlet Forest consists of $\prod_{r=1}^{R} Q^{(r)}$ Dirichlet trees, represented by the template in Figure 2. Each Dirichlet tree has

[2]Dirichlet distributions with very small concentration do have some selection effect. For example, Beta(0.1,0.1) tends to concentrate probability mass on one of the two variables. However, such priors are weak – the "pseudo counts" in them are too small because of the small concentration. The posterior will be dominated by the data, and we would lose any encoded domain knowledge.

R branches beneath the root, one for each connected component. The trees differ in which subtrees they include under these branches. For the r-th branch, there are $Q^{(r)}$ possible Dirichlet subtrees, corresponding to cliques $M_{r1} \dots M_{rQ^{(r)}}$. Therefore, a tree in the forest is uniquely identified by an index vector $\mathbf{q} = (q^{(1)} \dots q^{(R)})$, where $q^{(r)} \in \{1 \dots Q^{(r)}\}$.

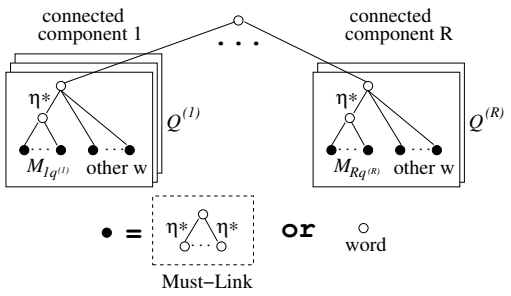

Figure 2. Template of Dirichlet trees in the Dirichlet Forest

To draw a Dirichlet tree $\mathbf{q}$ from the prior DirichletForest(β, η), we select the subtrees independently because the R connected components are independent with respect to Cannot-Links: $p(\mathbf{q}) = \prod_{r=1}^{R} p(q^{(r)})$. Each $q^{(r)}$ is sampled according to (6), and corresponds to choosing a solid box for the r-th branch in Figure 2. The structure of the subtree within the solid box has been defined in Section 3.3. The black nodes may be a single word, or a Must-Link transitive closure having the subtree structure shown in the dotted box. The edge weight leading to most nodes k is $\gamma^{(k)} = |L(k)|\beta$, where $L(k)$ is the set of leaves under k. However, for edges coming out of a Must-Link internal node or going into a Cannot-Link internal node, their weights are multiplied by the strength parameter η. These edges are marked by "$\eta*$" in Figure 2.

We now define the complete Dirichlet Forest model, integrating out ("collapsing") θ and ϕ. Let $n_j^{(d)}$ be the number of word tokens in document d that are assigned to topic j. $\mathbf{z}$ is generated the same as in LDA:

$$p(\mathbf{z}|\alpha) = \left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T}\right)^D \prod_{d=1}^{D} \frac{\prod_{j=1}^{T} \Gamma(n_j^{(d)} + \alpha)}{\Gamma(n_\cdot^{(d)} + T\alpha)}.$$

There is one Dirichlet tree $\mathbf{q}_j$ per topic $j = 1 \dots T$, sampled from the Dirichlet Forest prior $p(\mathbf{q}_j) = \prod_{r=1}^{R} p(q_j^{(r)})$. Each Dirichlet tree $\mathbf{q}_j$ implicitly defines its tree edge weights $\gamma_j^{(\cdot)}$ using β, η, and its tree structure $L_j, I_j, C_j(\cdot)$. Let $n_j^{(k)}$ be the number of word tokens in the corpus assigned to topic j that appear under the node k in the Dirichlet tree $\mathbf{q}_j$. The probability of generating the corpus $\mathbf{w}$, given the trees $\mathbf{q}_{1:T} \equiv \mathbf{q}_1 \dots \mathbf{q}_T$ and the topic assignment $\mathbf{z}$, can be derived using (5): $p(\mathbf{w}|\mathbf{q}_{1:T}, \mathbf{z}, \beta, \eta) =$

$$\prod_{j=1}^{T} \prod_{s}^{I_j} \left(\frac{\Gamma\left(\sum_k^{C_j(s)} \gamma_j^{(k)}\right)}{\Gamma\left(\sum_k^{C_j(s)} (\gamma_j^{(k)} + n_j^{(k)})\right)} \prod_k^{C_j(s)} \frac{\Gamma(\gamma_j^{(k)} + n_j^{(k)})}{\Gamma(\gamma_j^{(k)})} \right).$$

Finally, the complete generative model is

$$p(\mathbf{w}, \mathbf{z}, \mathbf{q}_{1:T}|\alpha, \beta, \eta) = p(\mathbf{w}|\mathbf{q}_{1:T}, \mathbf{z}, \beta, \eta) p(\mathbf{z}|\alpha) \prod_{j=1}^{T} p(\mathbf{q}_j).$$

4. Inference for Dirichlet Forest

Because a Dirichlet Forest is a mixture of Dirichlet trees, which are conjugate to multinomials, we can efficiently perform inference by Markov Chain Monte Carlo (MCMC). Specifically, we use collapsed Gibbs sampling similar to Griffiths and Steyvers (2004). However, in our case the MCMC state is defined by both the topic labels $\mathbf{z}$ and the tree indices $\mathbf{q}_{1:T}$. An MCMC iteration in our case consists of a sweep through both $\mathbf{z}$ and $\mathbf{q}_{1:T}$. We present the conditional probabilities for collapsed Gibbs sampling below.

(Sampling z_i): Let $n_{-i,j}^{(d)}$ be the number of word tokens in document d assigned to topic j, excluding the word at position i. Similarly, let $n_{-i,j}^{(k)}$ be the number of word tokens in the corpus that are under node k in topic j's Dirichlet tree, excluding the word at position i. For candidate topic labels $v = 1 \dots T$, we have

$$p(z_i = v|\mathbf{z}_{-i}, \mathbf{q}_{1:T}, \mathbf{w}) \propto (n_{-i,v}^{(d)} + \alpha) \prod_{s}^{I_v(\uparrow i)} \frac{\gamma_v^{(C_v(s\downarrow i))} + n_{-i,v}^{(C_v(s\downarrow i))}}{\sum_k^{C_v(s)} \left(\gamma_v^{(k)} + n_{-i,v}^{(k)}\right)},$$

where $I_v(\uparrow i)$ denotes the subset of internal nodes in topic v's Dirichlet tree that are ancestors of leaf w_i, and $C_v(s\downarrow i)$ is the unique node that is s's immediate child *and* an ancestor of w_i (including w_i itself).

(Sampling $q_j^{(r)}$): Since the connected components are independent, sampling the tree $\mathbf{q}_j$ factors into sampling the cliques for each connected component $q_j^{(r)}$. For candidate cliques $q' = 1 \dots Q(r)$, we have

$$p(q_j^{(r)} = q'|\mathbf{z}, \mathbf{q}_{-j}, \mathbf{q}_j^{(-r)}, \mathbf{w}) \propto \left(\sum_k^{M_{rq'}} \beta_k\right) \times$$

$$\prod_{s}^{I_{j,r=q'}} \left(\frac{\Gamma\left(\sum_k^{C_j(s)} \gamma_j^{(k)}\right)}{\Gamma\left(\sum_k^{C_j(s)} (\gamma_j^{(k)} + n_j^{(k)})\right)} \prod_k^{C_j(s)} \frac{\Gamma(\gamma_j^{(k)} + n_j^{(k)})}{\Gamma(\gamma_j^{(k)})} \right)$$

where $I_{j,r=q'}$ denotes the internal nodes below the r-th branch of tree $\mathbf{q}_j$, when clique $M_{rq'}$ is selected.

(Estimating ϕ and θ): After running MCMC for sufficient iterations, we follow standard practice (e.g. (Griffiths & Steyvers, 2004)) and use the last sample $(\mathbf{z}, \mathbf{q}_{1:T})$ to estimate ϕ and θ. Because a Dirichlet tree is a conjugate distribution, its posterior is a Dirichlet tree with the same structure and updated edge weights. The posterior for the Dirichlet tree of the j-th topic is $\gamma_j^{post(k)} = \gamma_j^{(k)} + n_j^{(k)}$, where the counts $n_j^{(k)}$ are collected from $\mathbf{z}, \mathbf{q}_{1:T}, \mathbf{w}$. We estimate ϕ_j by the first moment under this posterior (Minka, 1999):

$$\widehat{\phi}_j^{(w)} = \prod_s^{I_j(\uparrow w)} \gamma_j^{post(C_j(s\downarrow w))} \left(\sum_{s'}^{C_j(s)} \gamma_j^{post(s')} \right)^{-1}. \quad (7)$$

The parameter θ is estimated the same way as in standard LDA: $\widehat{\theta}_j^{(d)} = (n_j^{(d)} + \alpha)/(n_\cdot^{(d)} + T\alpha)$.

5. Experiments

Synthetic Corpora: We present results on synthetic datasets to show how the Dirichlet Forest (DF) incorporates different types of knowledge. Recall that DF with $\eta = 1$ is equivalent to standard LDA (verified with the code of (Griffiths & Steyvers, 2004)).

Previous studies often take the last MCMC sample ($\mathbf{z}$ and $\mathbf{q}_{1:T}$), and discuss the topics $\phi_{1:T}$ derived from that sample. Because of the stochastic nature of MCMC, we argue that more insight can be gained if multiple independent MCMC samples are considered. For each dataset, and each DF with a different η, we run a long MCMC chain with 200,000 iterations of burn-in, and take out a sample every 10,000 iterations afterward, for a total of 200 samples. We have some indication that our chain is well-mixed, as we observe all expected modes, and that samples with "label switching" (i.e., equivalent up to label permutation) occur with near equal frequency. For each sample, we derive its topics $\phi_{1:T}$ with (7) and then greedily align the ϕ's from different samples, permuting the T topic labels to remove the label switching effect. Within a dataset, we perform PCA on the baseline ($\eta = 1$) ϕ and project all samples into the resulting space to obtain a common visualization (each row in Figure 3. Points are dithered to show overlap.).

Must-Link (B,C): The corpus consists of six documents over a vocabulary of five "words." The documents are: ABAB, CDCD, and EEEE, each represented twice. We let $T = 2, \alpha = 0.5, \beta = 0.01$. LDA produces three kinds of $\phi_{1:T}$: roughly a third of the time the topics are around $[\frac{A}{2}\frac{B}{2}|\frac{C}{4}\frac{D}{4}\frac{E}{2}]$, which is shorthand for $\phi_1 = (\frac{1}{2}, \frac{1}{2}, 0, 0, 0)$ $\phi_2 = (0, 0, \frac{1}{4}, \frac{1}{4}, \frac{1}{2})$ on the vocabulary ABCDE. Another third are around

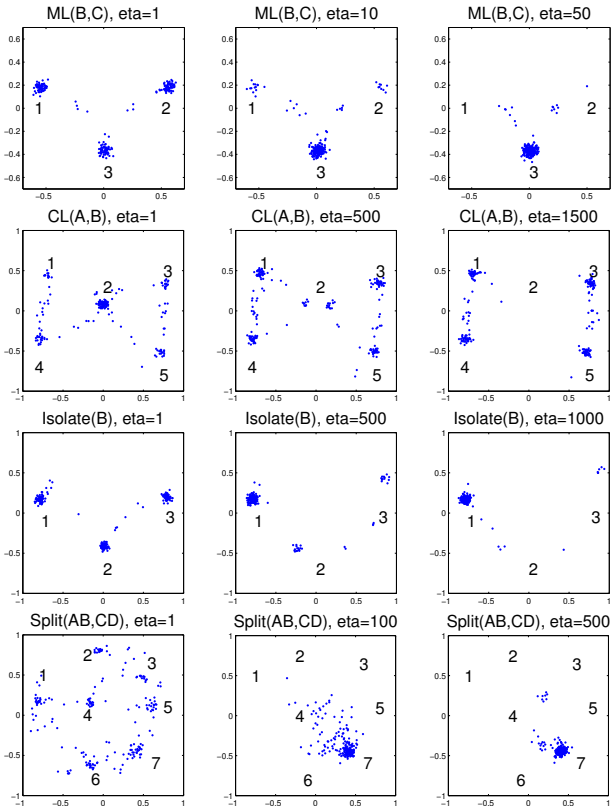

Figure 3. PCA projections of permutation-aligned ϕ samples for the four synthetic data experiments.

$[\frac{A}{4}\frac{B}{4}\frac{E}{2}|\frac{C}{2}\frac{D}{2}]$, and the final third around $[\frac{A}{4}\frac{B}{4}\frac{C}{4}\frac{D}{4}|E]$. They correspond to clusters 1,2 and 3 respectively in the upper-left panel of Figure 3. We add a single Must-Link (B,C). When $\eta = 10$, the data still override our Must-Link somewhat because clusters 1 and 2 do not disappear completely. As η increases to 50, Must-Link overrides the data and clusters 1 and 2 vanish, leaving only cluster 3. That is, running DF and taking the last sample is very likely to obtain the $[\frac{A}{4}\frac{B}{4}\frac{C}{4}\frac{D}{4}|E]$ topics. This is what we want: B and C are present or absent *together* in the topics and they also "pull" A, D along, even though A, D are not in the knowledge we added.

Cannot-Link (A,B): The corpus has four documents: ABCCABCC, ABDDABDD, twice each; $T = 3, \alpha = 1, \beta = 0.01$. LDA produces six kinds of $\phi_{1:T}$ evenly: $[\frac{B}{2}\frac{D}{2}|A|C]$, $[\frac{A}{2}\frac{B}{2}|C|D]$, $[\frac{A}{2}\frac{D}{2}|B|C]$, $[\frac{B}{2}\frac{C}{2}|A|D]$, $[\frac{A}{2}\frac{C}{2}|B|D]$, $[\frac{C}{2}\frac{D}{2}|A|B]$, corresponding to clusters 1–5 and the "lines". We add a single Cannot-Link (A,B). As DF η increases, cluster 2 $[\frac{A}{2}\frac{B}{2}|C|D]$ disappears, because it involves a topic $\frac{A}{2}\frac{B}{2}$ that violates the Cannot-Link. Other clusters become uniformly more likely.

Isolate(B): The corpus has four documents, all of which are ABC; $T = 2, \alpha = 1, \beta = 0.01$. LDA pro-

duces three clusters evenly: $[\frac{A}{2}\frac{C}{2}|B]$, $[\frac{A}{2}\frac{B}{2}|C]$, $[\frac{B}{2}\frac{C}{2}|A]$. We add **Isolate**(B), which is compiled into Cannot-Link (B,A) and Cannot-Link (B,C). The DF's samples concentrate to cluster 1: $[\frac{A}{2}\frac{C}{2}|B]$, which indeed isolates B into its own topic.

Split(AB,CD): The corpus has six documents: ABCDEEEE, ABCDFFFF, each present three times; $\alpha = 0.5, \beta = 0.01$. LDA with $T = 3$ produces a large portion of topics around $[\frac{A}{4}\frac{B}{4}\frac{C}{4}\frac{D}{4}|E|F]$ (not shown). We add **Split**(AB,CD), which is compiled into Must-Link (A,B), Must-Link (C,D), Cannot-Link (B,C), and increase $T = 4$. However, DF with $\eta = 1$ (i.e., LDA with $T = 4$) produces a large variety of topics: e.g., cluster 1 is $[\frac{A}{4}\frac{3B}{8}\frac{3D}{8}|\frac{A}{8}\frac{7F}{8}|C|E]$, cluster 2 is $[\frac{C}{8}\frac{7D}{8}|\frac{3A}{8}\frac{3B}{8}\frac{C}{4}|E|F]$, and cluster 7 is $[\frac{A}{2}\frac{B}{2}|\frac{C}{2}\frac{D}{2}|E|F]$. That is, simply adding one more topic does not clearly separate AB and CD. On the other hand, with η increasing, DF eventually concentrates on cluster 7, which satisfies the Split operation.

Wish Corpus: We now consider *interactive topic modeling* with DF. The corpus we use is a collection of 89,574 New Year's wishes submitted to The Times Square Alliance (Goldberg et al., 2009). Each wish is treated as a document, downcased but without stopword removal. For each step in our interactive example, we set $\alpha = 0.5$, $\beta = 0.1$, $\eta = 1000$, and run MCMC for 2000 iterations before estimating the topics from the final sample. The domain knowledge in DF is accumulative along the steps.

Step 1: We run LDA with $T = 15$. Many of the most probable words in the topics are conventional ("to, and") or corpus-specific ("wish, 2008") stopwords, which obscure the meaning of the topics.

Step 2: We manually create a 50-word stopword list, and issue an **Isolate** preference. This is compiled into Must-Links among this set and Cannot-Links between this set and all other words in the top 50 for all topics. T is increased to 16. After running DF, we end up with two stopword topics. Importantly, with the stopwords explained by these two topics, the top words for the other topics become much more meaningful.

Step 3: We notice that one topic conflates two concepts: enter college and cure disease (top 8 words: "go school cancer into well free cure college"). We issue **Split**("go,school,into,college", "cancer,free,cure,well") to separate the concepts. This is compiled into Must-Links within each quadruple, and a Cannot-Link between them. T is increased to 18. After running DF, one of the topics clearly takes on the "college" concept, picking up related words which we did not explicitly encode in our prior. Another topic does likewise for the "cure" concept (many wishes are like "mom stays cancer free"). Other topics have minor changes.

Table 1. Wish topics from interactive topic modeling

Topic	Top words sorted by $\phi = p$(word|topic)
Merge	love lose weight together forever marry meet
success	health happiness family good friends prosperity
life	life happy best live time long wishes ever years
-	as do not what someone so like don much he
money	out make money up house work able pay own lots
people	no people stop less day every each other another
iraq	home safe end troops iraq bring war return
joy	love true peace happiness dreams joy everyone
family	happy healthy family baby safe prosperous
vote	better hope president paul ron than person bush
Isolate	and to for a the year in new all my
god	god bless jesus everyone loved know heart christ
peace	peace world earth win lottery around save
spam	com call if u 4 www 2 3 visit 1
Isolate	i to wish my for and a be that the
Split	job go great school into good college hope move
Split	mom hope cancer free husband son well dad cure

Step 4: We then notice that two topics correspond to romance concepts. We apply **Merge**("love, forever, marry, together, loves", "meet, boyfriend, married, girlfriend, wedding"), which is compiled into Must-Links between these words. T is decreased to 17. After running DF, one of the romance topics disappears, and the remaining one corresponds to the merged romance topic ("lose", "weight" were in one of them, and remain so). Other previous topics survive with only minor changes. Table 1 shows the wish topics after these four steps, where we place the DF operations next to the most affected topics, and color-code the words explicitly specified in the domain knowledge.

Yeast Corpus: Whereas the previous experiment illustrates the utility of our approach in an interactive setting, we now consider a case in which we use background knowledge from an ontology to guide topic modeling. Our prior knowledge is based on six concepts. The concepts transcription, translation and replication characterize three important *processes* that are carried out at the molecular level. The concepts initiation, elongation and termination describe *phases* of the three aforementioned processes. Combinations of concepts from these two sets correspond to concepts in the Gene Ontology (e.g., GO:0006414 is translational elongation, and GO:0006352 is transcription initiation). We guide our topic modeling using Must-Links among a small set of words for each concept. Moreover, we use Cannot-Links among words to specify that we prefer (i) transcription, translation and replication to be represented in separate topics, and (ii) initiation, elongation and termination to be represented in separate topics. We do not set any preferences between the "process" topics and the "phase" topics, however.

Table 2. Yeast topics. The left column shows the seed words in the DF model. The middle columns indicate the topics in which at least 2 seed words are among the 50 highest probability words for LDA, the "o" column gives the number of other topics (not shared by another word). The right columns show the same topic-word relationships for the DF model.

	LDA									DF									
	1	2	3	4	5	6	7	8	o	1	2	3	4	5	6	7	8	9	10
transcription	•			••					1	•		•						•	
transcriptional	•			••					2	•		•						•	
template				•					1	•		•						•	
translation							••				•				•				
translational							•				•				•				
tRNA									1		•				•				
replication	•								2				•			•		•	
cycle	••												•			•		•	
division		•							3				•			•		•	
initiation	•			•••				•			•	•	•	•					
start		••		•							•	•	•	•					
assembly					•			•	7		•	•	•	•					
elongation				•				•	1									•	
termination							••				•								
disassembly											•								
release									2		•								
stop								•			•								

The corpus that we use for our experiments consists of 18,193 abstracts selected from the MEDLINE database for their relevance to yeast genes. We induce topic models using DF to encode the Must-Links and Cannot-Links described above, and use standard LDA as a control. We set $T = 100, \alpha = 0.5, \beta = 0.1, \eta = 5000$. For each word that we use to seed a concept, Table 2 shows the topics that include it among their 50 most probable words. We make several observations about the DF-induced topics. First, each concept is represented by a small number of topics and the Must-Link words for each topic all occur as highly probable words in these topics. Second, the Cannot-Link preferences are obeyed in the final topics. Third, the topics use the process and phase topics compositionally. For example, DF Topic 4 represents transcription initiation and DF Topic 8 represents replication initiation. Moreover, the topics that are significantly influenced by the prior typically include highly relevant terms among their most probable words. For example, the top words in DF Topic 4 include "TATA", "TFIID", "promoter", and "recruitment" which are all specifically germane to the composite concept of transcription initiation. In the case of standard LDA, the seed concept words are dispersed across a greater number of topics, and highly related words, such as "cycle" and "division" often do not fall into the same topic. Many of the topics induced by ordinary LDA are semantically coherent, but the specific concepts suggested by our prior do not naturally emerge without using DF.

Acknowledgments: This work was supported by NIH/NLM grants T15 LM07359 and R01 LM07050, and the Wisconsin Alumni Research Foundation.

References

Basu, S., Davidson, I., & Wagstaff, K. (Eds.). (2008). *Constrained clustering: Advances in algorithms, theory, and applications*. Chapman & Hall/CRC.

Blei, D., & Lafferty, J. (2006). Correlated topic models. In *Advances in neural information processing systems 18*, 147–154. Cambridge, MA: MIT Press.

Blei, D., Ng, A., & Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Chemudugunta, C., Holloway, A., Smyth, P., & Steyvers, M. (2008). Modeling documents by combining semantic concepts with unsupervised statistical learning. *Intl. Semantic Web Conf.* (pp. 229–244). Springer.

Dennis III, S. Y. (1991). On the hyper-Dirichlet type 1 and hyper-Liouville distributions. *Communications in Statistics – Theory and Methods*, *20*, 4069–4081.

Goldberg, A., Fillmore, N., Andrzejewski, D., Xu, Z., Gibson, B., & Zhu, X. (2009). May all your wishes come true: A study of wishes and how to recognize them. *Human Language Technologies: Proc. of the Annual Conf. of the North American Chapter of the Assoc. for Computational Linguistics*. ACL Press.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proc. of the Natl. Academy of Sciences of the United States of America*, *101*, 5228–5235.

Griggs, J. R., Grinstead, C. M., & Guichard, D. R. (1988). The number of maximal independent sets in a connected graph. *Discrete Math.*, *68*, 211–220.

Li, W., & McCallum, A. (2006). Pachinko allocation: DAG-structured mixture models of topic correlations. *Proc. of the 23rd Intl. Conf. on Machine Learning* (pp. 577–584). ACM Press.

Minka, T. P. (1999). *The Dirichlet-tree distribution* (Technical Report). http://research.microsoft.com/~minka/papers/dirichlet/minka-dirtree.pdf.

Tam, Y.-C., & Schultz, T. (2007). Correlated latent semantic model for unsupervised LM adaptation. *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing* (pp. 41—44).

The Gene Ontology Consortium (2000). Gene Ontology: Tool for the unification of biology. *Nature Genetics*, *25*, 25–29.