Linguistic Extensions of Topic Models

Jordan Boyd-Graber

A Dissertation
Presented to the Faculty
of Princeton University
in Candidacy for the Degree
of Doctor of Philosophy

Recommended for Acceptance
by the Department of
Computer Science
Adviser: David Blei

September 2010

Abstract

Topic models like latent Dirichlet allocation (LDA) provide a framework for analyzing large datasets where observations are collected into groups. Although topic modeling has been fruitfully applied to problems social science, biology, and computer vision, it has been most widely used to model datasets where documents are modeled as exchangeable groups of words. In this context, topic models discover topics, distributions over words that express a coherent theme like "business" or "politics." While one of the strengths of topic models is that they make few assumptions about the underlying data, such a general approach sometimes limits the type of problems topic models can solve.

When we restrict our focus to natural language datasets, we can use insights from linguistics to create models that understand and discover richer language patterns. In this thesis, we extend LDA in three different ways: adding knowledge of word meaning, modeling multiple languages, and incorporating local syntactic context. These extensions apply topic models to new problems, such as discovering the meaning of ambiguous words, extend topic models for new datasets, such as unaligned multilingual corpora, and combine topic models with other sources of information about documents' context.

In Chapter 2, we present latent Dirichlet allocation with WordNet (LDAWN), an unsupervised probabilistic topic model that includes word sense as a hidden variable. LDAWN replaces the multinomial topics of LDA with Abney and Light's distribution over *meanings*. Thus, posterior inference in this model discovers not only the topical domains of each token, as in LDA, but also the meaning associated with each token. We show that considering more topics improves the problem of word sense disambiguation.

LDAWN allows us to separate the representation of meaning from how that meaning is expressed as word forms. In Chapter 3, we extend LDAWN to allow meanings

to be expressed using different word forms in different languages. In addition to the disambiguation provided by LDAWN, this offers a new method of using topic models on corpora with multiple languages.

In Chapter 4, we relax the assumptions of multilingual LDAWN. We present the multilingual topic model for unaligned text (MuTo). Like multilingual LDAWN, it is a probabilistic model of text that is designed to analyze corpora composed of documents in multiple languages. Unlike multilingual LDAWN, which requires the correspondence between languages to be painstakingly annotated, MuTo also uses stochastic EM to simultaneously discover both a matching between the languages while it simultaneously learns multilingual topics. We demonstrate that MuTo allows the meaning of similar documents to to be recovered across languages.

In Chapter 5, we address a recurring problem that hindered the performance of the models presented in the previous chapters: the lack of a local context. We develop the syntactic topic model (STM), a non-parametric Bayesian model of parsed documents. The STM generates words that are both thematically and syntactically constrained, which combines the semantic insights of topic models with the syntactic information available from parse trees. Each word of a sentence is generated by a distribution that combines document-specific topic weights and parse-tree-specific syntactic transitions. Words are assumed to be generated in an order that respects the parse tree. We derive an approximate posterior inference method based on variational methods for hierarchical Dirichlet processes, and we report qualitative and quantitative results on both synthetic data and hand-parsed documents.

In Chapter 6, we conclude with a discussion of how the models presented in this thesis can be applied in real world applications such as sentiment analysis and how the models can be extended to capture even richer linguistic information from text.

Acknowledgements

This work was supported by grants from Google and Microsoft and Office of Naval Research Grant 175-6343.

I have been very fortunate in my time at Princeton to have such a great network of support, both academically and otherwise.

I was fortunate to see a great group of students appear just as I needed them. Jonathan Chang, a frequent and tireless collaborator who was there from the beginning to share my crazy schemes in building group infrastructure and whose wonderful kitsch provided the homey atmosphere of the AI lab; Chong Wang, whose productivity and considerateness I can only hope to achieve; Indraneel Mukherjee, who is willing to think hard about *anything*, no matter how off-the-wall; and Sean Gerrish, who approached his first years at Princeton with all the deliberation and foresight that I lacked.

This core nucleus was part of a broader group of great individuals doing machine learning research at Princeton. I was fortunate to be a part of a large cohort that entered Princeton in the fall of 2004: Umar Syed, Melissa Carroll, and Berk Kapicioglu, who, along with our elders, Zafer Barutcuoglu and Miroslav Dudik, made the machine learning reading group engaging and exciting.

The Princeton Aphasia project is one of the reasons I came to Princeton, and my collaborations and interactions with the group — even if they did not find a way into this thesis — are a key part of my time at Princeton: Maria Klawe, who welcomed me into her home and lab, and selflessly guided me through my start at Princeton; Marilyn Tremaine, who helped me understand what human-computer interaction actually was; Sonya Nikolova, with whom I shared the experience of field trials and drives to Hackensack; Karyn Moffatt, who paradropped in to show us how to do

things right; and Xiaojuan Ma, who was generous and crazy enough to carry some of my ideas through to implementation.

That I could interact with both the machine learning group and the aphasia project are evidence of the close-knit nature of Princeton's computer science department. From Rob Schapire and Moses Charikar, who helped me cut my teeth on machine learning and WordNet when I first arrived, to Brian Kernighan and Robert Sedgewick, who helped make precepting an easy, fulfilling, and rewarding experience, to Perry Cook, who ably took the reigns of the aphasia project, I have been impressed with the friendliness, depth, and intelligence of the entire Princeton computer science faculty. As is often the case, only after leaving Princeton have I realized how wonderful the support staff at Princeton is. Melissa Lawson and Donna O'Leary helped keep red tape to a blessed minimum and were always friendly and approachable.

But this thesis would not have the flavor and direction it has without the support of people outside of computer science, particularly Christiane Fellbaum and Daniel Osherson. Christiane Fellbaum has been a constant source of encouragement and support since my undergraduate days, and Daniel Osherson's insightful questions have helped and guided my research since I set foot at Princeton.

I am thankful to the many people at Princeton who, outside of computer science provided friendship and sanity: Eric Plutz for organizing the Wednesday organ concerts, Andy Yang for helping me actually get to the gym, the Llinás lab — my lab-in law — for being such a fun and welcoming group, and Princeton College Bowl. Princeton College Bowl has been a wonderful source of exciting road trips, odd facts, and lasting friendships. It was the primary outlet for my socialization, and I'm glad that it was a conduit for me to meet and get to know such great people as Leonard Kostovetsky, Guy David, Kunle Demuren, Dan Benediktson, and Sid Parameswaran. Princeton College Bowl was also responsible for introducing me to Ben Gross and Clay Hambrick, who helped me revive and are now sustaining Princeton's only pub

quiz, another cherished memory of my time at Princeton.

But for maintaining my sanity, I must thank most of all my family, particularly my wife Irene. Irene was foolish enough to follow me to Jersey and brave the humidity, snow, and insects that went with the move. She never lets me forget how fortunate I am that she did.

Penultimately, I'd like to thank Dave Blei, for being a patient and understanding advisor. Dave has been a font of wisdom and sage advice, the depth and quality of which I am only now beginning to appreciate. He helped foster the great research environment that I described above, and I'm a grateful that I was able to be a part of it.

Finally, I would like to thank my wonderful committee, whose patience allowed me to finish this thesis while knee deep in a postdoc and whose suggestions made this thesis deeper and more readable:

- David Blei
- Christiane Fellbaum (reader)
- Ryan McDonald (reader)
- Daniel Osherson (non-reader)
- Robert Schapire (non-reader)

Contents

Previous Publication

The work presented here represents expanded versions of the following publications:

- Jordan Boyd-Graber, David Blei, and Xiaojin Zhu. **A Topic Model for Word Sense Disambiguation**. *Empirical Methods in Natural Language Processing*, 2007. (Boyd-Graber et al., 2007)

- Jordan Boyd-Graber, David Blei. **Syntactic Topic Models**. *Neural Information Processing Systems*, 2008. (Boyd-Graber and Blei, 2008)

- Jordan Boyd-Graber, David Blei. **Multilingual Topic Models for Unaligned Text**. *Uncertainty in Artificial Intelligence*, 2009. (Boyd-Graber and Blei, 2009)

The presentation here attempts to provide more explanation, deeper background, and consistent notation.

Other publications during my time at Princeton are not discussed in this thesis, but represent related ideas, applications, or synergistic collaborations:

- Alexander Geyken and **Jordan Boyd-Graber**. **Automatic classification of multi-word expressions in print dictionaries**. *Linguisticae Investigationes*, 2003. (Geyken and Boyd-Graber, 2003)

- **Jordan Boyd-Graber**, Sonya S. Nikolova, Karyn A. Moffatt, Kenrick C. Kin, Joshua Y. Lee, Lester W. Mackey, Marilyn M. Tremaine, and Maria M. Klawe. **Participatory design with proxies: Developing a desktop-PDA system to support people with aphasia**. *Computer-Human Interaction*, 2006. (Boyd-Graber et al., 2006b)

- **Jordan Boyd-Graber** and David M. Blei. **PUTOP: Turning Predominant Senses into a Topic Model for WSD**. *4th International Workshop on Semantic Evaluations*, 2007. (Boyd-Graber and Blei, 2007)

- Sonya S. Nikolova, **Jordan Boyd-Graber**, and Perry Cook. **The Design of ViVA: A Mixed-initiative Visual Vocabulary for Aphasia**. *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, 2009. (Nikolova et al., 2009a)

- Xiaojuan Ma, **Jordan Boyd-Graber**, Sonya S. Nikolova, and Perry Cook. **Speaking Through Pictures: Images vs. Icons**. *ACM Conference on Computers and Accessibility*, 2009. (Ma et al., 2009)

- Jonathan Chang, **Jordan Boyd-Graber**, and David M. Blei. **Connections between the Lines: Augmenting Social Networks with Text**. *Refereed Conference on Knowledge Discovery and Data Mining*, 2009. (Chang et al., 2009a)

- Jonathan Chang, **Jordan Boyd-Graber**, Chong Wang, Sean Gerrish, and David M. Blei. **Reading Tea Leaves: How Humans Interpret Topic Models**. *Neural Information Processing Systems*, 2009. (Chang et al., 2009b)

- Sonya S. Nikolova, **Jordan Boyd-Graber**, Christiane Fellbaum, and Perry Cook. **Better Vocabularies for Assistive Communication Aids: Connecting Terms using Semantic Networks and Untrained Annotators**. *ACM Conference on Computers and Accessibility*, 2009. (Nikolova et al., 2009b)

Update (2012): After the submission of this thesis, some of the unpublished ideas in this thesis have been published elsewhere:

- **Jordan Boyd-Graber** and Philip Resnik. **Holistic Sentiment Analysis Across Languages: Multilingual Supervised Latent Dirichlet Allocation**. *Empirical Methods in Natural Language Processing*, 2010. (Boyd-Graber and Resnik, 2010)

- Sonya S. Nikolova, **Jordan Boyd-Graber**, and Christiane Fellbaum. **Collecting Semantic Similarity Ratings to Connect Concepts in Assistive Communication Tools**. *Modeling, Learning and Processing of Text Technological Data Structures*, 2011. (Nikolova et al., 2011)

- Ke Zhai, **Jordan Boyd-Graber**, Nima Asadi, and Mohamad Alkhouja. **Mr. LDA: A Flexible Large Scale Topic Modeling Package using Variational Inference in MapReduce**. *ACM International Conference on World Wide Web*, 2012. (Zhai et al., 2012)

To Nanny and Papa

Chapter 1

Patterns of Word Usage: Topic Models in Context

Most people are familiar with patterns of word usage. Patterns of how words are written dictate how crossword puzzles and games of scrabble fit together. Patterns of how words sound engage us through song and poetry, and the interplay between meaning and sound both annoy and delight us through the creation of puns.

These intriguing patterns are not just the subject of lighthearted leisure; they also are the foundation of serious academic study. Lexicographers investigate patterns of words' meaning to compose dictionaries. Morphologists' understanding of the internal structure of words helps uncover the history and structure of the world's languages. Syntax, how words come together to form sentences, allows computers to correct your grammar and automatically answer questions like "what is the capital of Botswana?"

Another way of looking at how words are used is at the document level. A document presents a discrete unit of meaning that is relevant to how we often interact with text; we usually think of text at the level of a book, a webpage, or a newspaper article rather than at the word, sentence, or paragraph level. Because the Internet can be considered a large collection of documents, finding the documents that are relevant to

your interests is an important (and profitable) problem in search and advertising.

The question that this thesis asks is if the patterns studied in syntax, morphology, and semantics are influenced by the document. For instance, can we discover that the sentence structure in Faulkner's *Absalom, Absalom* is somehow different from the sentence structure in Hemingway's *The Old Man and the Sea*? Can we discover if similarly spelled words often appear in same documents? Can we discover that a document talks about "disease" even if it never mentions that word?

This thesis attempts to answer questions like these by combining linguistic insights with models that are aware of documents. This chapter introduces the field of topic modeling, which provides a formalism for capturing document context. After introducing topic modeling, this chapter gives a cursory overview of the linguistic formalisms and techniques we will combine with topic modeling to create models that are linguistically relevant and also aware of the context of a document. These models in the following chapters seek to answer questions like contrast between Faulkner and Hemingway.

1.1 Topic Models

A topic model is a model that, given a corpus of documents, discovers the topics that permeate the corpus and assigns documents to these topics. Thus, at a high level, one can think of a topic model as a black box with two outputs: the assignment of words to topics and the assignment of topics to documents.[1] The first output, the topics, are distributions over words; in this thesis (as in most work dealing with topic models) we

[1]We will refine this view later. Other communities might take a different approach to this black box view. For instance, the psychology literature would object to actual inspection of the outputs of latent semantic analysis (LSA) (Landauer and Dumais, 1997). LSA is a matrix factorization-based technique that is the forerunner of probabilistic topic models (Hofmann, 1999). In the psychological community, the outputs of LSA are almost exclusively used as tools to make associations between words and documents, documents and documents, or words and words. However, mathematically, the outputs of both LSA models and topic models are identical: assignments of words to topics and assignments of documents to topics (Griffiths and Steyvers, 2006).

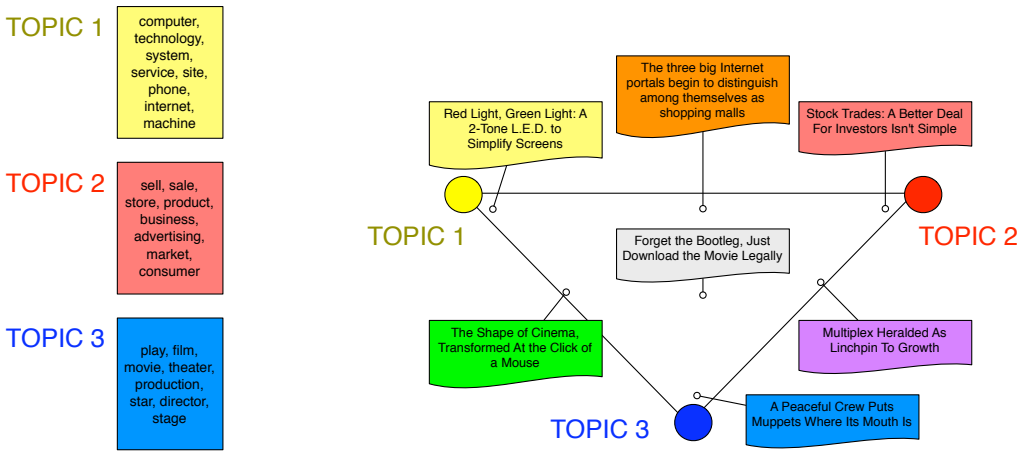
normally present topics as lists of words, as in Figure 1.2(a), which shows three topics discovered from articles in the New York Times. Each word has a probability given a topic; to create the lists in Figure 1.2(a), we sort the words in descending probability and show the top handful of words. In practice, this is usually enough to get a rough understanding of the topic.

The other primary output of a topic model is an assignment of documents to topics. The bag of words representation of each document (see Figure 1.1 for a simple example) is modeled as a mixture of topics. Returning the New York Times corpus, we can see how documents are associated with a few of the topics. For example, the story entitled "Red Light, Green Light: A 2-Tone L.E.D. to Simplify Screens" uses words mostly from the technology topic, while the story entitled "The three big Internet portals begin to distinguish among themselves as shopping malls" also requires the business topic to cover all the words in the text, and the story "Forget the bootleg, just download the movie legally" adds the arts topic.

Original Document	Bag of Words			
one fish, two fish	fish:	8	new:	1
red fish, blue fish	blue:	2	one:	1
black fish, blue fish	black:	1	red:	1
old fish, new fish				

Figure 1.1: The first lines of Dr. Seuss's *One Fish, Two Fish, Red Fish, Blue, Fish* turned into a bag of words representation. From the bag of words representation it's clear that the document is about fish. However, important information is lost. For instance, every word that isn't "fish" modifies a noun (in this case, "fish") and there are clear classes of adjectives that appear often (e.g. cardinal numbers, colors, etc.).

There are a wide variety of methods of finding these topics and the assignment of topics to documents. In this work, we focus on latent Dirichlet allocation (LDA) (Blei et al., 2003). In the literature, LDA is called a probabilistic, generative model. It is generative because it tells the story of how our data came to be, and it is probabilistic because it tells this story using the language of probability. Not all of the details of



(a) Topics (b) Document Assignments to Topics

Figure 1.2: The latent space of a topic model consists of topics, which are distributions over words, and a distribution over these topics for each document. On the left are three topics from a fifty topic LDA model trained on articles from the New York Times. On the right is a simplex depicting the distribution over topics associated with seven documents. The line from each document's title shows the document's position in the topic space.

the story are known in advance, however; some of the pieces are missing. We call these missing pieces latent variables. We use the mathematical technique of statistical inference to discover the latent variables that statistically best explain our observed data.[2] We stress that the latent topics are not observed or annotated in any way; LDA is an unsupervised technique for finding these topics from raw documents. That they correspond to human notions of topics is a product of how language is used (Griffiths and Steyvers, 2006; Chang et al., 2009c).

LDA serves as the starting point for the models discussed in all of the subsequent chapters. Each chapter presents a model that augments the modeling assumptions of LDA with linguistic assumptions.

[2]More precisely, we discover the posterior *distribution* over the latent variables. There are many possible settings of the latent variables that explain our data, but some are better than others. The detail that we discover a distribution over the latent variables should become clear in later chapters that deal with inference; for now, we focus on the intuitions of the models.

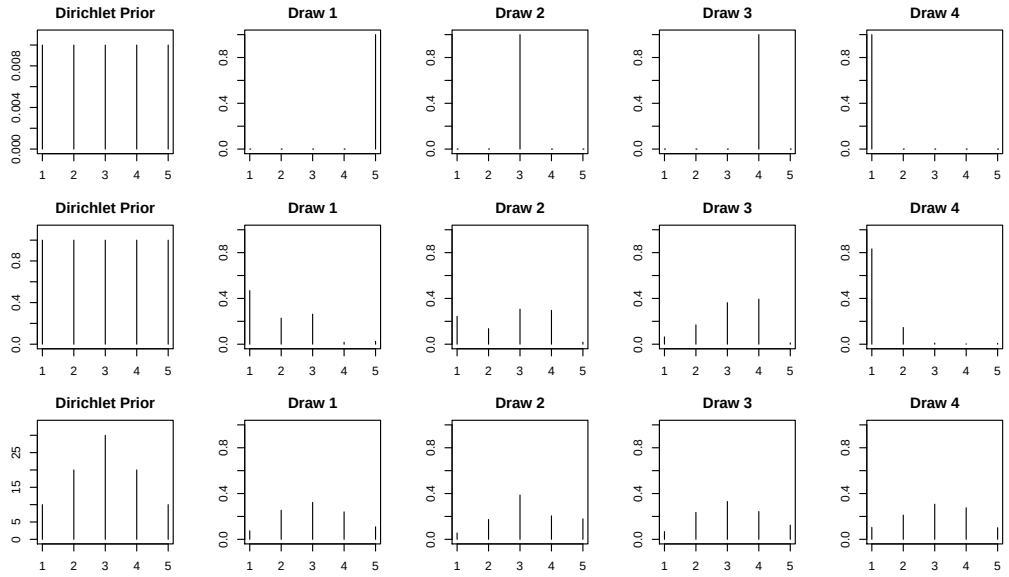


Figure 1.3: Draws of a multinomial distribution of with five components from three different settings of the parameter of a Dirichlet distribution. When the parameter is substantially less than one (top), very sparse distributions are favored. When the parameter is one (middle), all multinomial distributions are equally likely. Finally, if the magnitude of the parameter is large, draws from the Dirichlet are constrained to be close to the distribution defined by the normalized parameter.

1.1.1 Latent Dirichlet Allocation

Before we can formally define LDA, we will first cover some statistical formalities. Readers familiar with the Dirichlet distribution, the multinomial distribution, and how the two distributions are conjugate should feel free to skip to section 1.1.3.

1.1.2 The Dirichlet Distribution

A Dirichlet distribution is a distribution over finite discrete probability distributions. A Dirichlet distribution of dimension K gives a distribution over vectors $\boldsymbol{\theta} \equiv \{\theta_1, \ldots, \theta_K\}$ such that $\sum_k \theta_k = 1$ and $\min_k \theta_k > 0$. It is parameterized by a vector $\{\alpha_1, \ldots, \alpha_K\}$ of non-negative real numbers, and its expected value is $\frac{1}{\sum_k \alpha_k}\{\alpha_1, \ldots, \alpha_K\}$.[3]

[3]We will use bold to denote vectors and normal script (with a subscript) for elements of the vectors. In the particular case of Dirichlet parameters, we symbolize a symmetric prior with an unbolded scalar. This is equivalent to a vector with identical elements.

A Dirichlet distributed random variable is distributed according to

$$\text{Dir}(\boldsymbol{\theta} \,|\, \alpha_1, \ldots, \alpha_K) = \underbrace{\frac{\Gamma\left(\sum_k \alpha_k\right)}{\prod_k \Gamma\left(\alpha_k\right)}}_{\text{normalization}} \prod_k \theta_k^{\alpha_k - 1}.$$

The draw θ is a distribution over discrete observations. Draws from a Dirichlet distribution for various settings of the parameter $\boldsymbol{\alpha}$ are presented in Figure 1.3. When $\boldsymbol{\alpha}$ is relatively small, the Dirichlet distribution favors sparse multinomial distributions where only a few components have weight. This corresponds to our intuition that a document should have a handful of topics rather than gradations of hundreds. When $\boldsymbol{\alpha}$ is all ones, the distribution is uniform; when $\boldsymbol{\alpha}$ is larger, it favors a more peaked distribution around the normalized $\boldsymbol{\alpha}$ parameter, $\frac{\boldsymbol{\alpha}}{|\boldsymbol{\alpha}|}$.

The draws from a Dirichlet distribution are multinomial distributions. Multinomial distributions have parameter θ and are distributions over counts $\{n_1, \ldots, n_K\}$ over K discrete events distributed according to

$$\text{Mult}(\boldsymbol{n} \,|\, \theta_1, \ldots, \theta_K) = \underbrace{\frac{(\sum_k n_k)!}{\prod_k n_k!}}_{\text{normalization}} \prod_k \theta_k^{n_k}.$$

Suggestively, we used the same symbol, θ, for the random variable of the Dirichlet distribution and the parameter for the multinomial distribution. Often, multinomials are modeled as coming from a Dirichlet distribution (later on, we will see that this is also the case in LDA). When we then use Bayes' rule to determine the posterior distribution of a multinomial θ given a set of observed counts $\boldsymbol{n}$

$$p(\boldsymbol{\theta} \,|\, \boldsymbol{n}, \boldsymbol{\alpha}) \propto \prod_k \theta_k^{n_k} \prod_k \theta_k^{\alpha_k - 1} = \prod_k \theta_k^{n_k + \alpha_k - 1}, \tag{1.1}$$

we discover that it has the same form as a Dirichlet distribution parameterized by $\boldsymbol{n} + \boldsymbol{\alpha}$. Thus, the posterior distribution of a multinomial given counts has the same

form as the prior. When such a relationship holds, the prior is said to be *conjugate*. This relationship allows for easier inference in models based on LDA, as we see in Sections 2.2 and 5.3.2.

For both the Dirichlet and the multinomial, we must specify the desired dimension K before we learn our models, which can be difficult especially when we are dealing with unsupervised models. In section 5.2.2, we present techniques that don't require a fixed dimension to model the data.

1.1.3 Defining LDA

Now that we're done with the statistical formalities, we can define LDA more rigorously. LDA assumes the following generative process to create a corpus of M documents with N_d words in document d using K topics $\{\boldsymbol{\beta}_1, \dots \boldsymbol{\beta}_K\}$:

1. For each document $d \in \{1, \dots M\}$:
   (a) Choose the document's topic weights $\boldsymbol{\theta}_d \sim \text{Dir}(\boldsymbol{\alpha})$
   (b) For each word $n \in \{1, \dots N_d\}$:
      i. Choose topic assignment $z_{d,n} \sim \text{Mult}(\boldsymbol{\theta}_d)$
      ii. Choose word $w_{d,n} \sim \text{Mult}(\boldsymbol{\beta}_{z_{d,n}})$

In this process, Dir() represents a Dirichlet distribution (its properties and relationship to the multinomial are discussed in Section 1.1.2), and Mult() is a multinomial distribution. $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are parameters.

The topic weights, $\boldsymbol{\theta}_d$ are vectors of length K, the number of topics in the model, and there is a topic weight distribution for each document d. This distribution corresponds to the simplex shown in Figure 1.2(b). The distribution $\boldsymbol{\theta}_d$ is used to choose the topic assignment z_n for each of the n words in document d. This is a selector variable that chooses which topic $\boldsymbol{\beta}_k$ the observed token w_n comes from. The k^{th} topic $\boldsymbol{\beta}_k$ is a vector of length V; each component corresponds to a word's

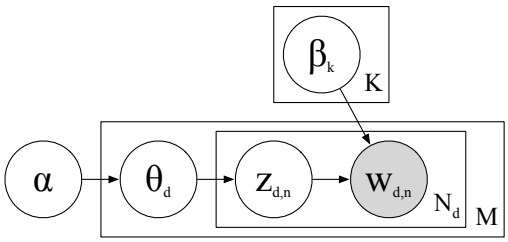


Figure 1.4: The graphical model for latent Dirichlet allocation. Each line represents a possible statistical dependence, shaded nodes are observed, and plates denote replication.

probability in that topic. (We use boldface in equations to denote when symbols are being used as vectors.) In our cartoon picture of topics and documents, $\boldsymbol{\beta}_{z_n}$ is one of the sorted word lists in Figure 1.2(b) and $\boldsymbol{\theta}_d$ is a position in the simplex depicted in Figure 1.2. For example, "stock" would have a high weight in the finance topic but a low weight in the entertainment topic, and the opposite would be true of "movie." Again, we stress that these topics are unobserved in real data; the topics we learn are the statistical signatures of how terms appear together in data and do not reflect any human annotation.

Another tool that we use for describing generative processes is the plate diagram. An example of a plate diagram is Figure 1.4, which represents the generative process for LDA. Each latent variable, observed piece of data, and parameter is represented by a node in a graph. Each possible statistical dependence is represented by a line between the nodes. Observations are represented by shaded nodes, and replication is denoted by rectangular plates; the symbol in the bottom right corner represents how many times the variables inside the plate are repeated. For example, each document d has N_d words, denoted by the inner plate. Inside the inner plate are two nodes, z_n and w_n, representing the topic assignment and the token observed for the n^{th} word. Because these words are observed, the node corresponding to each word is shaded. Because these words are drawn from the z_n^{th} topic β_{z_n}, arrows go from both z and $\boldsymbol{\beta}$ to w_n.

Because we only observe words collected into documents, discovering the topics $\boldsymbol{\beta}_{1:K}$, topic assignments $\boldsymbol{z}_{1:D}$, and per-document topic distributions $\boldsymbol{\theta}_{1:D}$ is a problem of statistical inference. We want to find the random variables that maximize the likelihood

$$p(\boldsymbol{w}|\alpha, \boldsymbol{\beta}) = \prod_d \int_{\boldsymbol{\theta}_d} p(\boldsymbol{\theta}_d \,|\, \alpha) \prod_n \sum_{z_n} p(z_n \,|\, \boldsymbol{\theta}_d) p(w_n \,|\, \boldsymbol{\beta}_{z_n}) d\boldsymbol{\theta}_d$$

However, directly maximizing the likelihood is not tractable because of the coupling between $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. Therefore, in LDA and other models that use LDA as a foundation, we must appeal to approximate inference methods.

1.1.4 Approximate Posterior Inference

Approximate inference allows us to uncover a distribution of the values of the latent variables that best explain our observed data even when the posterior, as in the case of LDA, is intractable. In this work, we expand on inference techniques originally developed for LDA, particularly Markov chain Monte Carlo (MCMC) techniques and variational inference.

We delay discussing inference in depth to preserve the focus here on issues of modeling and to couple explanation of inference techniques with examples specific to the models we develop. MCMC techniques are introduced in Section 2.2, and variational techniques are introduced in Section 5.3.2.

1.1.5 Applications and Related Work

The first topic models were developed in the psychology and text retrieval communities, where they were called latent semantic analysis (LSA) (Deerwester et al., 1990) and probabilistic latent semantic analysis (pLSA) (Hofmann, 1999). Like LDA, the represent documents as a combination of topics, but unlike LDA, LSA and pLSA do

not embody fully generative probabilistic processes. By adopting a fully generative model, LDA exhibits better generalization performance and is more easily used as a module in more complicated models. (Blei et al., 2003; Blei and Lafferty, 2009).

This flexibility has allowed LDA to be used for a wide variety of applications. In a traditional information retrieval setting, Wei and Croft (2006) interpolated LDA with a standard language model to better determine which documents were relevant to a query. Also in the information retrieval domain, Rexa (Information Extraction and Synthesis Laboratory, 2009) is a document browser that exposes the topics of a collection to a user to help guide her to relevant documents using models built on LDA (Rosen-Zvi et al., 2004).

In addition to discovering individual documents, LDA has also served as a tool for finding trends and patterns within the entire corpus. Hall et al (2008) used the topics created by LDA to explore how the field of computational linguistics has changed over time. This is similar to dynamic topic models (Blei and Lafferty, 2006) and continuous time dynamic topic models (Wang et al., 2008), which explicitly model the evolution of topics over time.

Researchers have also extended LDA to model other facets of text corpora such as the words particular authors use (Rosen-Zvi et al., 2004), patterns of citations that appear in documents (Mimno and McCallum, 2007), the latent emotions expressed in product reviews (Blei and McAuliffe, 2007; Titov and McDonald, 2008), part-of-speech labeling (Toutanova and Johnson, 2008), discourse segmentation (Purver et al., 2006), and word sense induction (Brody and Lapata, 2009).

LDA has applications outside text as well; it has been used in understanding images (Li Fei-Fei and Perona, 2005; Blei and Jordan, 2003; Wang et al., 2009; Fergus et al., 2005), computer source code (Maskeri et al., 2008), biology (Pritchard et al., 2000), and music (Hu and Saul, 2009). There are many exciting applications of topic models to many domains; for more information, we suggest one of the reviews of topic

modeling and related literature (Blei and Lafferty, 2009; Griffiths et al., 2007).

1.1.6 Assumptions

LDA's limited assumptions provide the flexibility that allows it to be applied in so many diverse fields. However, LDA's view of text is simplistic and ignores much of the structure that is present in natural language. Because the topic assignments of words are independent given the per-document topic distribution, the order of the words in a document doesn't matter; the order of words is exchangeable. For many applications, such as information retrieval, this so-called "bag of words" assumption is reasonable.

Consider the bag of words representation of Dr. Seuss in Figure 1.1. The bag of words representation does an excellent job of showing that the document is about fish, and LDA would be able find other fish-related documents that it would share topics with. However, the bag of words representation loses much of the relationship that is clear from the original text.

Observe that the text is a sequence of noun phrases modified with "fish" as the head and every word that is not "fish" modifies "fish." Another document with this construction would not be deemed as similar by LDA unless it used the same words, as this regularity is lost in the bag of words representation.

There are contexts, however, where this regulation is important. Suppose you wanted to know what kind of fish appear in a document. There are computational methods to discover that "one," "two," "red," "blue," etc. all change the kind or number of fish being mentioned (how this can be more rigorously codified is discussed in Section 1.2.1). Question answering systems are another example of a context where local syntax is crucially important. Imagine you learned that "A, in its recent acquisition of B," is a good signal that A bought B (Banko et al., 2007). You wouldn't want to throw that information away, and you might want to combine that analysis with the insight offered by topic models.

Relationships between the words that are implicit in a human's reading of the text are also lost. A human realizes that that "red," "blue," and "black" are all colors, but LDA has no reason to correlate "red" with "blue" instead of "marzipan." Similarly, LDA would be unable to associate this document with a German translation, even though the German translation would be half composed of the word "Fisch" and a reader can see that these words are likely related. To LDA "fish" is no more similar to "Fisch" than it is to "marzipan."

Again, there are applications where retaining this information would be of value; imagine searching a database for articles about "fish;" if the person searching also spoke German, you might also want to give them documents that also feature the word "Fisch" prominently.

1.2 Sources for Capturing More Nuanced Patterns from Text

In our efforts to allow LDA-based models to capture these patterns, we draw upon formalisms developed in multiple subfields of linguistics: syntax, semantics, and cross-language investigations of semantics and morphology. We briefly introduce the resources that we use from each of these subfields.

1.2.1 Syntax

When we discussed the Dr. Seuss example, we were able to talk about classes of words like "nouns" because of a shared understanding of language. Syntax provides formal definitions to such terms and has been applied to language in a formal, statistical manner.

One such formalism was developed in the early fifties (Chomsky, 1956; Chomsky and Miller, 1958). The analysis that would eventually become known as a context

free grammar formalized the ideas of syntactic category and part of speech that had been the domain of descriptive grammars. A context free grammar consists of a set of non-terminals (e.g. phrase or part of speech markers), a set of terminals (e.g. the words in a language), production rules (a function giving rules for replacing a non-terminal symbol with other non-terminals or terminals), and a start state (which gives the non-terminal that generates all productions).

For example, a simple context free grammar might be

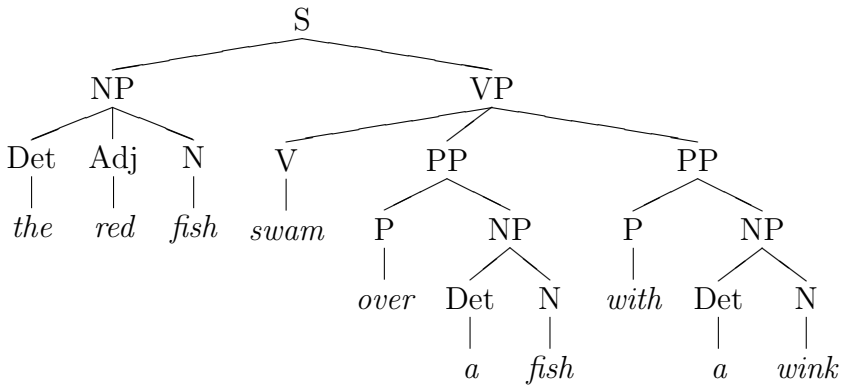
Non-terminal Productions			Terminal Productions		
S	$\rightarrow$	NP VP	V	$\rightarrow$	("swim", "sleep")
VP	$\rightarrow$	V (NP) (Adv) (PP)	P	$\rightarrow$	("on", "over", "with")
NP	$\rightarrow$	(Det) (Adj) N (PP)	Det	$\rightarrow$	("a", "the")
PP	$\rightarrow$	P NP	Adj	$\rightarrow$	("blue", "green", "red")
			Adv	$\rightarrow$	("fast", "lazily")
			N	$\rightarrow$	("rock", "fish", "wink"),

where non-terminals are in quotes and parentheses denote optional productions (e.g. the nonterminal VP can produce either V NP or V NP Adv). The start state is S.

Non-terminals can be any arbitrary symbol, so long as they lead to productions that can generate the language. However, they are typically chosen in a way that is consistent with intuition about language and linguistic theory. [4]

This grammar allows the following analysis for the sentence "The red fish swam over a fish with a wink."

[4]It's not important to understand the exact meaning of the non-terminals in this example, but they roughly correspond to common sense intuitions of the grammatical categories of words. Nouns (N) are the foundation of noun phrases (NP), which are the cluster of words that modify a particular noun (e.g. the noun phrase "the green fish with a rock"). Prepositions (P) can be form prepositional phrases with a noun phrase (e.g. "with a rock"), and verbs can form verb phrases. The important intuition is that this is a framework for encoding the grammatical structure of a sentences.



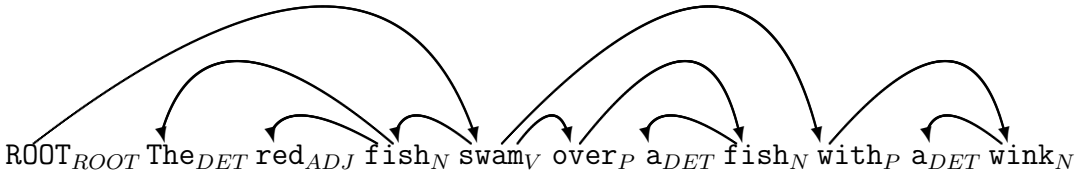
in addition to a parse where the red fish swims over a fish who has the property of winking.

In contrast, the dependency grammar formalism places observed symbols (terminals in the language of a CFG) within the tree structure (Hays, 1964; Nivre, 2005). In such models, internal states are a tuple of terminal symbols and nonterminal states. A dependency grammar gives rules for the possible children of each of the internal nodes and vocabularies for each of the symbols. For example, a dependency grammar might be [5]

Allowed Dependencies			Terminal Productions		
V	→	(N * N Adv)	V	→	(“swim”, “sleep”)
V	→	(N * N)	P	→	(“on”, “over”, “with”)
V	→	(N * N P)	Det	→	(“a”, “the”)
N	→	(Det Adj *)	Adj	→	(“blue”, “green”, “red”)
N	→	(Det *)	Adv	→	(“fast”, “lazily”)
N	→	(Adj *)	N	→	(“rock”, “fish”, “wink”),
N	→	(Det * P)			
N	→	(Adj * P)			
P	→	(* N)			

[5]The grammar presented here is not representative of the breadth of the grammars allowed by the dependency formalism and is meant as a minimal introduction to the representation needed for understanding later chapters. Other formalisms include the Prague school and Mel’chuk’s dependency syntax, which are discussed in the excellent overview (Kübler et al., 2009).

which would offer as one possible interpretation of the sentence as



Dependency parses of a sentence offer an alternative to the phrase structure of context free grammars. Both of these formalisms have been used as the foundation for probabilistic models of language (Charniak, 1997; Eisner, 1996) that can be learned from real-world datasets (Marcus et al., 1994). In practice, dependency grammars are more appropriate for languages where word order is more fluid, such as Czech (Hajič, 1998) and where defining constituency grammars becomes more difficult.[6]

The formalism of syntax allows us to discover patterns of word usage that describe function at a very local level. In contrast to topic models, syntactic models offer a view of local context that allows us to address some of the problems of a bag of words model.

1.2.2 Semantics

A fundamental component of linguistics is the lexicon, which encompasses a speaker's understanding of the vocabulary of a language. Allan (2001) argues that a lexicon must not only posses organization by its surface form, as in a typical dictionary, and syntax, as expressed in the terminal rules in the grammars above; a lexicon must also include organization by meaning.

One such attempt to create a computer-readable lexicon that incorporates meaning into its organization is WordNet (Kilgarriff, 2000). WordNet puts words with the same meaning together into clusters of words called synonym sets (synset). For example, WordNet considers `['car', 'auto', 'automobile', 'machine', 'motorcar']`

[6]The example presented above, which is context free, does not offer this flexibility. However, the formalisms discussed in Footnote [5] do.

to be a single synset.

These synsets are connected by a network of relationships. For example, the car synset has the following links encoded in WordNet:

hyponym is a specific instance of the concept, for example, `[limo, limousine, sedan, saloon]`, `[stock car]`, `[SUV, sport utility, sport utility vehicle]`, and `[convertible]`.

hypernym is less specific instance of the concept, e.g. a child is an example of a parent (thus, it is sometimes called an "is a" relationship). The car synset's hypernym is `[motor vehicle, automotive vehicle]`.

meronym is a part of the parent. Some of the car synset's meronyms include `[horn, hooter, car horn, automobile horn]` and `[cowl, hood]`. Conversely, the opposite relationship is that the car synset is a holonym of `[cowl, hood]`.

The network of WordNet is based on psychological experiments that suggest that the human lexicon is based on properties of inheritance (Miller, 1990). For example, when asked if a canary is yellow, has feathers, has a spine, or has skin, each question takes longer to answer than the last because it asks about properties of more distant hyperyms (bird, chordate, animal).

WordNet is the de facto resource for annotating the sense of ambiguous words in a text.[7] For example, "cowl" has two meanings in WordNet: the hood of a car and the garment worn by a monk. Determining whether an appearance of the string "cowl" is talking about a piece of clothing or a car part is called word sense disambiguation (WSD). In Chapter 2, we develop an algorithm for this important NLP problem that builds upon topic models.

[7]Kilagrriff (2000) notes that "not using [WordNet] requires explanation and justification." Almost all word sense disambiguation bakeoffs use WordNet as the sense inventory (Kilgarriff and Rosenzweig, 2000), and the first sense-disambiguated corpora were constructed using WordNet as a sense inventory (Miller et al., 1993). While there are alternative resources, none have been as widely embraced as WordNet.

1.2.3 Linguistic Representation of Multiple Languages

The formalism of WordNet has been applied to many languages from different language families, e.g. Japanese (Isahara et al., 2008), Hebrew (Ordan and Wintner, 2007), and German (Hamp and Feldweg, 1997), using both manual and automatic methods (Sagot and Fišer, 2008). Thus, the representation of meaning has been made independent of the actual lexical realization of those meanings.

In addition to explicit connections being drawn between languages, there are deep connections that implicitly exist between languages. These implicit connections are so thoroughly studied that eighty years ago, Sapir's assessment of the state of linguistics (1929) declared that there was little left to be done in describing the processes that change a language over time or change one language into another.

Despite the field being well studied even at the start of the twentieth century, understanding the common foundations of the worlds' languages is the focus of an active branch of linguistics called typology (Ramat, 1987; McMahon and McMahon, 2005). There are deep and subtle connections across languages (which we will not discuss here), but some other commonalities between languages are obvious to even casual observers. Many common terms (called "cognates") appear across languages with similar meaning, sound, and representation (Carroll, 1992). In addition to common words that appear in multiple language because of borrowing or a shared history, proper names are often direct transliterations (Knight and Graehl, 1997; Sproat et al., 2006). In sum, we can often discover simple, consistent relationships that algorithms like LDA can use to examine documents in multiple languages simultaneously.

1.2.4 Roadmap for Subsequent Chapters: Adding Linguistic Intuitions

In the next chapters, we will use these tools to enhance LDA's ability to represent of text. In Chapter 2, we combine LDA's automatic discovery of implicit meaning with the human-generated meaning expressed through WordNet. We imbue WordNet's explicit representation of meaning into a topic model to create a model called LDAWN (LDA with WordNet). By modeling the meaning of each word as a latent variable, we allow LDA to perform word sense disambiguation.

The next two chapters discuss how to use non-parallel multilingual data with topic models, which allow the many applications developed using topic models (as discussed in Section 1.1.5) to be applied to a broader range of corpora. In Chapter 3, we extend LDAWN to use WordNet to provide a way of modeling topics and meaning that is independent of language. Multilingual LDAWN uses WordNet as a common bridge to express the meaning of a topic, allowing the discovery of topics that are consistent across languages.

Although many languages have a WordNet, not every language does, and for many languages a nascent WordNet might be insufficient or licensing restrictions might preclude its use. In Chapter 4, we demonstrate a model that allows the discovery of multilingual topics without a prespecified WordNet across languages. It does so by learning correspondences across languages as it simultaneously discovers the latent topics in a document.

Finally, in Chapter Chapter 5, we present a new method for moving beyond the bag of words representation of topic models by explicitly modeling the patterns of dependency representations of sentences discussed in Section 1.2.1. Such models give the "best of both worlds," allowing the semantic properties captured by topic models to complement the syntactic information captured by the syntactic representations of language discussed in Section 1.2.1.

Chapter 2

LDAWN: Adding a Semantic Ontology to Topic Models

Latent Dirichlet Allocation (LDA), as we discussed in the previous chapter, does not model any inherent connection between words. While LDA can discover that words are used in similar context, it has no way of knowing that a "collie" and a "spaniel" are both dogs or that "burn" is a specific type of "injury." But it's clear that humans share common notions of how words are connected semantically (Miller and Charles, 1991; Boyd-Graber et al., 2006a); if you ask different people how similar "dog" and "collie" are, you will get similar answers.

If a topic model could model meaning directly, rather than working through the intermediate representation of words, we could explicitly model the meaning of a document. As a side effect, explicitly modeling the sense of a word also allows us to apply topic models to new applications.

One such application is word sense disambiguation (WSD), the task of determining the meaning of an ambiguous word in its context. It is an important problem in natural language processing (NLP) because effective WSD can improve systems for tasks such as information retrieval, machine translation, and summarization. In

this chapter, we develop latent Dirichlet allocation with WordNet (LDAWN), a generative probabilistic topic model for WSD where the sense of the word is a hidden random variable that is inferred from data.

There are two central advantages to this approach. First, with LDAWN we automatically learn the context in which a word is disambiguated. Rather than disambiguating at the sentence-level or the document-level, our model uses the other words that share the same hidden topic across many documents.

Second, LDAWN is a fully-fledged generative model. Generative models are modular and can be easily combined and composed to form more complicated models. (As a canonical example, the ubiquitous hidden Markov model is a series of mixture models chained together.) Thus, developing a generative model for WSD gives other generative NLP algorithms a natural way to take advantage of the hidden senses of words.

While topic models capture the polysemous use of words (Griffiths and Steyvers, 2006), they do not carry the explicit notion of *sense* that is necessary for WSD. LDAWN extends the topic modeling framework to include a hidden meaning in the word generation process. In this case, posterior inference discovers both the topics of the corpus and the meanings assigned to each of its words.

We begin by introducing a disambiguation scheme based on probabilistic walks over the WordNet hierarchy (Section 2.1), and we then embed the WordNet-Walk in a topic model, where each topic is associated with walks that prefer different neighborhoods of WordNet (Section 2.1.1). Next, we derive a Gibbs sampling algorithm for approximate posterior inference that learns the senses and topics that best explain a corpus (Section 2.2). Finally, we evaluate our system on real-world WSD data, discuss the properties of the topics and disambiguation accuracy results, and draw connections to other WSD algorithms from the research literature.

2.1 Probabilistic Approaches that Use WordNet

The WordNet-Walk is a probabilistic process of word generation that is based on the hyponomy relationship in WordNet (Miller, 1990). WordNet, a lexical resource designed by psychologists and lexicographers to mimic the semantic organization in the human mind, links "synsets" (short for synonym sets) with myriad connections. The specific relation we're interested in, hyponomy, points from general concepts to more specific ones and is sometimes called the "is-a" relationship.

As first described by Abney and Light (1999), we imagine an agent who starts at synset `[entity]`, which points to every noun in WordNet by some sequence of hyponomy relations, and then chooses the next node in its random walk from the hyponyms of its current position. The agent repeats this process until it reaches a leaf node, which corresponds to a single word (each of the synset's words are unique leaves of a synset in our construction). For an example of all the paths that might generate the word "colt" see Figure 2.1. The WordNet-Walk is parameterized by a set of distributions over children for each synset s in WordNet, $\boldsymbol{\beta}_s$.

2.1.1 A topic model for WSD

The WordNet-Walk has two important properties. First, it describes a random process for word generation. Thus, it is a distribution over words and thus can be integrated into any generative model of text, such as topic models. Second, the synset that produces each word is a hidden random variable. Given a word assumed to be generated by a WordNet-Walk, we can use posterior inference to predict which synset produced the word.

These properties allow us to develop LDAWN, which is a fusion of these WordNet-Walks and latent Dirichlet allocation (LDA) (Blei et al., 2003). LDA assumes that there are K "topics," multinomial distributions over words, which describe a document

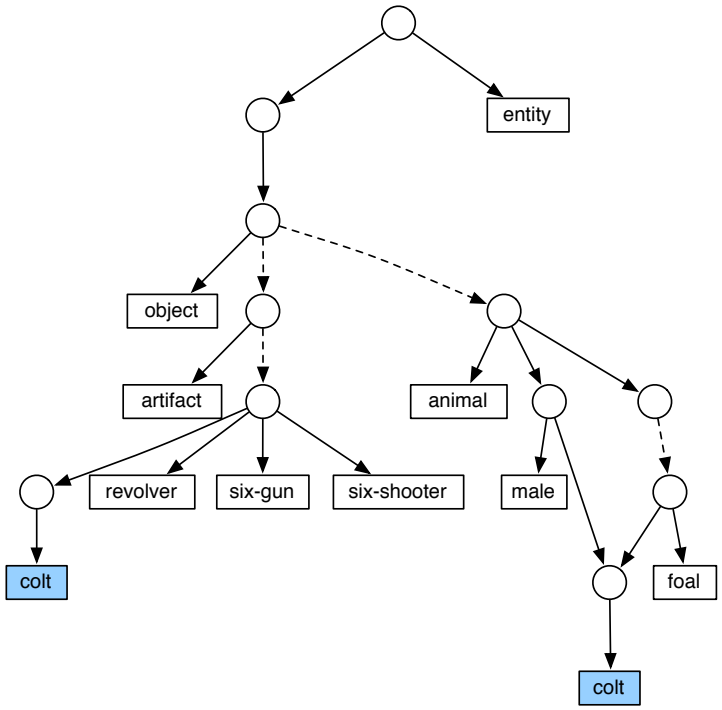


Figure 2.1: The possible paths to reach the word "colt" in WordNet. Dashed lines represent omitted links. All words in the synset containing "revolver" are shown, but only one word from other synsets is shown.

collection. Each document exhibits multiple topics, and each word in each document is associated with one of them.

Although the term "topic" evokes a collection of ideas that share a common theme and although the topics derived by LDA evince semantic coherence (Figure 1.2(a)), there is no reason to believe this would be true of the most likely multinomial distributions that could have created the corpus given the assumed generative model. That semantically similar words are likely to occur together is a byproduct of how language is used.

In LDAWN, we replace the multinomial topic distributions with a WordNet-Walk, as described above. LDAWN assumes a corpus is generated by the following process (for an overview of the notation, see Table 2.1).

1. For each topic, $k \in \{1, \ldots, K\}$

Symbol	Meaning
K	number of topics
$\boldsymbol{\beta}_{k,h}$	multinomial probability vector over the successors of synset h in topic k
S	scalar that, when multiplied by τ_h gives the prior for $\beta_{k,h}$
$\boldsymbol{\tau}_h$	normalized vector whose i^{th} entry, when multiplied by S, gives the prior probability for going from synset h to its i^{th} child
$\boldsymbol{\theta}_d$	multinomial distribution over the topics that generate document d
α	prior for θ
$z_{d,n}$	assignment for word n in document d to a topic
$\Lambda_{d,n}$	assignment for word n in document d to a path through a WordNet ending at a word.
$\lambda_{i,j}$	one link in a path λ going from synset i to synset j.
$t_{d,i}^{-u}$	The number of words in document d that have been assigned topic i (ignoring word u).
$b_{k,p,c}^{-u}$	The number of transitions in the WordNet-Walk for topic k that have been observed going from synset p to its child synset c (not counting the path assignment of word u).

Table 2.1: A summary of the notation used in this chapter. Bold vectors correspond to collections of variables (i.e. z_u refers to a topic of a single word, but $\boldsymbol{z}_{1:D}$ are the topics assignments of words in document 1 through D).

   (a) For each synset s in the hierarchy, draw transition probabilities $\boldsymbol{\beta}_{k,h} \sim \text{Dir}(S\boldsymbol{\tau}_h)$.

2. For each document $d \in \{1, \ldots, D\}$

   (a) Select a topic distribution $\boldsymbol{\theta}_d \sim \text{Dir}(\tau)$

   (b) For each word $n \in \{1, \ldots, N_d\}$

      i. Select a topic $z_{d,n} \sim \text{Mult}(\theta_d)$

      ii. Create a path $\Lambda_{d,n}$ starting with the root node h_0 of the hierarchy.[1]

      iii. Given the current node h, choose a new node in the path:

         A. Choose the next node in the walk $h' \sim \text{Mult}(\boldsymbol{\beta}_{z_{d,n},h})$; add the step $\lambda_{h,h'}$ to the path $\Lambda_{d,n}$.

         B. If h' is a leaf node, generate the associated word w_n. Otherwise, repeat with h' as the current node.

Only the the words of a document is observed; everything else is hidden. Thus, given a collection of documents, our goal is to perform *posterior inference*, which is

[1] For nouns in WordNet, this is the "entity" concept.

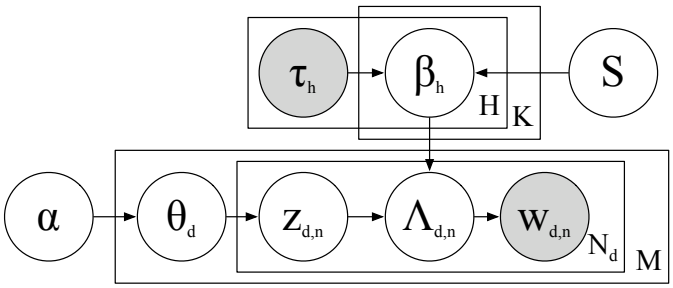


Figure 2.2: The graphical model for LDAWN. Some indices have been suppressed for clarity.

the task of determining the conditional distribution of the hidden variables given the observations. In the case of LDAWN, the hidden variables are the parameters of the K WordNet-Walks, the topic assignments of each word in the collection, and the synset path of each word. In a sense, posterior inference reverses the process described above.

Specifically, given a document collection $\boldsymbol{w}_{1:D}$, the full posterior is

$$\begin{aligned} p(\boldsymbol{\beta}_{1:K,1:H}, \boldsymbol{z}_{1:D}, \boldsymbol{\theta}_{1:D}, \boldsymbol{\Lambda}_{1:D} \mid \boldsymbol{w}_{1:D}, \alpha, S\boldsymbol{\tau}) \\ \propto \left(\prod_k \prod_h p(\boldsymbol{\beta}_{k,h} \mid S\boldsymbol{\tau}_{\boldsymbol{h}}) \right) \\ \left(\prod_{d=1} p(\theta_d \mid \alpha) \prod_{n=1}^{N_d} p(\Lambda_{d,n} \mid \boldsymbol{\beta}_{1:K}) p(w_{d,n} \mid \Lambda_{d,n}) \right), \end{aligned} \tag{2.1}$$

where the constant of proportionality is the marginal likelihood of the observed data.

Note that by encoding the synset paths as a hidden variable, we have posed the WSD problem as a question of posterior probabilistic inference. Further note that we have developed an unsupervised model. No labeled data is needed to disambiguate a corpus. Learning the posterior distribution amounts to simultaneously decomposing a corpus into topics and its words into their synsets.

The intuition behind LDAWN is that the words in a topic will have similar meanings and thus share paths within WordNet. For example, WordNet has two senses

for the word "colt;" one referring to a young male horse and the other to a type of handgun (see Figure 2.1).

Although we have no *a priori* way of knowing which of the two paths to favor for a document, we assume that similar concepts will also appear in the document. Documents with unambiguous nouns such as "six-shooter" and "smoothbore" would make paths that pass through the synset `[firearm, piece, small-arm]` more likely than those going through `[animal, animate being, beast, brute, creature, fauna]`. In practice, we hope to see a WordNet-Walk that looks like Figure 2.4, which points to the right sense of cancer for a medical context.

LDAWN is a Bayesian framework, as each variable has a prior distribution. In particular, the Dirichlet prior for $\boldsymbol{\beta}_{z,h}$, specified by a scaling factor S and a normalized vector $\boldsymbol{\tau}_h$ fulfills two functions. First, as the overall strength of S increases, we place a greater emphasis on the prior. This is equivalent to the need for balancing as noted by Abney and Light (1999).

The other function that the Dirichlet prior serves is to enable us to encode any information we have about how we suspect the transitions to children nodes will be distributed. For instance, we might expect that the words associated with a synset will be produced in a way roughly similar to the token probability in a corpus. For example, even though "meal" might refer to both ground cereals or food eaten at a single sitting and "repast" exclusively to the latter, the synset `[meal, repast, food eaten at a single sitting]` still prefers to transition to "meal" over "repast" given the overall corpus counts (see Figure 2.1, which shows prior transition probabilities for "revolver").

By setting $\tau_{s,i}$, the prior probability of transitioning from synset s to node i, proportional to the total number of observed tokens in the children of i, we introduce a Bayesian variation on information content (Resnik, 1995). As in Resnik's definition, this value for non-word nodes is equal to the sum of all the frequencies of hyponym

words. Unlike Resnik, we do not divide frequency among all senses of a word; each sense of a word contributes its full frequency to τ.

Because we initially thought that path length might bias our selection of paths, we experimented with dividing by the path length during the sampling procedure. Better results were achieved, however, by using the unweighted probability.

2.2 Posterior Inference with Gibbs Sampling

As described above, the problem of WSD corresponds to posterior inference: determining the probability distribution of the hidden variables given observed words and then selecting the synsets of the most likely paths as the correct sense. Directly computing this posterior distribution, however, is not tractable.

To approximate the posterior, we use Gibbs sampling, which has proven to be a successful approximate inference technique for LDA (Griffiths and Steyvers, 2004). In Gibbs sampling, like all Markov chain Monte Carlo methods, we repeatedly sample from a Markov chain whose stationary distribution is the posterior of interest (Robert and Casella, 2004). Even though we don't know the full posterior, the samples can be used to form an empirical estimate of the target distribution (Neal, 1993).

Gibbs sampling reproduces the posterior distribution by repeatedly sampling each hidden variable conditioned on the current state of the other hidden variables and the observations. In LDAWN, for every word in every document we sample the topic assignments $z_{d,n}$ and a path through a topic $\lambda_{d,n}$. For both of these variables, we must compute a conditional distribution over the possible values for these latent variables. In this model, we integrate over θ, the document-specific distribution over topics and β, the transition distribution over children of a synset. [2]

In LDAWN, the state of the chain is given by a set of assignments where each

[2] An alternative sampling scheme would be to also sample $\boldsymbol{\theta}$ and $\boldsymbol{\beta}$. This would yield (slightly) simpler conditional distributions for the topic and paths, but would force us to sample multivariate, continuous variables. In many models, it is simpler to integrate out such variables when possible.

word is assigned to a path through one of K WordNet-Walk topics: the u^{th} word w_u has a topic assignment z_u and a path assignment Λ_u. We use $\boldsymbol{z}^{-u}$ and $\boldsymbol{\Lambda}^{-u}$ to represent the topic and path assignments of all words except for u, respectively.

We sample the topic assignment and path jointly, conditioned on the assignments of all other words. Compared to sampling the path and topic separately, this approach is faster to converge, is similar to Gibbs sampling techniques for LDA, is easier to implement, and is intuitively appealing. For instance, suppose our model were trying to understand what a "colt" was. Not sampling the path simultaneously would force the word to keep a gun interpretation in every putative topic; moving the word into a bucolic topic with a horse interpretation would require either forcing the word to take a gun meaning in a bucolic topic or a horse meaning in a gunslinger topic.

The conditional distribution of a topic assignment and path is

$$\begin{aligned} p(z_{d,u} = y, \Lambda_{d,u} = \pi \mid \boldsymbol{z}_d^{-u}, \boldsymbol{\Lambda}^{-u}, S, \boldsymbol{\tau}, \boldsymbol{\alpha}) \\ = \underbrace{p(\Lambda_{d,u} = \pi \mid z_{d,u} = y, \boldsymbol{\Lambda}^{-u}, S, \boldsymbol{\tau})}_{\text{path}} \underbrace{p(z_{d,n} = y \mid \boldsymbol{z}_d^{-u}, \boldsymbol{\alpha})}_{\text{topic}}. \end{aligned} \tag{2.2}$$

To expand these conditional terms, it is convenient to have terms that count the number words that take each topic in a document and the number of transitions with in a topic's WordNet-Walk . We use $t^d_{-u,j}$ to denote the number of words assigned topic j—in the document d that word u is in—and $b^k_{-u,p,c}$ to denote the number of transitions in the topic walk for the k^{th} topic that go from a synset p to its child c. Both counts exclude counts associated with word u.

First, we consider the topic term of Equation 2.2:

$$p(z_{d,u} = y \mid \boldsymbol{z}_d^{-u}, \boldsymbol{\alpha}) = \frac{p(\boldsymbol{z}_d^{u=y} \mid \boldsymbol{\alpha})}{p(\boldsymbol{z}_d^{-u} \mid \alpha)} = \frac{\int_\theta p(\boldsymbol{z}_{u=y}^d \mid \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) d\boldsymbol{\theta}}{\int_\theta p(\boldsymbol{z}_d^{-u} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) d\theta}$$

the posterior has Dirichlet distribution form (as in Equation 1.1)

$$= \frac{\int_\theta \left(\prod_{k \neq y} \theta_k^{t_k + \alpha_k - 1} \right) \theta_y^{t_y + \alpha_y} d\theta}{\int_\theta \prod_k \theta_k^{t_k + \alpha - 1} d\theta}$$

these integrals are the inverse of Dirichlet normalizer (Equation 1.1)

$$= \frac{\Gamma(t_y + \alpha_y + 1)\, \Gamma\left(\sum_k t_k + \alpha_k\right)}{\Gamma(t_y + \alpha_y)\, \Gamma\left(\sum_k t_k + \alpha_k + 1\right)}$$

and the gamma function can be reduced using $\Gamma(z+1) = z\Gamma(z)$

$$= \frac{t_y + \alpha_y}{\sum_k (t_k + \alpha_k)}. \tag{2.3}$$

Similarly, the path term of Equation 2.2 is expanded using the same techniques but taking the product over all of the transitions in the path π. The final expansion,

$$p(\Lambda_{d,u} = \pi \mid z_d^u = y, \boldsymbol{\Lambda}^{-u}, S, \boldsymbol{\tau}) = \prod_{\lambda_{i,j} \in \pi} \frac{\int_{\boldsymbol{\beta}_{y,i}} p(\boldsymbol{\Lambda}^{u=\pi} \mid \boldsymbol{\beta}_{y,i}) p(\boldsymbol{\beta}_{y,i} \mid \tau_i) d\beta_{y,i}}{\int_{\beta_{y,i}} p(\boldsymbol{\Lambda}^{-u} \mid \boldsymbol{\beta}_{y,i}) p(\boldsymbol{\beta}_{y,i} \mid \tau_i) d\beta_{y,i}} \mathbb{1}[w_u \in \pi]$$

$$\propto \left(\prod_{\lambda_{i,j} \in \pi} \frac{b_{y,i,j}^{-u} + S_i \tau_{i,j}}{\sum_k b_{y,i,k}^{-u} + S_i} \right) \mathbb{1}[w_u \in \pi], \tag{2.4}$$

only allows paths words that end in the u^{th} word. As mentioned in Section 2.1.1, we parameterize the prior for synset i as a vector τ_i, which sums to one, and a scale parameter S. Putting equations 2.3 and 2.4 together, we have a joint probability of a topic assignment and path assignment conditioned on all the topic and path

assignments of the other words in the corpus

$$p(z_{d,u} = y, \Lambda_{d,u} = \pi \mid \boldsymbol{z}_d^{-u}, \boldsymbol{\Lambda}^{-u}, S, \boldsymbol{\tau}, \boldsymbol{\alpha}) \propto \frac{t_y + \alpha_y}{\sum_k (t_k + \alpha_k)} \left(\prod_{\lambda_{i,j} \in \pi} \frac{b_{y,i,j}^{-u} + S_i \tau_{i,j}}{\sum_k b_{y,i,k}^{-u} + S_i} \right) \mathbb{1}[w_u \in \pi]$$

The Gibbs sampler is essentially a randomized hill climbing algorithm on the posterior likelihood as a function of the configuration of hidden variables. The numerator of Equation 2.1 is proportional to that posterior and thus allows us to track the sampler's progress. We assess convergence to a local mode of the posterior by monitoring this quantity.

2.3 Experiments

In this section, we test the efficacy of using hyponomy links for WSD, describe the properties of the topics induced by running the previously described Gibbs sampling method on corpora, and how these topics improve WSD accuracy.

This setup begs the question of whether the structure of WordNet is actually useful for WSD as a probabilistic hierarchy. To test this, we created partial permutations of WordNet by taking a proportion of the synsets in WordNet and randomly choosing a new parent for the node (this moves the entire subtree rooted at that synset; we disallowed moves that would introduce a cycle).

The accuracy (Figure 2.3) steadily decreases as the fraction of synsets permuted increases. Note, however, that even permuting all synsets in WordNet does not do as poorly as the random baseline. We believe that this is because information is still available inside synsets (the word constituents of synsets are never permuted).

Of the two data sets used during the course of our evaluation, the primary dataset was SemCor (Miller et al., 1993), which is a subset of the Brown corpus with

Figure 2.3: Permuting the structure of WordNet results in decreased disambiguation accuracy, showing that the structure of WordNet is helpful in creating improved disambiguation.

many nouns manually labeled with the correct WordNet sense. The words in this dataset are lemmatized, and multi-word expressions that are present in WordNet are identified. Only the words in SemCor were used in the Gibbs sampling procedure; the synset assignments were only used for assessing the accuracy of the final predictions. We choose SemCor for evaluation because it is labeled; however, additional data may be useful to learn better topics and correlations.

We also used the British National Corpus, which is not lemmatized and which does not have multi-word expressions. The text was first run through a lemmatizer, and then sequences of words which matched a multi-word expression in WordNet were joined together into a single word. We took nouns that appeared in SemCor twice or in the BNC at least 25 times and used the BNC to compute the information-content analog $\boldsymbol{\tau}$ for individual nouns. We split counts across senses, following (Kilgarriff, 2000).

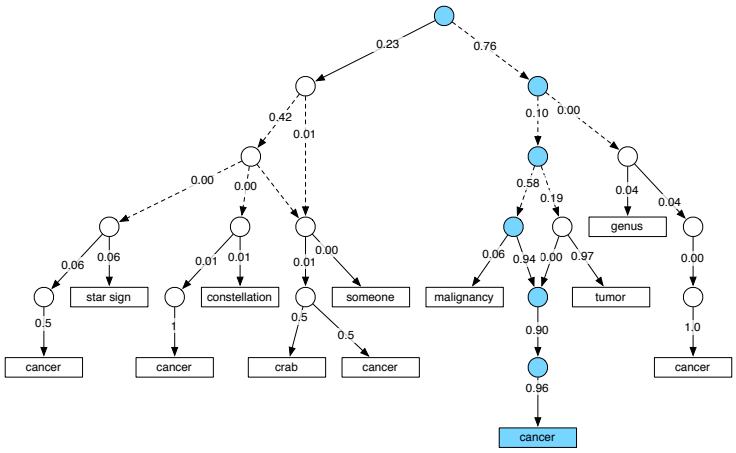


Figure 2.4: The possible paths to reach the word "cancer" in WordNet along with transition probabilities from the medically-themed Topic 2 in Table 2.2, with the most probable path highlighted. The dashed lines represent multiple links that have been consolidated. Some words for immediate hypernyms have also been included to give context. In all other topics, the person, animal, or constellation sense was preferred.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7
president	growth	material	point	water	plant	music
party	age	object	number	house	change	film
city	treatment	color	value	road	month	work
election	feed	form	function	area	worker	life
administration	day	subject	set	city	report	time
official	period	part	square	land	mercer	world
office	head	self	space	home	requirement	group
bill	portion	picture	polynomial	farm	bank	audience
yesterday	length	artist	operator	spring	farmer	play
court	level	art	component	bridge	production	thing
meet	foot	patient	corner	pool	medium	style
police	maturity	communication	direction	site	petitioner	year
service	center	movement	curve	interest	relationship	show

Table 2.2: The most probable words from six randomly chosen WordNet-walks from a thirty-two topic model trained on the words in SemCor. These are summed over all of the possible synsets that generate the words. However, the vast majority of the contributions come from a single synset.

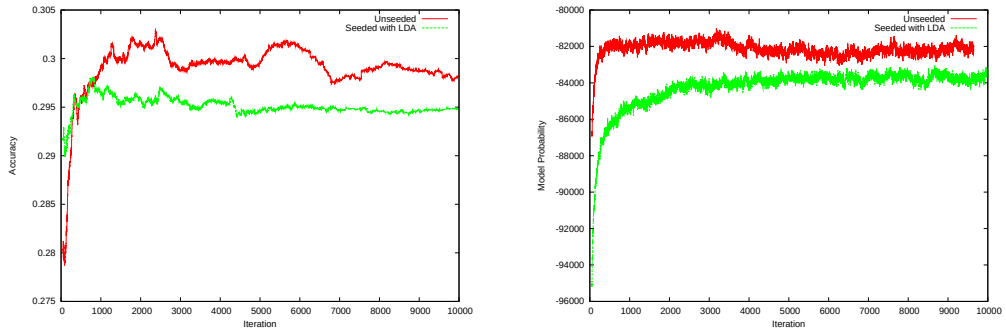
2.3.1 Topics

Like the topics created by structures such as LDA, the topics in Table 2.2 coalesce around reasonable themes. The word list was compiled by summing over all of the possible leaves that could have generated each of the words and sorting the words by decreasing probability. In the vast majority of cases, a single synset's high probability is responsible for the words' positions on the list.

Reassuringly, many of the top senses for the present words correspond to the most frequent sense in SemCor. For example, in Topic 4, the senses for "space" and "function" correspond to the top senses in SemCor, and while the top sense for "set" corresponds to "an abstract collection of numbers or symbols" rather than "a group of the same kind that belong together and are so used," it makes sense given the math-based words in the topic. "Point," however, corresponds to the sense used in the phrase "I got to the point of boiling the water," which is neither the top SemCor sense nor a sense which makes sense given the other words in the topic.

While the topics presented in Table 2.2 resemble the topics one would obtain through models like LDA (Blei et al., 2003), they are not identical. Because of the lengthy process of Gibbs sampling, we initially thought that using LDA assignments as an initial state would converge faster than a random initial assignment. While this was the case, chains initialized with LDA consistently converged to a state that was less probable than the randomly initialized state and did no better at sense disambiguation (and sometimes worse). The topics presented in Table 2.2 represent words both that co-occur together in a corpus and co-occur on paths through WordNet. Because topics created through LDA only have the first property, they usually do worse in terms of both total probability and disambiguation accuracy (see Figure 2.5).

Another interesting property of topics in LDAWN is that, with higher levels of smoothing, words that don't appear in a corpus (or appear rarely) but are in similar parts of WordNet might have relatively high probability in a topic. For example,



(a) Accuracy (b) Log Probability

Figure 2.5: Topics seeded with LDA initially have a higher disambiguation accuracy, but are quickly matched by unseeded topics. The probability for the seeded topics starts lower and remains lower.

"maturity" in topic two in Table 2.2 is sandwiched between "foot" and "center," both of which occur about five times more than "maturity."

2.3.2 Topics and the Weight of the Prior

Because the Dirichlet smoothing factor in part determines the topics, it also affects the disambiguation. Figure 2.6 shows the modal disambiguation achieved for each of the settings of $S = \{0.1, 1, 5, 10, 15, 20\}$. Each line is one setting of K and each point on the line is a setting of S. Each data point is a run for the Gibbs sampler for 10,000 iterations. The disambiguation, taken at the mode, improved with moderate settings of S, which suggests that the data are still sparse for many of the walks, although the improvement vanishes if S is very large. This makes sense, as each walk has over 100,000 parameters, there are fewer than 100,000 words in SemCor, and each word only serves as evidence to at most 19 parameters (the length of the longest path in WordNet).

Generally, a greater number of topics increased the accuracy of the mode, but after around sixteen topics, gains became much smaller. The effect of $\boldsymbol{\tau}$ is also related to the

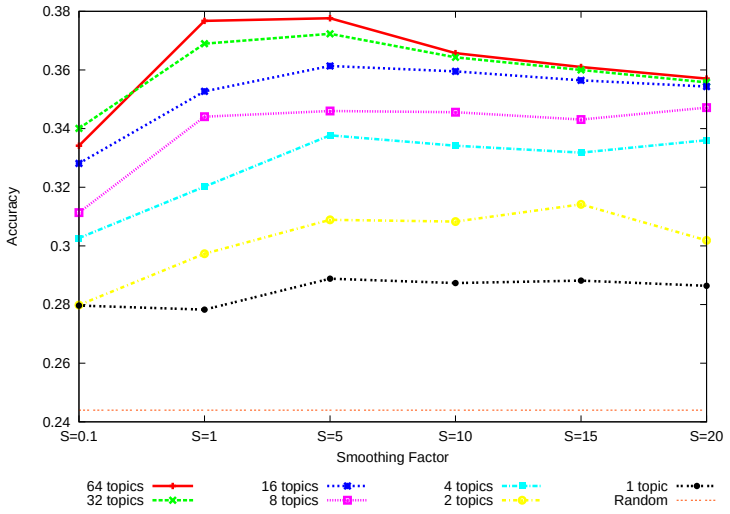


Figure 2.6: Each line represents experiments with a set number of topics and variable amounts of smoothing on the SemCor corpus. The random baseline is at the bottom of the graph, and adding topics improves accuracy. As smoothing increases, the prior (based on token frequency) becomes stronger. Accuracy is the percentage of correctly disambiguated polysemous words in SemCor at the mode.

number of topics, as a value of S for a very large number of topics might overwhelm the observed data, while the same value of S might be the perfect balance for a smaller number of topics. For comparison, the method of using a WordNet-Walk applied to smaller contexts such as sentences or documents achieves an accuracy of between 26% and 30%, depending on the level of smoothing.

2.3.3 Evaluation on Senseval

Using the best model as evaluated on SemCor, a model with 32 topics and $S = 1$, we applied the model to Senseval2 and Senseval3 English all-words task. The model gave an accuracy of 40.5% and 30.3%, respectively, outperforming some of the domain-aware algorithms in the Senseval3 contest evaluation (Mihalcea et al., 2004). While other systems performed better; they were supervised or used the predominant sysnsets

approach (McCarthy et al., 2004), which can be combined with topic modeling (Boyd-Graber and Blei, 2007).

2.4 Error Analysis

This method works well in cases where the delineation can be readily determined from the overall topic of the document. Words such as "kid," "may," "shear," "coach," "incident," "fence," "bee," and (previously used as an example) "colt" were all perfectly disambiguated by this method. Figure 2.4 shows the WordNet-Walk corresponding to a medical topic that correctly disambiguates "cancer."

Problems arose, however, with highly frequent words, such as "man" and "time" that have many senses and can occur in many types of documents. For example, "man" can be associated with many possible meanings,

1. man, adult male
2. serviceman, military man, man, military personnel
3. homo, man, human being, human
4. man (male subordinate)
5. man (virile and courageous person, worthy of respect)
6. man (husband or lover or boyfriend)
7. valet, valet de chambre, gentleman, gentleman's gentleman, man
8. Man, Isle of Man
9. man, piece
10. world, human race, humanity, humankind, human beings, humans, mankind, man

Although we know that the "adult male" sense should be preferred, the alternative meanings will also be likely if they can be assigned to a topic that shares common paths in WordNet; the documents contain, however, many other places, jobs, and animals which are reasonable explanations (to LDAWN) of how "man" was generated.

Unfortunately, "man" is such a ubiquitous term that topics, which are derived from the frequency of words within an entire document, are ultimately uninformative about its usage.

Even changing the size of the document would not help us disambiguate "man," however, as we would have no reason to suspect that we would then see "man" occurring more frequently with sibling terms like "chap," "fellow," "sirrah," or "boy wonder." This reveals that our underlying disambiguation method requires significant co-occurrence of words from the same semantic class. In order to successfully disambiguate words like "man," our method would have to be aware of syntagmatic relationships.

While mistakes on these highly frequent terms significantly hurt our accuracy, errors associated with less frequent terms reveal that WordNet's structure is not easily transformed into a probabilistic graph. For instance, there are two senses of the word "quarterback," a player in American football. One is position itself and the other is a person playing that position. While one would expect co-occurrence in sentences such as "quarterback is a well-paid position and is protected by burly linemen, so our quarterback is happy," the paths to both terms share only the root node, thus making it highly unlikely a topic would cover both senses.

Although WordNet is often criticized for its breadth and fine-grained senses, and this too impacts our accuracy, an extraneous sense causes us no problem so long as it is tucked away in an unvisited corner of WordNet. However, rare senses do present problems when they are placed next to more frequent terms in WordNet.

Because of WordNet's breadth, rare senses also impact disambiguation. For example, the metonymical use of "door" to represent a whole building as in the phrase "girl next door" is under the same parent as sixty other synsets containing "bridge," "balcony," "body," "arch," "floor," and "corner." Surrounded by such common terms that are also likely to co-occur with the more conventional meanings of door, this very

rare sense becomes the preferred disambiguation of "door."

2.5 Related Work

Abney and Light's initial probabilistic WSD approach (1999) was further developed into a Bayesian network model by Ciaramita and Johnson (2000), who likewise used the appearance of unambiguous terms close to ambiguous ones to "explain away" the usage of ambiguous terms in selectional restrictions. We have adapted these approaches and put them into the context of a topic model.

Recently, other approaches have created *ad hoc* connections between synsets in WordNet and then considered walks through the newly created graph. Given the difficulties of using existing connections in WordNet, Mihalcea (2005) proposed creating links between adjacent synsets that might comprise a sentence, initially setting weights to be equal to the Lesk overlap between the pairs, and then using the PageRank algorithm to determine the stationary distribution over synsets.

2.5.1 Topics and Domains

Yarowsky was one of the first to contend that "there is one sense for discourse" (1995). This has lead to the approaches like that of Magnini (Magnini et al., 2001) which assign one of a fixed set of categories to a text and then deterministically use the domain annotation attached to WordNet to assign a single synset.

LDAWN is different in that the categories are not an *a priori* concept that must be painstakingly annotated within WordNet and require no augmentation of WordNet. This technique could indeed be used with any hierarchy. The concepts discovered by our model are the ones that, via the assumed generative process, best describe the observed documents and hierarchy.

Recently, the use of unsupervised topics have gained popularity as a means for

improving WSD. Cai et al. (2007) used topic distributions as a feature in a standard discriminative WSD algorithm. This means that the flow of information is entirely one way; while topics can influence the sense distinction, the topics remain static. Other work (Brody and Lapata, 2009) has used tools from topic modeling on smaller contexts (at most a dozen words) to induce word senses (rather than using a precompiled dictionary).

2.5.2 Similarity Measures

Our approach gives a probabilistic method of using information content (Resnik, 1995) as a starting point that can be adjusted to cluster words in a given topic together; this is similar to the Jiang-Conrath similarity measure (1997), which has been used in many applications in addition to disambiguation. Patwardhan (2003) offers a broad evaluation of similarity measures for WSD.

Our technique for combining the cues of topics and distance in WordNet is adjusted in a way similar in spirit to Buitelaar and Sacaleanu (2001), but we consider the appearance of a single term to be evidence for not just that sense and its immediate neighbors in the hyponomy tree but for all of the sense's children and ancestors.

Like McCarthy (2004), our unsupervised system acquires a single predominant sense for a domain based on a synthesis of information derived from a textual corpus, topics, and WordNet-derived similarity, a probabilistic information content measure. By adding syntactic information from a thesaurus derived from syntactic features (taken from Lin's automatically generated thesaurus (1998)), McCarthy achieved 48% accuracy in a similar evaluation on SemCor; LDAWN is thus substantially less effective in disambiguation compared to state-of-the-art methods (c.f. results in Figure 2.6). This suggests, however, that other methods might be improved by adding topics and that our method might be improved by using more information than word counts.

2.6 Extensions

The model presented here serves as a bridge between the work in both topic modeling and WSD. From the perspective of the topic modeling community, the model here demonstrates a means for explicitly adding a notion of semantic coherence to the discovered topics and encouraging correlation between vocabulary terms in topics. From the perspective of the WSD community, this model demonstrates that, at least for a simple WSD model, introducing automatically discovered topics can improve accuracy.

In the next chapter, we further explore another consequence of this model. Because this model separates meaning (the structure of the ontology) from lexicalization (the leaves in the ontology), it enables us to create topic models for multilingual corpora.

Chapter 3

Bridging the Gap Between Languages

Latent Dirichlet Allocation (LDA) is a technique of discovering coherent topics in the documents in a corpus. LDA can capture coherence in a single language because semantically similar words tend to be used in similar contexts. This is not the case in multilingual corpora. For example, even though "Hund" and "hound" are orthographically similar and have nearly identical meanings in German and English (i.e., "dog"), they will likely not appear in similar contexts because almost all documents are written in a single language. Consequently, a topic model fit on a bilingual corpus reveals coherent topics but bifurcates the topic space between the two languages (Table 3.1). In order to build coherent topics across languages, there must be some connection to tie the languages together.

A topic model on unaligned text in multiple languages would allow the many applications developed for monolingual topic models (for an overview, see Section 1.1.5) to be applied to a broader class of corpora and would help monolingual users to explore and understand multilingual corpora. In this chapter, we develop *multilingual* LDAWN, an extension of the model presented in the previous chapter. This model discovers

topics that are consistent across multiple languages.

Previous multilingual topic models connect the languages by assuming parallelism at either the sentence level (Zhao and Xing, 2006) or document level (Kim and Khudanpur, 2004; Tam and Schultz, 2007; Ni et al., 2009; Mimno et al., 2009). Many parallel corpora are available, but they represent a small fraction of corpora. They also tend to be relatively well annotated and understood, making them less suited for unsupervised methods like LDA.

In contrast, we connect the languages by assuming a shared semantic space. In Chapter 2 we created an explicit semantic space for English. However, the semantic space created is not English specific. Since WordNet was first created, a number of other languages have used WordNet's internal structure as a guide to attach other languages to WordNet internal structure. After reviewing what a multilingual WordNet looks like, we expand the model from the previous chapter to accommodate multiple languages and evaluate the model on a WSD task and a task inspired by information retrieval.

Topic 0	Topic 1	Topic 2	Topic 3
market	group	bericht	praesident
policy	vote	fraktion	menschenrecht
service	member	abstimmung	jahr
sector	committee	kollege	regierung
competition	report	ausschuss	parlament
system	matter	frage	mensch
employment	debate	antrag	hilfe
company	time	punkt	volk
union	resolution	abgeordnete	region

Table 3.1: Four topics from a ten topic LDA model run on the German and English sections of Europarl. Without any connection between the two languages, the topics learned are language-specific.

3.1 Assembling a Multilingual Semantic Hierarchy

Linguists have also created WordNet- like networks for languages besides English. In this work, we focus on German, one of the first successors to the English WordNet (Hamp and Feldweg, 1997). However, we stress that there is nothing specific in our approach to German; this approach is applicable to any languages that have been been organized in a manner similar to WordNet's hyponomy relationship.[1]

Many concepts are shared across languages. For instance, the German synset `[Umleitung, Umgehung]` is equivalent to the English synset `[detour, roundabout_way]`. Some concepts are lexicalized in one language but not the other. For instance, the German synset `[Beinbruch]`, a leg fracture, doesn't have a fixed expression in English, but it still can be considered a hyponym of the English synset `[break, fracture]`.

The different level of equivalences are formalized in the interlingual index (ILI), which is used by the EuroWordNet project (Vossen, 1998). For German (Kunze and Lemnitzer, 2002), it explicitly links synsets across languages; in this work, we focus on the "synonym," "near synonym," and "hypernym" relationships. If an English synset is listed as a hypernym, synonym, or near synonym of a German synset, the German words in that synset are added to the English synset.

Because this WordNet, by construction, is much more complete for English than German, we attempted to reduce this imbalance using the following methods:

Balanced Only retain English words in a synset if there were also German words.

Dictionary Using a dictionary (Richter, 2008), if there is an English word with an unambiguous translation (the entry for the English word only points to one German word, and the entry for the German word only points to the same English word), add that German word to the synset.

Table 3.2 shows the relative sizes of the resulting multilingual WordNet when created

[1]For a review of this process for European languages, see (Vossen, 1998).

	Raw		Bal		Raw + Dict		Bal + Dict	
	Eng	Ger	Eng	Ger	Eng	Ger	Eng	Ger
Words	146347	13970	23481	13970	146347	17795	29942	17795
Synsets	82115	11198	11198	11198	82115	14524	14524	14524
Words per synset	1.78	1.25	2.10	1.25	1.78	1.23	2.06	1.23

Table 3.2: The size and coverage of a combined German-English WordNet. "Raw" includes all words in both languages, "Balanced" only includes synsets that have a representative from German, and "Dictionary" uses a dictionary to add translations to unambiguous words.

with these strategies.

Initial experiments showed that ensuring a balance between each WordNet was critical and that including more words leads to more comprehensive coverage of vocabulary (especially for technical terms), so for the rest of the experiments in this chapter, we focus on using the multilingual WordNet with balanced words expanded by using the dictionary.

With this new WordNet that has multiple means of expressing many synsets, we now need to adjust our generative model to account for multiple languages.

3.2 A Language Agnostic Generative Process

In Section 2.1, we described a monolingual process for drawing from a distribution over words specified by WordNet. In this section, we extend this process to handle a WordNet with multiple languages.

It is helpful to separate the generative process for producing a word in language l in topic walk k into two stages: choosing a synset and choosing a word given that synset. Only the second step is language dependent; the first step can be the same for all languages. The generative process we outline below is different from the generative process in Section 2.1 in that we explicitly differentiate between synsets and words in the generative process. In the monolingual case in Chapter 2, topology was sufficient to differentiate words from synsets (in that model, by construction, all leaf nodes were

words).

We must choose both a synset and a word. When we reach a synset, we flip a biased coin to choose whether we continue to another synset or we stop at the synset and emit a word. The stopping probability is $\omega_{k,h}$, specific to each topic and synset (but shared for all languages). If we choose not to stop, we continue on to one of the children of the current synset. The multinomial distribution $\beta_{k,h}$ gives a probability over each of the node's children.

If we did choose to stop and emit a word rather than continuing to a child, we now must emit a word from synset s. Here, it becomes important that we already know the language of the document, l_d. While each synset in each topic has a language-specific distribution over the words, we only use the distribution over words that is consistent with the document's language, ϕ_{k,s,l_d} to choose the word.

3.2.1 Multilingual Priors

As in the case of a monolingual walk over the concept hierarchy, we want to provide guidance to the multilingual model as to which parts of the hierarchy it should favor. We do this by defining a prior distributions over the variables ω, β, and ϕ discussed above. This is more difficult than with LDAWN because we are dealing with multiple languages; we shouldn't allow one language to overwhelm the other because of quirks of the corpus or WordNet.

To define the priors τ, α, and η, we will define frequency counts, specific to to a word and language, and hyponym counts, specific to synsets but independent of language.

As before, we take our frequency counts from a balanced corpus and derived a count $f_l(w)$ for every token w in language l. We now create a normalizer F_l for each

language,

$$F_l = \sum_{w \in WN_l} f_l(w),$$

where WN_l represents the set of synsets in language l. We are not summing over all of the tokens in the language, only the tokens which appear in the language's WordNet. This prevents the coverage of a WordNet from affecting the relative strength of a language. which gives us a raw normalized count for each token w,

$$\hat{f}_l(w) = \frac{f_l(w)}{F_l}. \tag{3.1}$$

These counts are at the token level, but we need counts at the synset level. We define $\text{hypo}_l(s)$ to be the multiset all of the tokens in language l that are a part of a synset that is a hyponym of synset s, and we define $s[l]$ to be all of the words in language l that are in synset s. This allows us to define a hyponym count for a synset s

$$h(s) = \underbrace{\sum_l \sum_{w \in s[l]} \hat{f}_l(w)}_{\text{words in synset}} + \underbrace{\sum_{t \in \text{hypo}_l(s)} \hat{f}_l(t)}_{\text{descendant synsets}} . \tag{3.2}$$

Note that the synset counts are independent of language and reflect the preferences of all languages.

We now have all of the counts needed to define our priors: the prior for transitions (synset to synset), the prior for emission (synset to word), and the prior for stopping (moving from choosing synset to choosing a word).

Transition The transition prior for transitioning from synset s to synset t is independent of language, is proportional to the hyponym count of each synset, and is defined as

$$\beta_{s,t} \equiv \frac{h(t)}{\sum_{v \in c(s)} h(v)}, \tag{3.3}$$

where $c(s)$ is the set of all the direct hyponyms of s.

Emission The emission prior is language dependent. For a synset s, its emission probability over its words in language l is proportional to the normalized frequency.

$$\eta_{l,w} \equiv \frac{\hat{f}_l(w)}{\sum_{u \in s[l]}}, \tag{3.4}$$

where $s[l]$ is the set of all the words in synset s from language l.

Stop The prior probability for whether to stop at a node and emit a word or continue by transitioning to a child combines the counts used in the previous two definitions. The probability of stopping is proportional to the total counts of all tokens in the synset in all languages, and the probability of continuing is proportional to the total of all hyponym counts. Thus, the prior probability of stopping is

$$\sigma_s \equiv \frac{\sum_l \sum_{w \in s[l]} f_l(w)}{\sum_l \sum_{w \in s[l]} f_l(w) + \sum_{t \in c(s)} h(t)}. \tag{3.5}$$

For these experiments, we used the British National Corpus (BNC) (University of Oxford, 2006) as our English balanced corpus and the Digitales Wörterbuch der deutschen Sprache (DWDS) (Geyken, 2007)

3.2.2 Specifying the Generative Process

We introduced the new aspects of this model in the previous section: the language of documents, a per-language distribution over words for each synset and a separate stopping probability for each synset (these are depicted as a generative model in Figure 3.1). The generative process for multilingual LDAWN is as follows (differences from LDAWN are highlighted in *i*talics):

1. For each topic, $k \in \{1, \ldots, K\}$

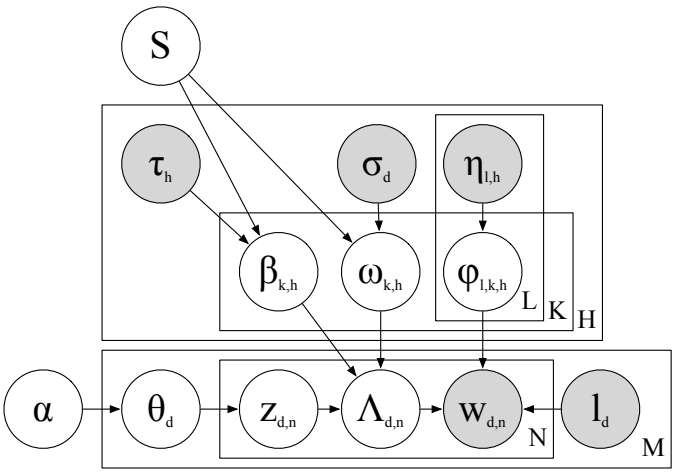


Figure 3.1: The graphical model for ML-LDAWN. The bottom plate is similar to LDA, but with the addition of a known language for each document. The top plate replaces the multinomial distribution of LDA with a hierarchical distribution over words from an ontology with known information content (the shaded hyperparameters).

(a) For each synset $h \in \{1, \ldots, H\}$ in the hierarchy

i. Choose transition probabilities $\boldsymbol{\beta}_{k,h} \sim \text{Dir}(S\tau_h)$.

ii. *C*hoose stopping probabilities $\omega_{k,h} \sim \text{Beta}(S\sigma, S(1-\sigma),)$.

iii. *F*or each language l, choose emission probabilities $\boldsymbol{\phi}_{k,h,l} \sim \text{Dir}(\eta_{h,l})$.

2. For each document $d \in \{1, \ldots, D\}$ *w*ith language l

(a) Select a topic distribution $\boldsymbol{\theta}_d \sim \text{Dir}(\tau)$

(b) For each word $n \in \{1, \ldots, N_d\}$

i. Select a topic $z_{d,n} \sim \text{Mult}(\boldsymbol{\theta}_d)$

ii. Create a path $\Lambda_{d,n}$ starting with the root node h_0.

iii. Given the current node h:

A. *C*hoose whether to emit a word with probability ω_h.

B. *I*f we chose to emit a word, choose $w_n \sim \text{Mult}(\boldsymbol{\phi}_{z_{d,n},h,l})$.

C. Otherwise, choose the next node in the walk $h' \sim \text{Mult}(\boldsymbol{\beta}_{z_{d,n},h})$; add the step $\lambda_{h,h'}$ to the path $\Lambda_{d,n}$. Repeat with h' as the current node.

3.3 Inference

We use posterior inference to discover the transition probabilities, stopping probabilities, and emission probabilities of the multilingual distributions over the synsets and words in WordNet in addition to the per-document topic distributions and topic assignments of each word. We are given the language and words in each document and the information content information as described in section 3.2.1.

Inference in ML-LDAWN proceeds in much the same way as for LDAWN. The probability of a word in position u having topic y and path π in document d with language l given the transition, stopping, and emission distributions for the multilingual WordNet walks is

$$p(z_{d,u} = y, \Lambda_{d,u} = \pi \,|\, \boldsymbol{\theta}_d, \boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\phi}) = \tag{3.6}$$

$$\underbrace{\boldsymbol{\theta}_{d,k}}_{\text{topic}} \underbrace{\prod_{(i,j)\in\lambda} [(1-\omega_i)\beta_{\lambda,i,j}]}_{\text{continue, pick child}} \underbrace{\omega_{\lambda_{end}} \boldsymbol{\phi}_{\lambda_{end},l_d,w_u}}_{\text{stop, emit word}} . \tag{3.7}$$

As in Section 2.2, we use Gibbs sampling to sample the current topic and path of each word, conditioning on the topic and paths of all of the other words.

As before, we integrate out all of the multinomial random variables to obtain the conditional distribution

$$\begin{aligned}
&p(z_{d,u} = y, \Lambda_{d,u} = \pi \,|\, \boldsymbol{z}_d^{-u}, \boldsymbol{\Lambda}^{-u}, S, \boldsymbol{\tau}, \boldsymbol{\alpha}, \boldsymbol{\sigma}) \\
&\quad = \int_\beta \int_\omega \int_\phi \int_{\theta_d} p(z_{d,u} = y, \Lambda_{d,u} \,|\, \theta_d, \boldsymbol{\beta}, \boldsymbol{\omega}, \boldsymbol{\phi}, \boldsymbol{z}^{-u}, \boldsymbol{\Lambda}^{-u}) \\
&\quad \propto \underbrace{\frac{t_{d,y} + \alpha_y}{t_{d,\cdot} + \sum_k \alpha_k}}_{\text{topic}} \underbrace{\prod_{(i,j)\in\pi} \left[\left(\frac{o_{k,i,\top} + S\sigma}{c_{k,i,\cdot} + S} \right) \left(\frac{b_{y,i,j} + S\tau_{i,j}}{b_{k,i,\cdot} + S} \right) \right]}_{\text{continue, pick child}} \\
&\quad \underbrace{\left(\frac{o_{\pi_{end},\perp} + S(1 - \sigma_{\pi_{end}})}{o_{\pi_{end},\cdot} + S} \right) \left(\frac{f_{l_d,\pi_{end},w_u} + \eta_{l_d,\pi_{end},w_u}}{f_{l_d,\pi_{end},\cdot} + \sum_k \eta_{l_d,\pi_{end},k}} \right)}_{\text{stop, emit word}} .
\end{aligned} \tag{3.8}$$

We use the convention of using $\cdot$ to represent marginal counts, i.e. $t_{d,\cdot} \equiv \sum_{i'} t_{d,i'}$. There are three components to this conditional distribution. The first term contributes to the probability of a topic, which is the same as LDA. The next term (the product over edges i, j) in the path chosen. The final term contributes to the probability the probability of the path emitting the observed word.

3.4 Experiments

In this section we show evidence that this model can discover consistent topics on multilingual datasets. We use a dataset called the Europarl corpus (Koehn, 2005), which is a collection of the proceedings of the European parliament translated into many of the languages of the countries in the European Union. We use the English and German versions and create documents based on the chapter breaks in the proceedings (a chapter is debate / discussion on a single topic). Note that even though these are parallel texts, we ignore the parallel component of these data (in the next chapter, we show how unaligned algorithms like these can recover the connections between documents).

3.5 Discussion

ML-LDAWN relies on having complete and accurate translations of WordNet in multiple languages. German has one of the more complete WordNet mappings, but is still relatively incomplete. Most other languages are even more sparse, and the model has no way to overcome a lack of connections, even if there are strong cues from the data that concepts **should** be related. Even worse, if a language lacks a WordNet, we cannot apply the method of this chapter at all.

In the next chapter, we propose a method that overcomes this limitation by learning and adapting a mapping across languages as it learns multilingual topics. It does so

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
president	member	*bericht*	*volk*	*kommission*
praesident	*mitglied*	mr	group	commission
gentleman	community	report	*gruppe*	*union*
herr	*menge*	fact	people	union
land	measure	*tatsache*	member	council
country	*markt*	amendment	*mitglied*	*rat*
european	market	*direktive*	*menschheit*	house
woman	*volk*	directive	world	parlament
frau	people	*bedingung*	place	year
gebiet	relation	member	time	*jahr*
area	*beziehung*	*mitglied*	item	*frage*
europaeer	region	agreement	country	proposal
development	level	*thema*	conflict	question
entwicklung	fund	*information*	*widerstreit*	parliament
citizen	*schritt*	information	man	*bundestag*

Table 3.3: Topics discovered by a MLDA on the Europarl dataset. Observe that topics have consistency across languages. For instance, in topic 4, "menschheit" (humanity), "widerstreit" (conflict), and "mitglied" (member) are clustered together with the English words "people," "conflict," and "member."

in a way that can adapt to relatively small amounts of interlingual annotation, in contrast to the method outlined here.

Chapter 4

Learning a Shared Semantic Space

In the previous chapter, we developed a multilingual topic model that required WordNets to be available in the languages of interest. In this chapter, we develop a model that does not require such a labor-intensive investment to create a topic model that is consistent across languages. Like the model of the previous chapter, we do not assume that we have text that is parallel at the sentence or paragraph level.

We propose the MUltilingual TOpic model for unaligned text (MuTo). MuTo does not assume that it is given any explicit parallelism but instead discovers a parallelism at the vocabulary level. To find this parallelism, the model assumes that similar themes and ideas appear in both languages. For example, if the word "Hund" appears in the German side of the corpus, "hound" or "dog" should appear somewhere on the English side.

4.1 Learning Dictionaries with Text Alone

The assumption that similar terms will appear in similar contexts has also been used to build lexicons from non-parallel but comparable corpora. What makes contexts similar can be evaluated through such measures as co-occurrence (Rapp, 1995; Tanaka and Iwasaki, 1996) or tf-idf (Fung and Yee, 1998). Although the emphasis of our work

is on building consistent topic spaces and not the task of building dictionaries *per se*, good translations are required to find consistent topics. However, we can build on successful techniques at building lexicons across languages.

This paper is organized as follows. We detail the model and its assumptions in Section 4.2, develop a stochastic expectation maximization (EM) inference procedure in Section 4.3, discuss the corpora and other linguistic resources necessary to evaluate the model in Section 4.4, and evaluate the performance of the model in Section 4.5.

4.2 Model

We assume that, given a bilingual corpus, similar themes will be expressed in both languages. If "dog," "bark," "hound," and "leash" are associated with a pet-related topic in English, we can find a set of pet-related words in German without having translated all the terms. If we can guess or we are told that "Hund" corresponds to one of these words, we can discover that words like "Leinen," "Halsband," and "Bellen" ("leash," "collar," and "bark," respectively) also appear with "Hund" in German, making it reasonable to guess that these words are part of the pet topic as expressed in German.

These steps—learning which words comprise topics within a language and learning word translations across languages—are both part of our model. In this section, we describe MuTo's generative model, first describing how a matching connects vocabulary terms across languages and then describing the process for using those matchings to create a multilingual topic model.

4.2.1 Matching across Vocabularies

We posit the following generative process to produce a bilingual corpus in a source language S and a target language T. First, we select a matching $\boldsymbol{m}$ over terms in both

languages. The matching consists of edges (v_i, v_j) linking a term v_i in the vocabulary of the first language V_S to a term v_j in the vocabulary of the second language V_T. A matching can be viewed as a bipartite graph with the words in one language V_S on one side and V_T on the other. A word is either unpaired or linked to a single node in the opposite language.

The use of a matching as a latent parameter is inspired by the matching canonical correlation analysis (MCCA) model (Haghighi et al., 2008), another method that induces a dictionary from arbitrary text. MCCA uses a matching to tie together words with similar meanings (where similarity is based on feature vectors representing context and morphology). We have a slightly looser assumption; we only require words with similar document level contexts to be matched. Another distinction is that instead of assuming a uniform prior over matchings, as in MCCA, we consider the matching to have a regularization term $\pi_{i,j}$ for each edge from source word v_i to target word v_j. We prefer larger values of $\pi_{i,j}$ in the matching.

This parameterization allows us to incorporate prior knowledge derived from morphological features, existing dictionaries, or dictionaries induced from non-parallel text. We can also use the knowledge gleaned from parallel corpora to understand the non-parallel corpus of interest. Sources for the matching prior π are discussed in Section 4.4.

4.2.2 From Matchings to Topics

In MuTo, documents are generated conditioned on the matching. As in LDA, documents are endowed with a distribution over topics. Instead of being distributions over terms, topics in MuTo are distributions over pairs in the matching $\boldsymbol{m}$. Going back to our intuition, one such pair might be ("hund", "hound"), and it might have high probability in a pet-related topic. Another difference from LDA is that unmatched terms don't come from a topic but instead come from a unigram distribution specific to

each language. The full generative process of the matching and both corpora follows:

1. Choose a matching $\boldsymbol{m}$ across languages where the probability of an edge $m_{i,j}$ being included is proportional to $\boldsymbol{\pi}_{i,j}$
2. Choose multinomial term distributions:
   (a) For languages $L \in \{S, T\}$, choose background distributions $\boldsymbol{\rho}_L \sim \text{Dir}(\gamma)$ over the words not in $\boldsymbol{m}$.
   (b) For topic index $i = \{1, \ldots, K\}$, choose topic $\boldsymbol{\beta}_i \sim \text{Dir}(\lambda)$ over the pairs (v_S, v_T) in $\boldsymbol{m}$.
3. For each document $d = \{1, \ldots D\}$ with language l_d:
   (a) Choose topic weights $\boldsymbol{\theta}_d \sim \text{Dir}(\alpha)$.
   (b) For each $n = \{1, \ldots, M_d\}$:
      i. Choose topic assignment $z_n \sim \text{Mult}(1, \boldsymbol{\theta}_d)$.
      ii. Choose c_n from $\{\text{matched}, \text{unmatched}\}$ uniformly at random.
      iii. If c_n matched, choose a pair $\sim \text{Mult}(1, \beta_{z_n}(\boldsymbol{m}))$ and select the member of the pair consistent with l_d, the language of the document, for w_n.
      iv. If c_n is unmatched, choose $w_n \sim \text{Mult}(1, \boldsymbol{\rho}_{l_d})$.

Both ρ and β are distributions over words. The background distribution ρ_S is a distribution over the $(|V_S| - |\boldsymbol{m}|)$ words not in $\boldsymbol{m}$, ρ_T similarly for the other language, and β is a distribution over the word pairs in $\boldsymbol{m}$. Because a term is either part of a matching or not, these distributions partition the vocabulary.

The background distribution is the same for all documents. We choose not to have topic-specific distributions over unmatched words for two reasons. The first reason is to prevent topics from having divergent themes in different languages. For example, even if a topic had the matched pair ("Turkei", "Turkey"), distinct language topic multinomials over words could have "Istanbul," "Atatürk," and "NATO" in German but "stuffing," "gravy," and "cranberry" in English. The second reason is to encourage

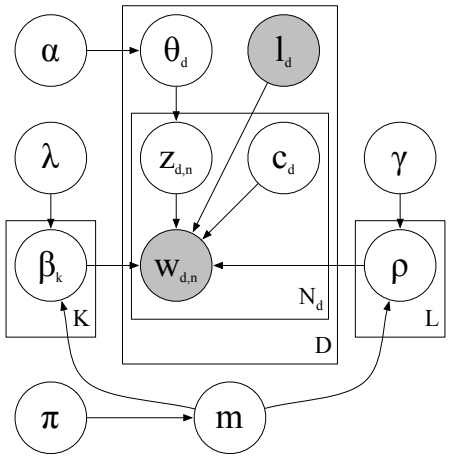


Figure 4.1: Graphical model for MuTo. The matching over vocabulary terms $\boldsymbol{m}$ determines whether an observed word w_n is drawn from a topic-specific distribution β over matched pairs or from a language-specific background distribution ρ over terms in a language.

very frequent nouns that can be well explained by a language-specific distribution (and thus likely not to be topical) to remain unmatched.

4.3 Inference

Given two corpora, our goal is to infer the matching $\boldsymbol{m}$, topics β, per-document topic distributions $\boldsymbol{\theta}$, and topic assignments z. We solve this posterior inference problem with a stochastic EM algorithm (Diebolt and Ip, 1996). There are two components of our inference procedure: finding the maximum a posteriori matching and sampling topic assignments given the matching.

We first discuss estimating the latent topic space given the matching. We use a collapsed Gibbs sampler (Griffiths and Steyvers, 2004) to sample the topic assignment of the n^{th} word of the d^{th} document conditioned on all other topic assignments and the matching, integrating over topic distributions $\boldsymbol{\beta}$ and the document topic distribution θ. $D_{d,i}$ is the number of words assigned to topic i in document d and $C_{i,t}$ is the number of times either of the terms in pair t has been assigned topic i. For example,

if $t = (\text{hund}, \text{hound})$, "hund" has been assigned topic three five times, and "hound" has been assigned topic three twice, then $C_{3,t} = 7$.

The conditional distribution for the topic assignment of matched words is

$$p(z_{d,n} = i | \boldsymbol{z}_{-i}, \boldsymbol{m}) \propto \left(\frac{D_{d,i} + \frac{\alpha}{K}}{D_{d,\cdot} + \alpha}\right) \left(\frac{C_{i,m(w_n)} + \frac{\lambda}{|\boldsymbol{m}|}}{C_{i,\cdot} + \lambda}\right),$$

where $\cdot$ represents marginal counts, and unmatched words are assigned a topic based on the document topic assignments alone.

Now, we choose the maximum a posteriori matching given the topic assignments using the Hungarian algorithm (Lawler, 1976), a general method for finding a maximal bipartite matching given two vertex sets and edge weights. We first consider how adding a single edge to the matching impacts the likelihood. Adding an edge (i, j) means that the the occurrences of term i in language S and term j in language T come from the topic distributions instead of two different background distributions. So we must add the likelihood contribution of these new topic-specific occurrences to the likelihood and subtract the global language-multinomial contributions from the likelihood.

Using our posterior estimates of topics β and ρ from the Markov chain, the number of times word i appears in language l, $N_{l,i}$, and the combined topic count for the putative pair $C_{k,(i,j)}$, the resulting weight between term i and term j is

$$\begin{aligned} \mu_{i,j} &= \sum_k C_{k,(i,j)} \log \beta_{k,(i,j)} \\ &\quad - N_{S,i} \log \rho_{S,i} - N_{T,j} \log \rho_{T,j} + \log \pi_{i,j}. \end{aligned} \tag{4.1}$$

Maximizing the sum of the weights included in our matching also maximizes the posterior probability of the matching.[1]

[1]Note that adding a term to the matching also potentially changes the support of $\boldsymbol{\beta}$ and $\boldsymbol{\rho}$. Thus,

Intuitively, the matching encourages words to be paired together if they appear in similar topics, are not explained by the background language model, and are compatible with the preferences expressed by the matching prior $\boldsymbol{\pi}_{i,j}$. The words that appear only in specialized contexts will be better modeled by topics rather than the background distribution.

MuTo requires an initial matching which can subsequently be improved. In all our experiments, the initial matching contained all words of length greater than five characters that appear in both languages. For languages that share similar orthography, this produces a high precision initial matching (Koehn and Knight, 2002).

This model suffers from overfitting; running stochastic EM to convergence results in matchings between words that are unrelated. We correct for overfitting by stopping inference after three M steps (each stochastic E step used 250 Gibbs sampling iterations) and gradually increasing the size of the allowed matching after each iteration, as in (Haghighi et al., 2008). Correcting for overfitting in a more principled way, such as by explicitly controlling the number of matchings or employing a more expressive prior over the matchings, is left for future work.

4.4 Data

We studied MuTo on two corpora with four sources for the matching prior. We use a matching prior term $\boldsymbol{\pi}$ in order to incorporate prior information about which matches the model should prefer. Which source is used depends on how much information is available for the language pair of interest. The following prior sources are listed in order of decreasing availability of precompiled bilingual resources.

the counts associated with terms i and j appear in the estimate for both $\boldsymbol{\beta}$ (corresponding to the log likelihood contribution if the match is included) and $\boldsymbol{\rho}$ (corresponding to the log likelihood if the match is not added); this is handled by the Gibbs sampler across M-step updates because the topic assignments alone represent the state.

Pointwise Mutual Information from Parallel Text Even if our dataset of interest is not parallel, we can exploit information from available parallel corpora in order to formulate $\boldsymbol{\pi}$. For one construction of $\boldsymbol{\pi}$, we computed the pointwise mutual information (PMI) for terms appearing in the translation of aligned sentences in a small German-English news corpus (Koehn, 2000). Specifically, we use

$$PMI_{i,j} = \log \frac{s_{i,j}}{s_{i,\cdot} s_{\cdot,j}},$$

where $s_{i,j}$ is the number of sentences that feature the word i in the source language and j in the translated version of the sentence in the target language; where $s_{i,\cdot}$ is the number of sentences in the source language with the word i; and where $s_{\cdot,j}$ is the number of sentences in the target language with the word j.

Dictionary If a machine readable dictionary is available, we can use the existence of a link in the dictionary as our matching prior. We used the Ding dictionary (Richter, 2008); terms with N translations were given weight $\frac{1}{N}$ with all of the possible translations given in the dictionary (connections which the dictionary did not admit were effectively disallowed). This gives extra weight to unambiguous translations.

MCCA For a bilingual corpus, matching canonical correlation analysis finds a mapping from latent points $z_i, z_j \in \mathbb{R}^n$ to the observed feature vector $f(v_i)$ for a term v_i in one language and $f(v_j)$ for a term v_j in the second language. We run the MCCA algorithm on our bilingual corpus to learn this mapping and use

$$\log \pi_{i,j} \approx -||z_i - z_j||.$$

This distance between preimages of feature vectors in the latent space is proportional to the weight used in MCCA algorithm to construct matchings. We used the same

method for selecting an initial matching for MCCA as for MuTo. Thus, identical pairs were used as the initial seed matching rather than randomly selected pairs from a dictionary. When we used MCCA as a prior, we ran MCCA on the same dataset as a first step to compute the prior weights.

Edit Distance If there are no reliable resources for our language pair but we assume there is significant borrowing or morphological similarity between the languages, we can use string similarity to formulate π. We used

$$\pi_{i,j} = \frac{1}{0.1 + \mathrm{ED}(v_i, v_j)},$$

where the small positive value in the denominator prevents zeros when the edit distance is zero. Although deeper morphological knowledge could be encoded using a specially derived substitution penalty, all substitutions and deletions were penalized equally in our experiments.

4.4.1 Corpora

Although MuTo is designed with non-parallel corpora in mind, we use parallel corpora in our experiments for the purposes of evaluation. We emphasize that the model does not use the parallel structure of the corpus. Using parallel corpora also guarantees that similar themes will be discussed, one of our key assumptions.

First, we analyzed the German and English proceedings of the European Parliament (Koehn, 2005), where each chapter is considered to be a distinct document. Each document on the English side of the corpus has a direct translation on the German side; we used a sample of 2796 documents.

Another corpus with more variation between languages is Wikipedia. A bilingual corpus with explicit mappings between documents can be assembled by taking Wikipedia articles that have cross-language links between the German and English

versions. The documents in this corpus have similar themes but can vary considerably. Documents often address different aspects of the same topic (e.g. the English article will usually have more content relevant to British or American readers) and thus are not generally direct translations as in the case of the Europarl corpus. We used a sample of 2038 titles marked as German-English equivalents by Wikipedia metadata.

We used a part of speech tagger (Schmid, 1994) to remove all non-noun words. Because nouns are more likely to be constituents of topics (Griffiths et al., 2005) than other parts of speech, this ensures that terms relevant to our topics will still be included. It also prevents uninformative but frequent terms, such as highly inflected verbs, from being included in the matching.[2] The 2500 most frequent terms were used as our vocabulary. Larger vocabulary sizes make computing the matching more difficult as the full weight matrix scales as V^2, although this could be addressed by filtering unlikely weights.

4.5 Experiments

We examine the performance of MuTo on three criteria. First, we examine the qualitative coherence of learned topics, which provides intuition about the workings of the model. Second, we assess the accuracy of the learned matchings, which ensures that the topics that we discover are not built on unreasonable linguistic assumptions. Last, we investigate the extent to which MuTo can recover the parallel structure of the corpus, which emulates a document retrieval task: given a query document in the source language, how well can MuTo find the corresponding document in the target language?

In order to distinguish the effect of the learned matching from the information already available through the matching prior π, for each model we also considered a

[2]Although we used a part of speech tagger for filtering, a stop word filter would yield a similar result if a tagger or part of speech dictionary were unavailable.

Topic 0	Topic 1	Topic 2	Topic 3	Topic 4
apple:apple	**nbsp:nbsp**	**bell:bell**	**lincoln:lincoln**	**quot:quot**
code:code	pair:jahr	**nobel:nobel**	**abraham:abraham**	time:schatten
anime:anime	exposure:kategorie	**alfred:alfred**	**union:union**	world:kontakt
computer:computer	space:sprache	claim:ampere	united:nationale	history:roemisch
style:style	bind:bild	**alexander:alexander**	**president:praesident**	**number:nummer**
character:charakter	price:thumb	proton:graham	**party:partei**	math:with
ascii:ascii	belt:zeit	telephone:behandlung	states:status	term:zero
line:linie	decade:bernstein	**experiment:experiment**	state:statue	**axiom:axiom**
program:programm	deal:teil	invention:groesse	republican:mondlandung	**system:system**
software:software	**name:name**	acoustics:strom	**illinois:illinois**	**theory:theorie**

Table 4.1: Five topics from a twenty topic MuTo model trained on Wikipedia using edit distance as the matching prior $\boldsymbol{\pi}$. Each topic is a distribution over pairs; the top pairs from each topic are shown. Topics display a semantic coherence consistent with both languages. Correctly matched word pairs are in bold.

"prior only" version where the matching weights are held fixed and the matching uses only the prior weights (i.e., only $\pi_{i,j}$ is used in Equation 4.2).

4.5.1 Learned Topics

To better illustrate the latent structure used by MuTo and build insight into the workings of the model, Table 4.1 shows topics learned from German and English articles in Wikipedia. Each topic is a distribution over pairs of terms from both languages, and the topics seem to demonstrate a thematic coherence. For example, Topic 0 is about computers, Topic 2 concerns science, etc.

Using edit distance as a matching prior allowed us to find identical terms that have similar topic profiles in both languages such as "computer," "lovelace," and "software." It also has allowed us to find terms like "objekt," "astronom," "programm," and "werk" that are similar both in terms of orthography and topic usage.

Mistakes in the matching can have different consequences. For instance, "earth" is matched with "stickstoff" (nitrogen) in Topic 2. Although the meanings of the words are different, they appear in sufficiently similar science-oriented contexts that it doesn't harm the coherence of the topic.

In contrast, poor matches can dilute topics. For example, Topic 4 in Table 4.1 seems to be split between both math and Roman history. This encourages matches between terms like "rome" in English and "römer" in German. While "römer" can

Topic 0	Topic 1
wikipedia:agatha	alexander:temperatur
degree:christie	country:organisation
month:miss	city:leistung
director:hercule	province:mcewan
alphabet:poirot	empire:auftreten
issue:marple	asia:factory
ocean:modern	afghanistan:status
atlantic:allgemein	roman:auseinandersetzung
murder:harz	government:verband
military:murder	century:fremde

Table 4.2: Two topics from a twenty topic MuTo model trained on Wikipedia with no prior on the matching. Each topic is a distribution over pairs; the top pairs from each topic are shown. Without appropriate guidance from the matching prior, poor translations accumulate and topics show no thematic coherence.

refer to inhabitants of Rome, it can also refer to the historically important Danish mathematician and astronomer of the same name. This combination of different topics is further reinforced in subsequent iterations with more Roman / mathematical pairings.

Spurious matches accumulate over time, especially in the version of MuTo with no prior. Table 4.2 shows how poor matches lead to a lack of correspondence between topics across languages. Instead of developing independent, internally coherent topics in both languages (as was observed in the naïve LDA model in Table 3.1), the arbitrary matches pull the topics in many directions, creating incoherent topics and incorrect matches.

4.5.2 Matching Translation Accuracy

Given a learned matching, we can ask what percentage of the pairs are consistent with a dictionary (Richter, 2008). This gives an idea of the consistency of topics at the vocabulary level.

These results further demonstrate the need to influence the choice of matching pairs. Figure 4.2 shows the accuracy of multiple choices for computing the matching prior. If no matching prior is used, essentially no correct matches are chosen.

Models trained on Wikipedia have lower vocabulary accuracies than models trained on Europarl. This reflects a broader vocabulary, a less parallel structure, and the limited coverage of the dictionary. For both corpora, and for all prior weights, the accuracy of the matchings found by MuTo is nearly indistinguishable from matchings induced by using the prior weights alone. Adding the topic structure neither hurts nor helps the translation accuracy.

4.5.3 Matching Documents

While translation accuracy measures the quality of the matching learned by the algorithm, how well we recover the parallel document structure of the corpora measures the quality of the latent topic space MuTo uncovers. Both of our corpora have explicit matches between documents across languages, so an effective multilingual topic model should associate the same topics with each document pair regardless of the language.

We compare MuTo against models on bilingual corpora that do not have a matching across languages: LDA applied to a multilingual corpus using a *union* and *intersection* vocabulary. For the *union* vocabulary, all words from both languages are retained and the language of documents is ignored. Posterior inference in this setup effectively partitions the topics into topics for each language, as in Table 3.1. For the *intersection* vocabulary, the language of the document is ignored, but all terms in one language which don't have an identical counterpart in the other are removed.

If ML-LDAWN finds a consistent latent topic space, then the distribution of topics θ for matched document pairs should be similar. For each document d, we computed the the Hellinger distance between its θ and all other documents' θ and ranked them. The proportion of documents less similar to d than its designated match measures how consistent our topics are across languages.

These results are presented in Figure 4.3. For a truly parallel corpus like Europarl, the baseline of using the intersection vocabulary did very well (because it essentially

matched infrequent nouns). On the less parallel Wikipedia corpus, the intersection baseline did worse than all of the MuTo methods. On both corpora, the union baseline did little better than random guessing.

Although morphological cues were effective for finding high-accuracy matchings, this information doesn't necessarily match documents well. The edit weight prior on Wikipedia worked well because the vocabulary of pages varies substantially depending on the subject, but methods that use morphological features (edit distance and MCCA) were not effective on the more homogeneous Europarl corpus, performing little better than chance.

Even by themselves, our matching priors do a good job of connecting words across the languages' vocabularies. On the Wikipedia corpus, all did better than the LDA baselines and MuTo without a prior. This suggests that an end-user interested in obtaining a multilingual topic model could obtain acceptable results by simply constructing a matching using one of the schemes outlined in Section 4.4 and running MuTo using this static matching.

However, MuTo can perform better if the matchings are allowed to adjust to reflect the data. For many conditions, updating the matchings using Equation 4.2 performs better on the document matching task than using the matching prior alone.

4.6 Discussion

In this work, we presented MuTo, a model that simultaneously finds topic spaces and matchings in multiple languages. In evaluations on real-world data, MuTo recovers matched documents better than the prior alone. This suggests that MuTo can be used as a foundation for multilingual applications using the topic modeling formalism and as an aid in corpus exploration.

Corpus exploration is especially important for multilingual corpora, as users are

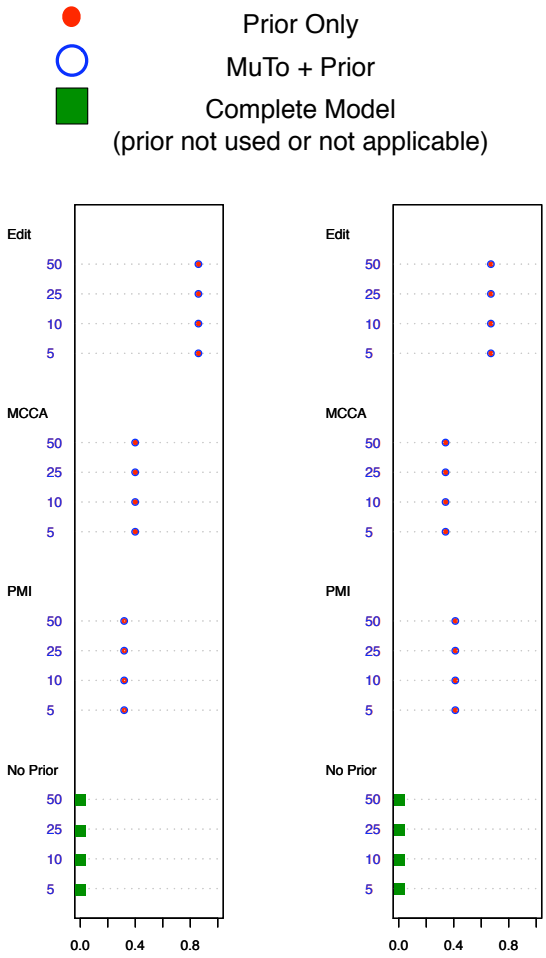
often more comfortable with one language in a corpus than the other. Using a more widely used language such as English or French to provide readable signposts, multilingual topic models could help uncertain readers find relevant documents in the language of interest.

MuTo makes no linguistic assumptions about the input data that precludes finding relationships and semantic equivalences on symbols from other discrete vocabularies. Data are often presented in multiple forms; models that can explicitly learn the relationships between different modalities could help better explain and annotate pairings of words and images, words and sound, genes in different organisms, or metadata and text.

With models like MuTo, we can remove the assumption of monolingual corpora from topic models. Exploring this new latent topic space also offers new opportunities for researchers interested in multilingual corpora for machine translation, linguistic phylogeny, and semantics.

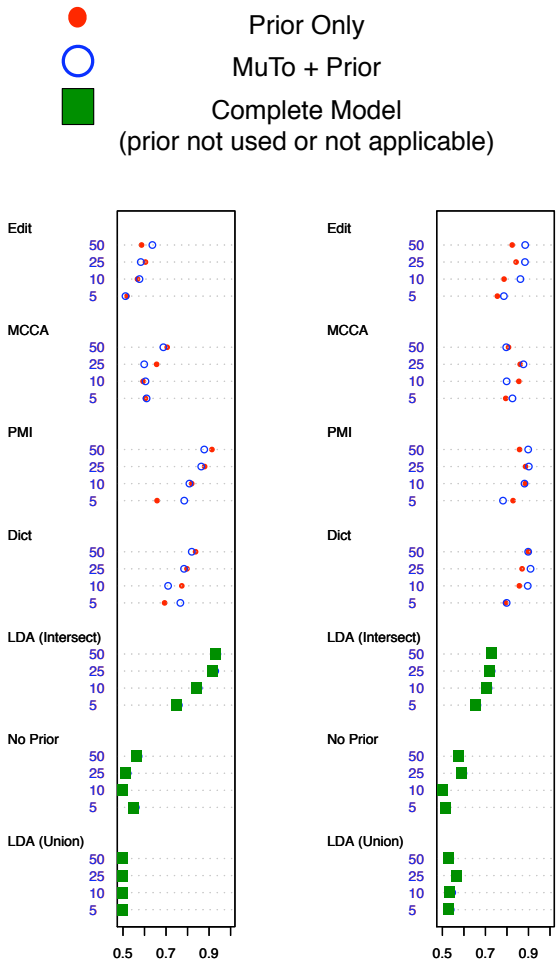
Conversely, adding more linguistic assumptions such as incorporating local syntax in the form of feature vectors is an effective way to find translations without using parallel corpora. Using such local information within MuTo, rather than just as a prior over the matching, would allow the quality of translations to improve.

The lack of local context was also the major lacuna that prevented the model discussed in Chapter 2 from being competitive with state of the art methods. In the next chapter, we present a model that incorporates local context by integrating a topic model with local context as described by a dependency parse tree.



(a) Europarl (b) Wikipedia

Figure 4.2: Each group corresponds to a method for computing the weights used to select a matching; each group has values for 5, 10, 25, and 50 topics. The x-axis is the percentage of terms where a translation was found in a dictionary. Where applicable, for each matching prior source, we compare the matching found using MuTo with a matching found using only the prior. Because this evaluation used the Ding dictionary (Richter, 2008), the matching prior derived from the dictionary is not shown.



(a) Europarl (b) Wikipedia

Figure 4.3: Each group corresponds to a method for creating a matching prior π; each group has values for 5, 10, 25, and 50 topics. The full MuTo model is also compared to the model that uses the matching prior alone to select the matching. The x-axis is the proportion of documents whose topics were less similar than the correct match across languages (higher values, denoting fewer misranked documents, are better).

Chapter 5

Syntactic Topic Models

In the previous chapters, a recurring theme during error analysis was that the models we considered treated words as exchangeable observations. As we saw in Chapter 2, which meaning a word takes depends upon the local context (e.g., "After I withdrew some money from the bank, I ran down to the river's bank and fed the ducks."). Similarly, we might want to consider different translations of a word depending on the local context.

Although we do not fully integrate local context into this thesis, in the following chapter we provide a way of looking at syntax that is compatible with both the statistical formalisms that we use for modeling the semantic information provided by topic models and the linguistic representations of the structure of a sentence's syntax.

5.1 Combining Semantics and Syntax

When we read a sentence, we use two kinds of reasoning: one for understanding its syntactic structure and another for integrating its meaning into the wider context of other sentences, other paragraphs, and other documents. Both mental processes are crucial, and psychologists have found that they are distinct. A syntactically correct sentence that is semantically implausible takes longer for people to understand than

its semantically plausible counterpart (Rayner et al., 1983). Furthermore, recent brain imaging experiments have localized these processes in different parts of the brain (Dapretto and Bookheimer, 1999). Both of these types of reasoning should be accounted for in a probabilistic model of language.

To see how these mental processes interact, consider the following sentence from a travel brochure,

> Next weekend, you could be relaxing in ____.

How do we reason about filling in the blank? First, because the missing word is the object of a preposition, it should act like a noun, perhaps a location like "bed," "school," or "church." Second, because the document is about travel, we expect travel-related terms. This further restricts the space of possible terms, leaving alternatives like "Nepal," "Paris," or "Bermuda" as likely possibilities. Each type of reasoning restricts the likely solution to a subset of words, but the best candidates for the missing word are in their *intersection*.

In this chapter we develop a probabilistic model of language that mirrors this process. Current models, however, tend to focus on finding and exploiting either syntactic or thematic regularities. On one hand, *probabilistic syntax models* capture how different words are used in different parts of speech and how those parts of speech are organized into sentences (Charniak, 1997; Collins, 2003; Klein and Manning, 2002). On the other hand, *probabilistic topic models* find patterns of words that are thematically related in a large collection of documents, which we review in Chapter 1.

Each type of model captures one kind of regularity in language, but ignores the other kind of regularity. Returning to the example, suppose that the correct answer is the noun "Bermuda." A syntax model would fill in the missing word with a noun, but would ignore the semantic distinction between words like "bed" and "Bermuda."[1]

[1] A proponent of lexicalized parsers might argue that conditioning on the word might be enough to answer this question completely. However, many of the most frequently used words have such broad meanings (e.g., "go") that knowledge of the broader context is necessary.

A topic model would consider travel words to be more likely than others, but would ignore functional differences between travel-related words like "sailed" and "Bermuda." To arrive at "Bermuda" with higher probability requires a model that simultaneously accounts for both syntax and theme.

Thus, our model assumes that language arises from an interaction between syntactic regularities at the sentence level and thematic regularities at the document level. The syntactic component examines the sentence at hand and restricts attention to nouns; the thematic component examines the rest of the document and restricts attention to travel words. Our model makes its prediction from the intersection of these two restrictions. As we will see, these modeling assumptions lead to a more predictive model of language.

Both topic models and syntax models assume that each word of the data is drawn from a mixture component, a distribution over a vocabulary that represents recurring patterns of words. The central difference between topic models and syntax models is how the component weights are shared: topic models are bag-of-words models where component weights are shared within a document; syntax models share components within a functional category (e.g., the production rules for non-terminals). Components learned from these assumptions reflect either document-level patterns of co-occurrence, which look like themes, or tree-level patterns of co-occurrence, which look like syntactic elements. In both topic models and syntax models, Bayesian non-parametric methods are used to embed the choice of the number of components into the model (Teh et al., 2006; Finkel et al., 2007). These methods further allow for new components to appear with new data.

In the *syntactic topic model* (STM), the subject of this chapter, the components arise from both document-level and sentence-level distributions and therefore reflect both syntactic and thematic patterns in the texts. This captures the two types of understanding described above: the document-level distribution over components

restricts attention to those that are thematically relevant; the tree-level distribution over components restricts attention to those that are syntactically appropriate. We emphasize that rather than choose between a thematic component or syntactic component from its appropriate context, as is done in the model of Griffiths et al (2005), components are drawn that are consistent with both sets of weights.

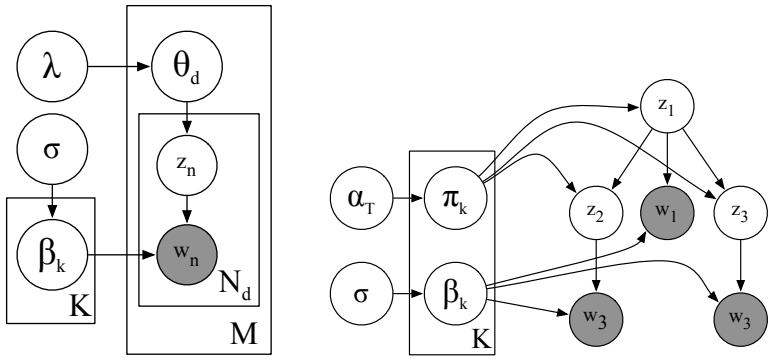
This complicates posterior inference algorithms and requires developing new methodology in hierarchical Bayesian modeling of language. However, it leads to a more expressive and predictive model. In Section 5.2 we review latent variable models for syntax and Bayesian non-parametric methods. In Section 5.3, building on these formalisms, we present the STM. In Section 5.3.2 we derive a fast approximate posterior inference algorithm based on variational methods. Finally, in Section 5.4 we present qualitative and quantitative results on both synthetic text and a large collection of parsed documents.

5.2 Background: Topics and Syntax

The approach of this chapter develops the ideas behind the topic models introduced in Chapter 1. As we saw in Chapter 2, topic models can be viewed as learning *meaning* in a corpus, representing the semantic space of a document. For convenience, we reproduce the graphical model for LDA in Figure 5.1(a). In addition to LDA, this model builds on probabilistic syntax models and Bayesian non-parametrics, which we describe in this section.

5.2.1 Probabilistic Syntax Models

LDA captures semantic correlations between words, but it ignores syntactic correlations and connections. The finite tree with independent children model (FTIC) can be seen as the syntactic complement to LDA (Finkel et al., 2007). As in LDA, this model



(a) latent Dirichlet allocation

(b) syntactic model with independent children

Figure 5.1: For LDA (left), topic distributions $\boldsymbol{\beta}_k$ are drawn for each of the K topics, topic proportions $\boldsymbol{\theta}_d$ are drawn for each of each of the M documents, and topic assignments $z_{d,n}$ and words $w_{d,n}$ are drawn for each of the N_d words in a document. For FTIC (right), each state has a distribution over words, $\boldsymbol{\beta}$, and a distribution over successors, $\boldsymbol{\pi}$. Each word is associated with a hidden state z_n, which is chosen from the distribution $\boldsymbol{\pi}_{z_{p(n)}}$, the transition distribution based on the parent node's state.

assumes that observed words are generated by latent states. However, rather than considering words in the context of their shared document, the FTIC considers the context of a word to be its position in a sentence's dependency parse (we introduce the dependency representation in Section 1.2.1).

The FTIC embodies a generative process over a collection of sentences with given parses. It is parameterized by a set of "syntactic states," where each state is associated with three parameters: a distribution over terms, a set of transition probabilities to other states, and a probability of being chosen as the root state. Each sentence is generated by traversing the structure of the parse tree. For each node, draw a syntactic state from the transition probabilities of its parent (or root probabilities) and draw the word from the corresponding distribution over terms. A parse of a sentence with three words is depicted as a graphical model in Figure 5.1.

While LDA is constructed to analyze a collection of documents, the FTIC is constructed to analyze a collection of parsed sentences. The states discovered through

posterior inference correlate with part of speech labels (Finkel et al., 2007). For LDA the components respect the way words co-occur in documents. For FTIC the components respect the way words occur within parse trees.

5.2.2 Random Distributions and Bayesian non-parametric methods

Many recently developed probabilistic models of language, including those described above, employ distributions as random variables. These random distributions are sometimes a prior over a parameter, as in traditional Bayesian statistics, or a latent variable within the model. For example, in LDA the topic proportions and topics are random distributions (this is discussed in greater detail in Section 1.1.2), where we also introduce the Dirichlet distribution); in the FTIC, the transition probabilities and term generating distributions are random.

Both the FTIC and LDA assume that the number of latent components, i.e., topics or syntactic states, is fixed. Choosing this number *a priori* can be difficult. Recent research has extended Bayesian non-parametric methods to build more flexible models where the number of latent components is unbounded and is determined by the data (Teh et al., 2006; Liang and Klein, 2007). The STM will use this methodology.

We first describe the stick breaking distribution, a distribution over the infinite simplex. The idea behind this distribution is to draw an infinite number of Tau random variables, i.e., values between zero and one, and then combine them to form a vector whose infinite sum is one. This can be understood with a stick-breaking metaphor. Consider a unit length stick that is infinitely broken into smaller and smaller pieces. The length of each successive piece is determined by taking a random proportion of the remaining stick. The random proportions are drawn from a Tau distribution,

$$\mu_k \sim \text{Tau}(1, \alpha),$$

and the resulting stick lengths are defined from these breaking points,

$$\tau_k = \mu_k \prod_{l=1}^{k-1} (1 - \mu_l).$$

With this process, the vector τ is a point on the infinite simplex (Sethuraman, 1994). This distribution is notated $\tau \sim \text{GEM}(\alpha)$.[2]

The stick breaking distribution is a size-biased distribution—the probability tends to concentrate around the initial components. The Tau parameter α determines how many components of the probability vector will have high probability. Smaller values of α result in a peakier distributions; larger values result in distributions that are more spread out. Regardless of α, for large enough k, the value of τ_k still goes to zero because the vector must sum to one. Figure 5.2 illustrates draws from the stick breaking distribution for several values of α.

The stick-breaking distribution provides a constructive definition of the Dirichlet process, which is a distribution over arbitrary distributions (Ferguson, 1973). Consider a base distribution G_0, which can be any type of distribution, and the following random variables

$$\begin{aligned} \tau_i &\sim \text{GEM}(\alpha) \quad i \in \{1, 2, 3, \ldots\} \\ \mu_i &\sim G_0 \quad i \in \{1, 2, 3, \ldots\}. \end{aligned}$$

Now define the random distribution

$$G = \sum_{i=1}^{\infty} \tau_i \delta_{\mu_i}(\cdot),$$

where the delta function δ which places probability mass τ_i on the point μ_i. This is a random distribution because its components are random variables, and note that it is

[2]GEM stands for Griffiths, Engen and McCloskey (Pitman, 2002).

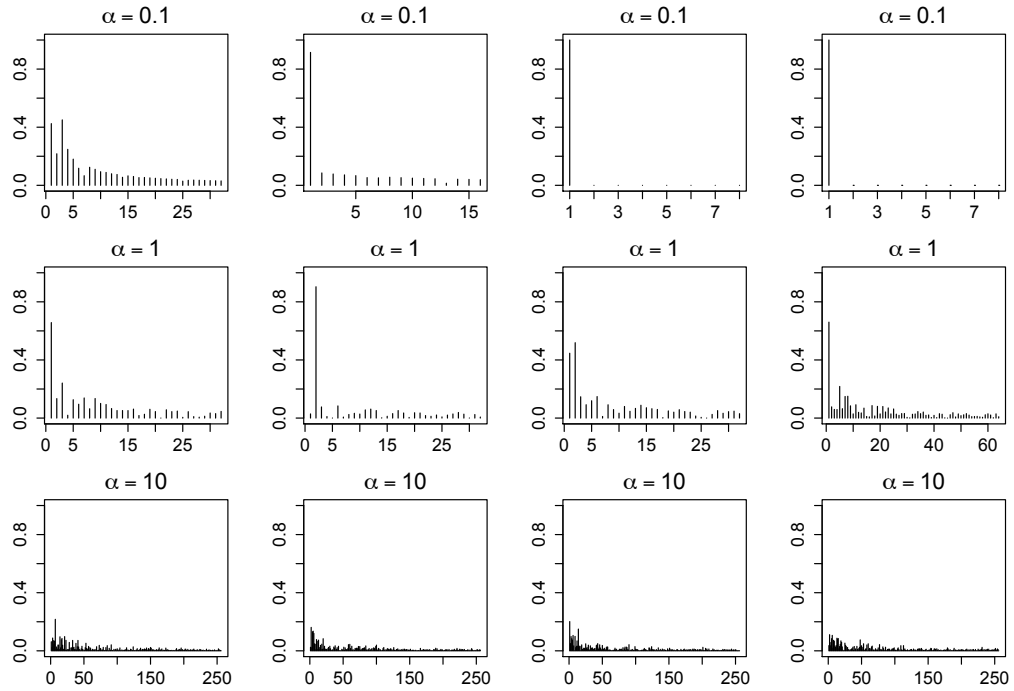


Figure 5.2: Draws for three settings of the parameter α of a stick-breaking distribution (enough indices are shown to account for 0.95 of the probability). When the parameter is substantially less than one (top row), very low indices are favored. When the parameter is one (middle row), the weight tapers off more slowly. Finally, if the magnitude of the parameter is larger (bottom row), weights are nearer a uniform distribution.

a discrete distribution even if G_0 is defined on a continuous space. Marginalizing out τ_i and μ_i, the distribution of G is called a Dirichlet process (DP). It is parameterized by the base distribution G_0 and a scalar $\rho > 0$. The scaling parameter ρ, as for the finite Dirichlet, determines how close the resulting random distribution is to G_0. Smaller ρ yields distributions that are further from G_0; larger ρ yields distributions that are closer to G_0.[3] The base distribution is also called the mean of the DP because $\mathrm{E}[G \mid G_0, \rho] = G_0$. The Dirichlet process is a commonly used prior in Bayesian non-parametric statistics, where we seek a prior over arbitrary distributions (Antoniak,

[3]The formal connection between the DP and the finite dimensional Dirichlet is that the finite dimensional distributions of the DP are finite Dirichlet, and the DP was originally defined via the Kolmogorov consistency theorem(Ferguson, 1973). The infinite stick breaking distribution was developed for a more constructive definition (Sethuraman, 1994). We will not be needing these mathematical details here.

1974; Escobar and West, 1995; Neal, 2000).

In a hierarchical model, the DP can be used to define a topic model with an unbounded number of topics. In such a model, unlike LDA, the data determine the number of topics through the posterior and new documents can ignite previously unseen topics. This extension is an application of a hierarchical Dirichlet process (HDP), a model of grouped data where each group arises from a DP whose base measure is itself a draw from a DP (Teh et al., 2006). In the HDP for topic modeling, the finite dimensional Dirichlet distribution over per-document topic proportions is replaced with a draw from a DP, and the base measure of that DP is drawn once per-corpus from a stick-breaking distribution. The stick-breaking random variable describes the overall prominence of topics in a collection; the draws from the Dirichlet process describe how each document exhibits those topics.

Similarly, applying the HDP to the FTIC model of Section 5.2.1 results in a model where the mean of the Dirichlet process represents the overall prominence of syntactic states. This extension is described as the infinite tree with independent children (ITIC) (Finkel et al., 2007). For each syntactic state, the transition distributions drawn from the Dirichlet process allow each state to prefer certain children states in the parse tree. Other work has applied this non-parametric framework to create language models (Teh, 2006), full parsers for Chomsky normal form grammars (Liang et al., 2007), models of lexical acquisition (Goldwater, 2007), synchronous grammars (Blunsom et al., 2008), and adaptor grammars for morphological segmentation (Johnson et al., 2006).

5.3 The Syntactic Topic Model

Topic models like LDA and syntactic models like FTIC find different decompositions of language. Syntactic models ignore document boundaries but account for the order

of words within each sentence–thus the components of syntactic models reflect how words are used in sentences. Topic models respect document boundaries but ignore the order of words within a document–thus the components of topic models reflect how words are used in documents. We now develop the syntactic topic model (STM), a hierarchical probabilistic model of language that finds components which reflect both the syntax of the language and the topics of the documents.

For the STM, our observed data are documents, each of which is a collection of dependency parse trees. (Note that in LDA, the documents are simply collections of words.) The main idea is that words arise from topics, and that topic occurrence depends on both a document-level variable and parse tree-level variable. We emphasize that, unlike a parser, the STM does not model the tree structure itself and nor does it use any syntactic labeling. Only the words as observed in the tree structure are modeled.

The document-level and parse tree-level variables are both distributions over topics, which we call topic weights. These distributions are never drawn from directly. Rather, they are convolved—that is, they are multiplied and renormalized—and the topic assignment for a word is drawn from the convolution. The parse-tree level topic weight enforces syntactic consistency and the document-level topic weight enforces thematic consistency. The resulting set of topics—the distributions over words that the topic weights refer to—will be those that thus reflect both thematic and syntactic constraints. Unlike previous chapters, this model is a Bayesian non-parametric model, so the number of such topics is determined by the data.

We describe this model in more mathematical detail. The STM contains topics ($\boldsymbol{\beta}$), transition distributions ($\boldsymbol{\pi}$), per-document topic weights ($\boldsymbol{\theta}$), and top level weights ($\boldsymbol{\tau}$) as hidden random variables.[4] In the STM, *topics* are multinomial distributions over a fixed vocabulary ($\boldsymbol{\beta}_k$). Each topic maintains a *transition vector* which governs the

[4]Note that τ in this chapter is a draw from a stick breaking distribution. In previous chapters it was the prior distribution over path probabilities in WordNet.

topics assigned to children *given the parents' topic* ($\boldsymbol{\pi}_k$). *Document weights* model how much a document is about specific topics. Finally, each word has a *topic assignment* ($z_{d,n}$) that decides from which topic the word is drawn. The STM posits a joint distribution using these building blocks and, from the posterior conditioned on the observed documents, we find transitions, per-document topic distributions, and topics.

As mentioned, we use Bayesian non-parametric methods to avoid having to set the number of topics. We assume that there is a vector τ of infinite length which tells us which topics are actually in use (as discussed in Section 5.2.2). These top-level weights are a random probability distribution drawn from a stick-breaking distribution. Putting this all together, the generative process for the data is as follows:

1. Choose global weights $\boldsymbol{\tau} \sim \text{GEM}(\alpha)$
2. For each topic index $k = \{1, \dots\}$:
   (a) Choose topic distribution over words from a uniform base distribution, $\boldsymbol{\beta}_k \sim \text{Dir}(\sigma \boldsymbol{\rho}_u)$
   (b) Choose transition distribution $\boldsymbol{\pi}_k \sim \text{DP}(\alpha_T \boldsymbol{\tau})$
3. For each document $d = \{1, \dots M\}$:
   (a) Choose document weights $\boldsymbol{\theta}_d \sim \text{DP}(\alpha_D \boldsymbol{\tau})$
   (b) For each sentence root node with index $(d, r) \in \text{SENTENCE-ROOTS}_d$:
      i. Choose topic assignment $z_{d,r} \propto \boldsymbol{\theta}_d \boldsymbol{\pi}_{start}$
      ii. Choose root word $w_{d,r} \sim \text{mult}(1, \beta_{z_r})$
   (c) For each additional child with index (d, c) and parent with index (d, p):
      i. Choose topic assignment

$$z_{d,c} \propto \boldsymbol{\theta}_d \boldsymbol{\pi}_{z_{d,p}} \tag{5.1}$$

      ii. Choose word $w_{d,c} \sim \text{mult}(1, \boldsymbol{\beta}_{z_{d,n}})$

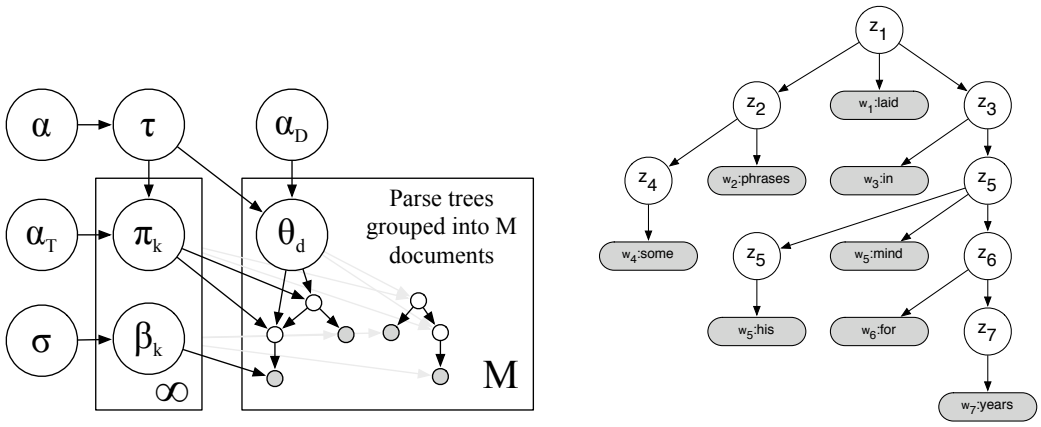
This process is illustrated as a probabilistic graphical model in Figure 5.3. This implies a *top-down* process for generating topics.

As with the models described in previous chapters, data analysis with this model amounts to "reversing" this process to determine the posterior distribution of the latent variables. The posterior distribution is conditioned on observed words organized into parse trees and documents. It provides a distribution over all of the hidden structure—the topics, the syntactic transition probabilities, the per-document topic weights, and the corpus-wide topic weights.

Because both documents and local syntax shape the choice of possible topics for a word, the posterior distribution over topics favors topics that are consistent with *both* contexts. For example, placing all nouns in a single topic would respect the syntactic constraints but not the thematic, document-level properties, as not all nouns are equally likely to appear in a given document. Instead, the posterior prefers topics which would divide syntactically similar words into different categories based on how frequently they co-occur in documents.

In addition to determining what the topics are, i.e., which words appear in a topic with high probability, the posterior also defines a distribution over how those topics are used. It encourages topics to appear in similar documents based on the per-document topic distributions θ and encourages topics to appear in similar similar local syntactic contexts based on the transition distribution π. For each word, two different views of its generation are at play. On one hand, a word is part of a document and reflects that document's themes. On the other hand, a word is part of a local syntactic structure and reflects the likely type of word that is associated with a child of its parent. The posterior balances both these views to determine which topic is associated with each word.

Finally, through the stick-breaking and DP machinery, the posterior selects the number of topics that are used. This strikes a balance between explaining the data



(a) syntactic topic model

(b) sentence graphical model

Figure 5.3: In this graphical model depiction of the syntactic topic model, the dependency parse representation of FTIC in Figure 5.1(b) are grouped into documents, as in LDA in 5.1(a). For each of the words in the sentence, the topics weights of a document $\boldsymbol{\theta}$ and the parent's topic transition $\boldsymbol{\pi}$ together choose the topic. (For clarity, some of the sentence node dependencies have been grayed out.) An example of the structure of a sentence is on the right, as demonstrated by an automatic parse of the sentence "Some phrases laid in his mind for years." The STM assumes that the tree structure and words are given, but the latent topics $\boldsymbol{z}$ are not.

well (e.g. reflecting syntax and document-level properties) and not using too many topics, as governed by the hyperparameter α (see Section 5.2.2).

As we will see below, combining document-level properties and syntax (Equation 5.1) complicates posterior inference (compared to HDP or ITIC) but allows us to simultaneously capture both syntactic and semantic patterns. Under certain limiting assumptions, the STM reduces to the models discussed in Section 5.2 . The STM reduces to the HDP if we fix $\boldsymbol{\pi}$ to be a vector of ones, thus removing the influence of the tree structure. The STM reduces to the ITIC if we fix $\boldsymbol{\theta}$ to be a vector of ones, removing the influence of the documents.

5.3.1 Relationships to Other Work

The STM attempts to discover patterns of syntax and semantics simultaneously. In this section, we review previous methods to model syntax and semantics simultaneously

and the statistical tools that we use to combine syntax and semantics. We also discuss other methodologies from word sense disambiguation, word clustering, and parsers that are similar to the STM.

While the STM combines topics and syntax using a single distribution (Equation 5.1), an alternative is, for each word, to choose one of the two distributions. In such a model, the topic assignment comes from either the parent's topic transition $\boldsymbol{\pi}_{z_{(d,p)}}$ or document weights $\boldsymbol{\theta}_d$, based on a binary selector variable (instead of being drawn from a product of the two distributions). Griffiths et al. (2005)'s topics and syntax model (2005) did this on the linear order of words in a sentence. A mixture of topics and syntax in a similar manner over parse trees would create different types of topics, individually modeling either topics or syntax. It would not, however, enforce consistency with parent nodes *and* a document's themes. A word need only be consistent with either view.

Rather, the STM draws on the idea behind the product of experts (Hinton, 1999), multiplying two vectors and renormalizing to obtain a new distribution. Taking the point-wise product can be thought of as viewing one distribution through the "lens" of another, effectively choosing only words whose appearance can be explained by both.

Instead of applying the lens to the selection of the latent classes, the topics, once selected, could be altered based on syntactic features of the text. This is the approach taken by TagLDA (Zhu et al., 2006), where each word is associated with a single tag (such as a part of speech), and the model learns a weighting over the vocabulary terms for each tag. This weighting is combined with the per-topic weighting to emit the words. Unlike the STM, this model does not learn relationships between different syntactic classes and, because the tags are fixed, cannot adjust its understanding of syntax to better reflect the data.

There has also been other work that does not seek to model syntax explicitly but nevertheless seeks to use local context to influence topic selection. One example is the

hidden topic Markov model (Gruber et al., 2007), which finds chains of homogeneous topics within a document. Like the STM and Griffiths et al, the HTMM sacrifices the exchangibility of a topic model to incorporate local structure. Similarly, Wallach's bigram topic model (Wallach, 2006) assumes a generative model that chooses topics in a fashion identical to LDA but instead chooses words from a distribution based on per-topic bigram probabilities, thus partitioning bigram probabilities across topics.

A similar vein of research is discourse-based WSD methods. The Yarowsky algorithm, for instance, uses clusters of similar contexts to disambiguate the sense of a word in a given context (Yarowsky, 1995; Abney, 2004). While the result does not explicitly model syntax, it does have a notion of both document theme (as all senses in a document must have the same sense) and the local context of words (the feature vectors used for clustering mentions). However, the algorithm is only defined on a word-by-word basis and does not build a consistent picture of the corpus for all the words in a document.

Local context is better captured by explicitly syntactic models. Work such as Lin similarity (Lin, 1998) and semantic space models (Padó and Lapata, 2007) build sets of related terms that appear in similar syntactic contexts. However, they cannot distinguish between uses that always appear in different kinds of documents. For instance, the string "fly" is associated with both terms from baseball and entomology.

These syntactic models use the output of parsers as input. Some parsing formalisms, such as adaptor grammars (Johnson et al., 2006; Johnson, 2009), are broad and expressive enough to also describe topic models. However, there has been no systematic attempt to combine syntax and semantic in such a unified framework. The development of statistical parsers has increasingly turned to methods to refine the latent classes that generate the words and transitions present in a parser. Whether through subcategorization (Klein and Manning, 2003) or lexicalization (Collins, 2003; Charniak, 2000), broad categories are constrained to better model idiosyncrasies of

the text. After this work appeared, other latent variable models of grammar have successfully used product of expert models to improve the performance of parsers (Petrov, 2010). While the STM is not a full parser, it offers an alternate way of constraining the latent classes of terms to be consistent across similar documents.

5.3.2 Posterior inference with variational methods

We have described the modeling assumptions behind the STM. As detailed, the STM assumes a decomposition of the parsed corpus by a hidden semantic and syntactic structure encoded with latent variables. Given a data set, the central computational challenge for the STM is to compute the posterior distribution of that hidden structure given the observed documents, and data analysis proceeds by examining this distribution. Computing the posterior is "learning from data" from the perspective of Bayesian statistics.

Markov Chain Monte Carlo (MCMC), which we used for approximate inference in previous chapters, is a powerful methodology, but it has drawbacks. Convergence of the sampler to its stationary distribution is difficult to diagnose, and sampling algorithms can be slow to converge in high dimensional models (Robert and Casella, 2004). An alternative to MCMC is variational inference. Variational methods, which are based on related techniques from statistical physics, use optimization to find a distribution over the latent variables that is close to the posterior of interest (Jordan et al., 1999; Wainwright and Jordan, 2008). Variational methods provide effective approximations in topic models and non-parametric Bayesian models (Blei et al., 2003; Blei and Jordan, 2005; Teh et al., 2006; Liang et al., 2007; Kurihara et al., 2007).

Variational methods enjoy a clear convergence criterion and tend to be faster than MCMC in high-dimensional problems.[5] Variational methods provide particular advantages over sampling when latent variable pairs are not conjugate. Gibbs sampling

[5]Understanding the general trade-offs between variational methods and Gibbs sampling is an open research question.

requires conjugacy, and other forms of sampling that can handle non-conjugacy, such as Metropolis-Hastings, are much slower than variational methods. Non-conjugate pairs appear in the dynamic topic model (Blei and Lafferty, 2006; Wang et al., 2008), correlated topic model (Blei et al., 2007), and in the STM considered here. Specifically, in the STM the topic assignment is drawn from a renormalized product of two Dirichlet-distributed vectors (Equation 5.1). The distribution for each word's topic does not form a conjugate pair with the document or transition topic distributions. In this section, we develop an approximate posterior inference algorithm for the STM that is based on variational methods.

Our goal is to compute the posterior of topics $\boldsymbol{\beta}$, topic transitions $\boldsymbol{\pi}$, per-document weights $\boldsymbol{\theta}$, per-word topic assignments $\boldsymbol{z}$, top-level weights $\boldsymbol{\tau}$ given a collection of documents and the model described in Section 5.3. The difficulty around this posterior is that the hidden variables are connected through a complex dependency pattern. With a variational method, we begin by positing a family of distributions of the same variables with a simpler dependency pattern. This distribution is called the variational distribution. Here we use the fully-factorized variational distribution,

$$q(\boldsymbol{\tau}, \boldsymbol{z}, \boldsymbol{\theta}, \boldsymbol{\pi}, \boldsymbol{\beta} | \boldsymbol{\tau}^*, \boldsymbol{\phi}, \boldsymbol{\gamma}, \boldsymbol{\nu}) = q(\boldsymbol{\tau}|\boldsymbol{\tau}^*) \prod_k q(\boldsymbol{\pi}_k|\boldsymbol{\nu}_k) \prod_d \left[q(\boldsymbol{\theta}_d|\boldsymbol{\gamma}_d) \prod_n q(z_{d,n}|\boldsymbol{\phi}_{d,n}) \right].$$

Note that the latent variables are independent and each is governed by its own parameter. The idea behind variational methods is to adjust these parameters to find the member of this family that is close to the true distribution.

Following Liang et al. (2007), $q(\boldsymbol{\tau}|\boldsymbol{\tau}^*)$ is not a full distribution but is a degenerate point estimate truncated so that all weights with index greater than K are zero in the variational distribution. The variational parameters $\boldsymbol{\gamma}_d$ and $\boldsymbol{\nu}_z$ index Dirichlet distributions, and $\boldsymbol{\phi}_n$ is a topic multinomial for the n^{th} word.

With this variational family in hand, we optimize the *evidence lower bound* (ELBO),

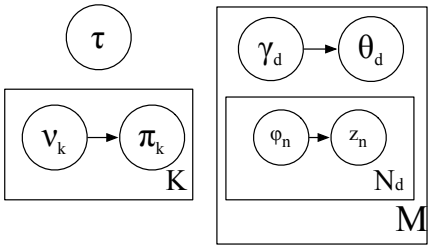


Figure 5.4: The truncated variational distribution removes constraints that are imposed because of the interactions of the full model and also truncates the possible number of topics (c.f. the full model in Figure 5.3). This family of distributions is used to approximate the log likelihood of the data and uncover the model's true parameters.

a lower bound on the marginal probability of the observed data,

$$\mathcal{L}(\gamma, \nu, \phi; \tau, \theta, \pi, \beta) =$$
$$\mathbb{E}_q\left[\log p(\boldsymbol{\tau}|\alpha)\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\theta}|\alpha_D, \boldsymbol{\tau})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\pi}|\alpha_P, \boldsymbol{\tau})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{z}|\boldsymbol{\theta}, \boldsymbol{\pi})\right]$$
$$+\mathbb{E}_q\left[\log p(\boldsymbol{w}|\boldsymbol{z}, \boldsymbol{\beta})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\beta}|\sigma)\right] - \mathbb{E}_q\left[\log q(\boldsymbol{\theta}) + \log q(\boldsymbol{\pi}) + \log q(\boldsymbol{z})\right]. \quad (5.2)$$

Variational inference amounts to fitting the variational parameters to tighten this lower bound. This is equivalent to minimizing the KL divergence between the variational distribution and the posterior. Once fit, the variational distribution is used as an approximation to the posterior.

Optimization of Equation 5.2 proceeds by coordinate ascent, optimizing each variational parameter while holding the others fixed. Each pass through the variational parameters increases the ELBO, and we iterate this process until reaching a local optimum. When possible, we find the per-parameter maximum value in closed form. When such updates are not possible, we employ gradient-based optimization (Galassi et al., 2003).

One can divide the ELBO into document terms and global terms. The document terms reflect the variational parameters of a single document and the global terms reflect variational parameters which are shared across all documents. This can be seen

in the plate notion in Figure 5.4; the variational parameters on the right hand side are specific to individual documents. We expand Equation 5.2 and divide it into a document component (Equation 7.9) and a global component (Equation 7.11), which contains a sum of all the document contributions, in the appendix.

In coordinate ascent, the global parameters are fixed as we optimize the document level parameters. Thus, we can optimize a single document's contribution to the ELBO ignoring all other documents. This allows us to parallelize our implementation at the document level; each parallel document-level optimization is followed by an optimization step for the global variational parameters. We iterate these steps until we find a local optimum. In practice, several random starting points are used and we select the variational parameters that reach the best local optimum.

In the next sections, we outline the variational updates for the word-specific terms, document-specific terms, and corpus-wide terms. This exposition preserves the parallelization in our implementation and highlights the separate influences of topic modeling and syntactic models.

Document-specific Terms

We begin with $\boldsymbol{\phi}_{d,n}$, the variational parameter that corresponds to the nth observed word's assignment to a topic. We can explicitly solve for the value of $\boldsymbol{\phi}_n$ which maximizes document d's contribution to the ELBO:

$$\begin{aligned}\phi_{n,i} \propto \exp \Bigg\{ & \Psi\left(\gamma_i\right) - \Psi\left(\textstyle\sum_{j=1}^{K} \gamma_j\right) + \sum_{j=1}^{K} \phi_{p(n),j} \left(\Psi\left(\nu_{j,i}\right) - \Psi\left(\textstyle\sum_{k=1}^{K} \nu_{j,k}\right)\right) \\ & - \sum_{c \in c(n)} \omega_c^{-1} \sum_{j}^{K} \frac{\gamma_j \nu_{i,j}}{\sum_k \gamma_k \sum_k \nu_{i,k}} \\ & + \sum_{c \in c(n)} \sum_{j=1}^{K} \phi_{c,j} \left(\Psi\left(\nu_{i,j}\right) - \Psi\left(\textstyle\sum_{k=1}^{K} \nu_{i,k}\right)\right) + \log \beta_{i,w_{d,n}} \Bigg\}. \end{aligned} \tag{5.3}$$

, where the function $p(n)$ gives the index of the parent of node n, which reflects the top-down generative process. (Note that we have suppressed the document index d on ϕ and γ.)

This update reveals the influences on our estimate of the posterior of a single word's topic assignment. In the first line, the first two terms with the Dirichlet parameter $\boldsymbol{\gamma}$ show the influence of the document's distribution over topics; the term with multinomial parameter $\boldsymbol{\phi}_{p(n)}$ and Dirichlet parameter $\boldsymbol{\nu}$ reflects the interaction between the topic of the parent and transition probabilities. In the second line, the interaction between the document and transitions forces the document and syntax to be consistent (this is mediated by an additional variational parameter ω_c discussed in Appendix 7.2.2). In the final line, the influence of the children's' topic on the current word's topic is expressed in the first term, and the probability of a word given a topic in the second.

The other document-specific term is the per-document variational Dirichlet over topic proportions $\boldsymbol{\gamma}_d$. Intuitively, topic proportions should reflect the expected number of words assigned to each topic in a document (the first two terms of equation 5.4), with the constraint that γ must be consistent with the syntactic transitions in the document, which is reflected by the ν term (the final term of Equation 5.4). This interaction prevents us from performing the update directly, so we use the gradient (derived in Appendix 5.3.2)

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \gamma_i} &= \Psi'(\gamma_i)\left(\alpha_{D,i}\tau^* + \sum_{n=1}^{N}\phi_{n,i} - \gamma_i\right) - \Psi'\left(\textstyle\sum_{j=1}^{N}\gamma_j\right)\displaystyle\sum_{j=1}^{K}\left[\alpha_{D,j}\tau^* + \sum_{n=1}^{N}\phi_{n,j} - \gamma_j\right] \\ &\quad - \sum_{n=1}^{N}\omega_n^{-1}\sum_{j=1}^{K}\left[\phi_{p(n),j}\frac{\nu_{j,i}\sum_{k\neq j}^{N}\gamma_k - \sum_{k\neq j}^{N}\nu_{j,k}\gamma_k}{\left(\sum_{k=1}^{N}\gamma_k\right)^2\sum_{k=1}^{N}\nu_{j,k}}\right] \end{aligned} \tag{5.4}$$

to optimize a document's ELBO contribution using numerical methods.

Now we turn to updates which require input from all documents and cannot be

parallelized. Each document optimization, however, produces expected counts which are summed together; this is similar to the how the the E-step of EM algorithms can be parallelized and summed as input to the M-step (Wolfe et al., 2008).

Global Variational Terms

In this section, we consider optimizing the variational parameters for the transitions between topics and the top-level topic weights. Note that these variational parameters, in contrast with the previous section, are more concerned with the overall syntax, which is shared across all documents. Instead of optimizing a single ELBO term for each document, we now seek to maximize the entirety of Equation 7.7, expanded in Equation 7.11 in the appendix.

The non-parametric models in Section 5.2.2 use a random variable τ drawn from a stick-breaking distribution to control how many components the model uses. The prior for τ attempts use as few topics as possible; the ELBO balances this desire against using more topics to better explain the data. We use numerical methods to optimize τ with respect to the gradient of the global ELBO, which is given in Equation 7.12 in the appendix.

Finally, we optimize the variational distribution ν_i. If there were no interaction between θ and π, the update for $\nu_{i,j}$ would be proportional to the expected number of transitions from parents of topic i to children of topic j (this will set the first two terms of Equation 5.5 to zero). However, the objective function also encourages ν to be consistent with γ (the final term of Equation 5.5); thus, if γ excludes topics from being observed in a document, the optimization will not allow transitions to those topics. Again, this optimization is done using numerical optimization using the

gradient of the ELBO,

$$\begin{aligned}\frac{\partial L}{\partial \nu_{i,j}} &= \Psi'(\nu_{i,j})\left(\alpha_{P,j} + \sum_{n=1}^{N}\sum_{c\in c(n)}\phi_{n,i}\phi_{c,j} - \nu_{i,j}\right) \\ &\quad -\Psi'\left(\textstyle\sum_{k=1}^{K}\nu_{i,k}\right)\displaystyle\sum_{k=1}^{K}\left[\alpha_{P,k} + \sum_{n=1}^{N}\sum_{c\in c(n)}\phi_{n,i}\phi_{c,k} - \nu_{i,k}\right] \\ &\quad -\sum_{n}^{N}\phi_{n,i}\sum_{c\in c(n)}\left[\omega_c^{-1}\frac{\gamma_j\sum_{k\neq j}^{N}\nu_{i,k} - \sum_{k\neq j}^{N}\nu_{i,k}\gamma_k}{\left(\sum_{k=1}^{N}\nu_{j,k}\right)^2\sum_{k=1}^{N}\gamma_k}\right]. \end{aligned} \tag{5.5}$$

5.4 Experiments

We demonstrate how the STM works on data sets of increasing complexity. First, we show that the STM captures properties of a simple synthetic dataset that elude both topic and syntactic models individually. Next, we use a larger real-word dataset of hand-parsed sentences to show that both thematic and syntactic information is captured by the STM.

5.4.1 Topics Learned from Synthetic Data

We demonstrate the STM on synthetic data that resemble natural language. The data were generated using the grammar specified in Table 5.1 (for a review of the context free formalism, see Section 1.2.1). Each of the parts of speech except for prepositions and determiners was divided into themes, and a document contains a single theme for each part of speech. For example, a document can only contain nouns from a single "economic," "academic," or "livestock" theme, verbs from a possibly different theme, etc. Documents had between twenty and fifty sentences. An example of two documents is shown in Figure 5.5.

Using a truncation level of 16, we fit three different non-parametric Bayesian

Fixed Syntax

S	→	VP
VP	→	NP V (PP) (NP)
NP	→	(Det) (Adj) N (PP)
PP	→	P NP
P	→	(“about”, “on”, “over”, “with”)
Det	→	(“a”, “that”, “the”, “this”)

Document-specific Vocabulary

V	→	(“falls”, “runs”, “sits”)	**or**
		(“bucks”, “climbs”, “falls”, “surges”)	...
N	→	(“COW”, “PONY”, “SHEEP”)	**or**
		(“MUTUAL_FUND”, “SHARE”, “STOCK”)	...
Adj	→	(“American”, “German”, “Russian”)	**or**
		(“blue”, “purple”, “red”, “white”)	...

Table 5.1: The procedure for generating synthetic data. Syntax is shared across all documents, but each document chooses one of the thematic terminal distribution for verbs, nouns, and adjectives. This simulates how all documents share syntax and subsets of documents share topical themes. All expansion rules are chosen uniformly at random.

language models to the synthetic data (Figure 5.6).[6] Because the infinite tree model is aware of the tree structure but not documents, it is able to separate all parts of speech successfully except for adjectives and determiners (Figure 5.6c). However, it ignores the thematic distinctions that divided the terms between documents. The HDP is aware of document groupings and treats the words exchangeably within them and is thus able to recover the thematic topics, but it misses the connections between the parts of speech, and has conflated multiple parts of speech (Figure 5.6b).

The STM is able to capture the topical themes and recover parts of speech (except prepositions placed in the same topic as nouns with a self loop). Moreover, it was able to identify the same interconnections between latent classes that were apparent from the infinite tree. Nouns are dominated by verbs and prepositions, and verbs are the

[6]In Figure 5.6 and Figure 5.7, we mark topics which represent a single part of speech and are essentially the lone representative of that part of speech in the model. This is a subjective determination of the authors, does not reflect any specialization or special treatment of topics by the model, and is done merely for didactic purposes.

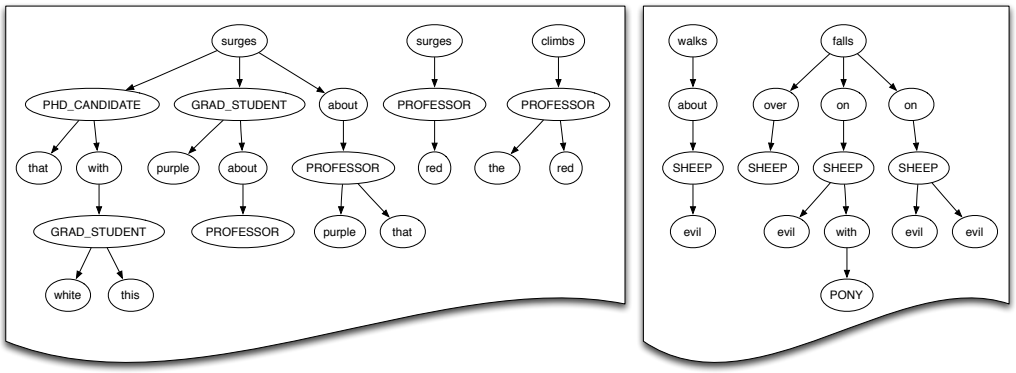


Figure 5.5: Two synthetic documents with multiple sentences. Nouns are upper case. Each document chooses a theme for each part of speech independently; for example, the document on the left uses motion verbs, academic nouns, and color adjectives. Various models are applied to these data in Figure 5.6.

root (head) of sentences. Figure 5.6d shows the two divisions as separate axes; going form left to right, the thematic divisions that the HDP was able to uncover are clear. Going from top to bottom, the syntactic distinctions made by the infinite tree are revealed.

5.4.2 Qualitative Description of Topics learned by the STM from Hand-annotated Data

The same general properties, but with greater variation, are exhibited in real data. We converted the Penn Treebank (Marcus et al., 1994), a corpus of manually curated parse trees, into a dependency parse (Johansson and Nugues, 2007). The vocabulary was pruned to terms that appeared in at least ten documents.

Figure 5.7 shows a subset of topics learned by the STM with truncation level 32. Many of the resulting topics illustrate both syntactic and thematic consistency. A few non-specific function topics emerged (pronoun, possessive pronoun, general verbs, etc.). Many of the noun categories were more specialized. For instance, Figure 5.7 shows clusters of nouns relating to media, individuals associated with companies

("mr," "president," "chairman"), and abstract nouns related to stock prices ("shares," "quarter," "earnings," "interest"), all of which feed into a topic that modifies nouns ("his," "their," "other," "last").

Griffiths et al (Griffiths et al., 2005) observed that nouns, more than other parts of speech, tend to specialize into distinct topics, and this is also evident here. In Figure 5.7, the unspecialized syntactic categories (shaded and with rounded edges) serve to connect many different specialized thematic categories, which are predominantly nouns (although the adjectives also showed bifurcation). For example, verbs are mostly found in a single topic, but then have a large number of outgoing transitions to many noun topics. Because of this relationship, verbs look like a syntactic "source" in Figure 5.7. Many of these noun topics then point to thematically unified topics such as "personal pronouns," which look like syntactic "sinks."

It is important to note that Figure 5.7 only presents half of the process of choosing a topic for a word. While the transition distribution of verb topics allows many different noun topics as possible dependents, because the topic is chosen from a product of $\boldsymbol{\theta}$ and $\boldsymbol{\pi}$, $\boldsymbol{\theta}$ can filter out the noun topics that are inconsistent with a document's theme.

This division between functional and topical uses for the latent classes can also been seen in the values for the per-document multinomial over topics. Some topics in Figure 5.7(b), such as 17, 15, 10, and 3, appear to some degree in nearly every document, while other topics are used more sparingly to denote specialized content. With $\alpha = 0.1$, this plot also shows that the non-parametric Bayesian framework is ignoring many later topics.

5.4.3 Quantitative Results on Synthetic and Hand-annotated Data

To study the performance of the STM on new data, we estimated the held out probability of previously unseen documents with an STM trained on a portion of

the dataset. For each position in the parse trees, we estimate the probability of the observed word. We compute the perplexity as the exponent of the inverse of the per-word average log probability. The lower the perplexity, the better the model has captured the patterns in the data. We also computed perplexity for individual parts of speech to study the differences in predictive power between content words, such as nouns and verbs, and function words, such as prepositions and determiners. This illustrates how different algorithms better capture aspects of context. We expect function words to be dominated by local context and content words to be determined more by the themes of the document.

This trend is seen not only in the synthetic data (Figure 5.8(a)), where syntactic models better predict functional categories like prepositions, and document-only models fail to account for patterns of verbs and determiners, but also in real data. Figure 5.8(b) shows that HDP and STM both perform better than syntactic models in capturing the patterns behind nouns, while both STM and the infinite tree have lower perplexity for verbs. Like syntactic models, our model was better able to predict the appearance of prepositions but also remained competitive with HDP on content words. On the whole, STM had lower perplexity than HDP and the infinite tree.

5.5 Conclusion

In this chapter, we explored the common threads that link syntactic and topic models and created a model that is simultaneously aware of both thematic and syntactic influences in a document. These models are aware of more structure than either model individually.

More generally, the STM serves as an example of how a mixture model can support two different, simultaneous explanations for how the latent class is chosen. Although this model used discrete observations, the variational inference setup is flexible enough

to support other distributions over the output.

While the STM's primary goal was to demonstrate how these two views of context could be simultaneously learned, there are a number of extensions that could lead to more accurate parsers. First, this model could be further extended by integrating a richer syntactic model that does not just model the words that appear in a given structure but one that also models the parse structure itself. This would allow the model to use large, diverse corpora without relying upon an external parser to provide the tree structure.

Removing the independence restriction between children also would allow for this model to closer approximate the state of the art syntactic models and to be better distinguish the children of parent nodes (this is especially the problem for head verbs, which often have many children). Finally, this model could also make use of labeled dependency relations and lexicalization.

With the ability to adjust to specific document or corpus-based contexts, a parser built using this framework could adapt to handle different domains while still sharing information between them. The classification and clustering implicitly provided by the topic components would allow the parser to specialize its parsing model when necessary, allowing both sentence-level and document-level information to shape the model's understanding of a document.

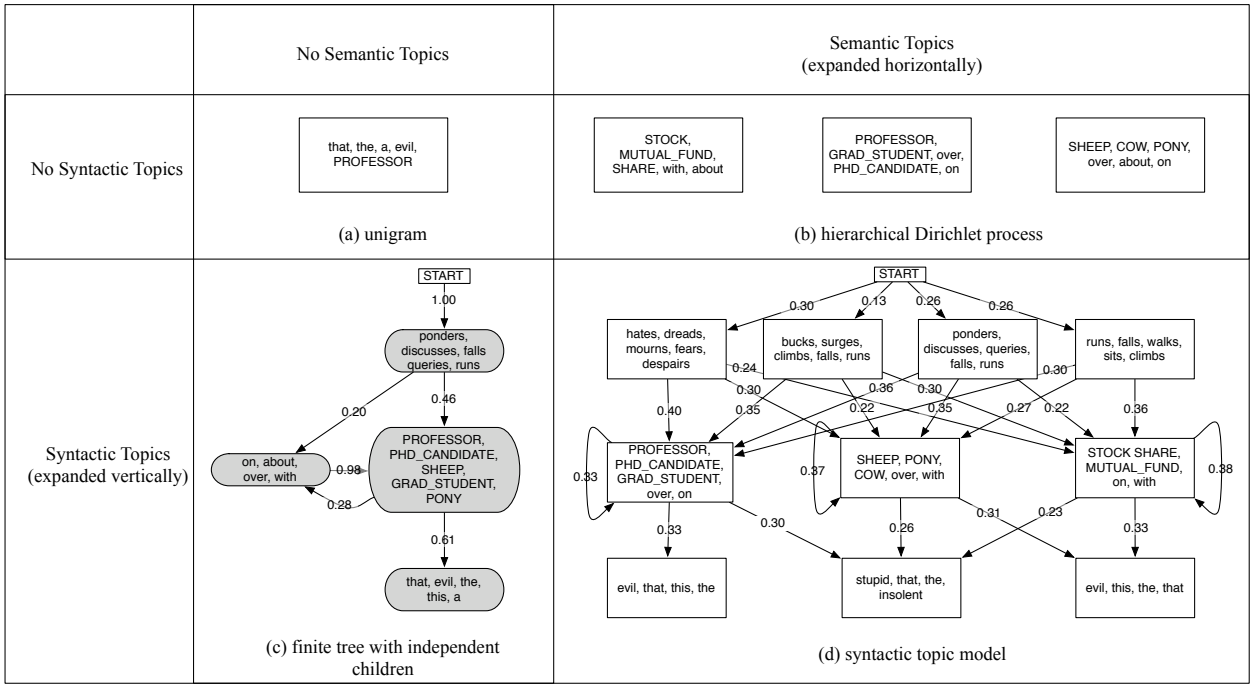
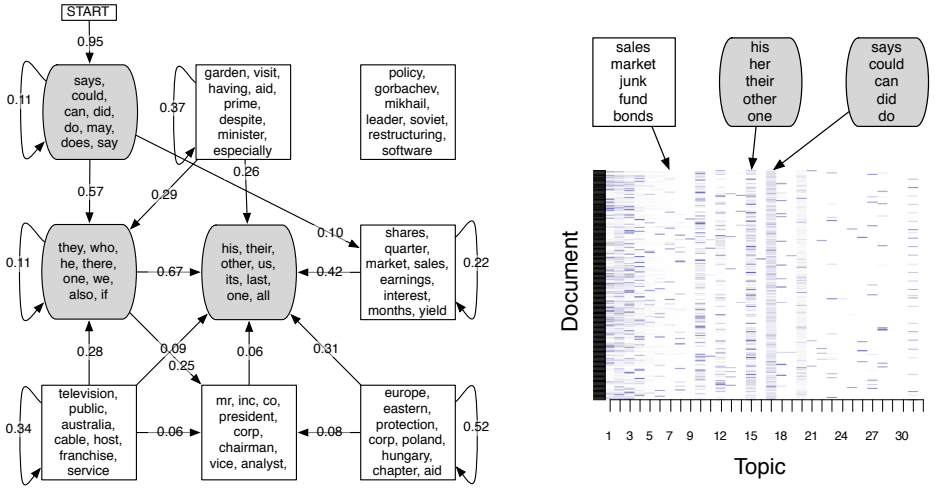


Figure 5.6: We contrast the different views of data that are available by using syntactic and semantic topics based on our synthetic data. Three models were fit to the synthetic data described in Section 5.4. Each box illustrates the top five words of a topic; boxes that represent homogeneous parts of speech have rounded edges and are shaded; and nouns are in upper case. Edges between topics are labeled with estimates of their transition weight π. If we have neither syntactic nor semantic topics, we have a unigram (a) model that views words as coming from a single distribution over words. Adding syntactic topics allows us to recover the parts of speech (c), but this lumps all topics together. Although the HDP (b) can discover themes of recurring words, it cannot determine the interactions between topics or separate out ubiquitous words that occur in all documents. The STM (d) is able to recover both the syntax and the themes.



(a) learned topics (b) topic prevalence

Figure 5.7: Topics discovered from fitting the syntactic topic model on the Treebank corpus (a, left). As in Figure 5.6, parts of speech that aren't subdivided across themes are indicated and edges between topics are labeled with estimates of the transition probability π. Head words (verbs) are shared across many documents and allow many different types of nouns as possible dependents. These dependents, in turn, share topics that look like pronouns as common dependents. The specialization of topics is also visible in plots of the estimates for the per-document topic distribution θ for the first 300 documents of the Treebank (b, right), where three topics columns have been identified. Many topics are used to some extent in every document, showing that they are performing a functional role, while others are used more sparingly for semantic content.

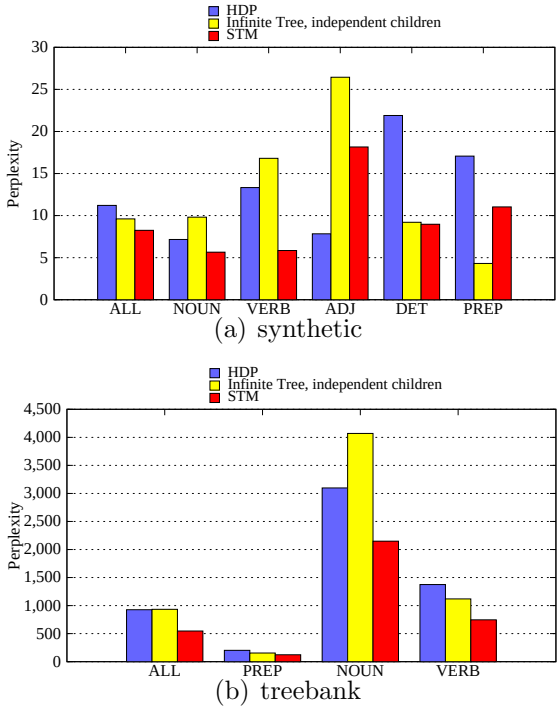


Figure 5.8: After fitting three models on synthetic data, the syntactic topic model has better (lower) perplexity on all word classes except for adjectives. HDP is better able to capture document-level patterns of adjectives. The infinite tree captures prepositions best, which have no cross-document variation. On real data 5.8(b), the syntactic topic model was able to combine the strengths of the infinite tree on functional categories like prepositions with the strengths of the HDP on content categories like nouns to attain lower overall perplexity.

Chapter 6

Conclusion and Future Work

Latent Dirichlet allocation (LDA) is a modern, ubiquitous technique for the unsupervised discovery of structure in data. However, despite its popularity, its assumptions about the nature of the data are linguistically uninformed, even though it is usually applied to text.

It assumes that documents are a bag of words, assumes that related words are no more likely to appear together than unrelated words, and cannot understand multilingual corpora. In the preceding chapters, we showed extensions that extend LDA to overcome these limitations using tools and resources from the linguistics community.

6.1 Building on Linguistic Data

One of the recurring themes in this thesis is the use of linguistic resources. These resources are used as a starting point and as a tool to guide our statistical approaches to find explanatory patterns that are consistent with both the foundational linguistic resources and the data we observe in the real world.

In Chapter 2, we proposed latent Dirichlet allocation with WordNet, LDAWN, a model that uses an ontology to encourage words with similar meanings to have

correlated probabilities within an individual topic. This creates topics that are consistent with the initial ontology, but it only creates topics that are also consistent with the documents that we observe in a corpus. In Chapter 3, we extended LDAWN to multiple languages.

In Chapter 4, we took the idea of combining information from curated linguistic resources a step further. The multilingual topic model for unaligned text (MuTo) uses resources like dictionaries to form an initial bridge between languages and then iteratively improves the connections between languages to build topics that make sense across languages.

Finally, in Chapter 5, we developed a technique that combined two views of text data: the view provided by syntactic models and the view provided by probabilistic topic models. Again, we use curated linguistic data as a starting point; our data are a collection of dependency trees split into documents. Using these data, we discover patterns of words that are consistent with both local syntactic context and global semantic patterns based on document co-occurrence.

As we discussed in Chapter 1, linguistics is deep and productive. Many theories from linguistics have been embodied in machine readable datasets or annotated corpora. While these resources are not without their flaws (e.g., Section 2.4), the intuitions and insight in such resources can guide statistical models to discover patterns that are consistent with data and human understanding.

6.2 Deeper Linguistic Models and New Applications

We were able to combine the insights provided by linguistic resources with the statistical formalism of topic models by first specifying a statistical model consistent with the linguistic information. For LDAWN, it was a probabilistic walk through a tree; for

MuTo, it was a matching over terms across languages; for the STM, it was the Infinite Tree with Independent Children (Finkel et al., 2007). In each case, we combined this statistical model of the linguistic information with the basic skeleton of LDA.

The ability to combine models so seamlessly is one of the strengths of statistical models. Statistical models speak the same language—probability—so combining them requires only specifying a composite model and then deriving inference for the new model.

While these models are the product of combination, they could themselves serve as components in a larger model. For example: the syntax modeling of the STM in Chapter 5 could be combined with the ontologies of LDAWN 2 to create context-specific disambiguations; the matching of MuTo could be combined with LDAWN to create an ad hoc alignment over paths in unaligned WordNets; or the syntax modeling of the STM could be combined with MuTo to learn translations that also depend on local context but still not requiring an explicit parallelism at the local syntax level.

This thesis show that LDA can be extended to draw upon linguistic insights, but there are many other possible applications that enable linguistically sound applications and also applications that use the insights presented here to explore applications beyond text.

6.2.1 Capturing Other Knowledge Sources

In Chapter 2, we demonstrated a method that allowed knowledge about word meanings encoded in an ontology to be incorporated into a probabilistic model in such a way that if concepts had similar meanings in an ontology, the words that express those concepts would have correlated probabilities. Other knowledge resources are organized in hierarchies similar to WordNet: elements are organized in a tree, elements can appear in ambiguous ways, and identifying the path in the hierarchy for a mention of

an element disambiguates the mention:

locations Gazetteers organize locations in hierarchies such as borough, city, county, state, and nation. These locations are mentioned in documents, but which location corresponds to which reference is uncertain.

genes Genes have been organized into hierarchies where subtrees share common localization (Ashburner, 2000). Treating pathways as documents could help determine where in a cell a particular interaction happens.

named entities Wikipedia categorizes named entities into (roughly) hierarchical categories. For example, Albert Einstein is categorized as a "Jewish Scientist," an "American Jewish Scientist," and as a "Jewish Scientist." However, the string "Einstein" could refer to the person, a medical school, or a bagel company. Organizing mentions into this hierarchy could discover correlations between categories that appear in documents.

6.2.2 Integrating Models into Applications

The models developed here focused on the representation and modeling challenges more than actual applications. However, because of the flexibility of probabilistic models, it is relatively simple to use the models in this thesis for the same applications that have been developed for other topic models.

For instance, supervised topic models (Blei and McAuliffe, 2007) and relational topic models (Chang and Blei, 2009) use the per-document topic distribution to make predictions a document's sentiment or connection to other documents. The models presented in Chapter 4 would allow these predictions to be made on multilingual corpora. For instance, instead of just making predictions based on reviews on Amazon.com's English website, predictions could be also share information gleaned from reviews on Amazon.com's Japanese and German language sites.

6.2.3 Learning Deeper Structures and Testing Cognitive Plausibility

The most exciting extensions of these models come from the ability to use the insights from linguistic information while not relying exclusively on the limited amount of data available from a single set of trained expert annotations from linguists. In Chapter 5, we relied on a corpus meticulously turned into machine readable parses by human annotators. Discovering this structure using unlabeled or partially labeled data would help increase the applicability of the methods discussed here.

Understanding how humans produce and understand language is the central question in linguistics. Allowing computers to reach the same level of understanding requires the synthesis of many of the insights and resources created by the linguistics community. Expressing these insights and resources in the language of probabilistic models makes them understandable to computers, makes them easier to test and combine with other methodologies, and makes them able to react and grow as more data are presented.

This thesis takes linguistic notions of semantics and syntax and casts them in a probabilistic framework to understand data through the framework of topic models. This is a step in building models that can process large amounts of data, a strength of probabilistic models, but can also retain the lessons learned from the knowledge and experience of linguists.

All models make assumptions, but doing so in a linguistically-grounded way means that as we explore and use these models more and more, how our models provide insight into not just the mechanics of how we engineered our particular models but also into the rich cognitive and and philosophical assumptions they inherit.

Bibliography

Abney, S. (2004). Understanding the Yarowsky Algorithm. *Computational Linguistics*, 30(3):365–395.

Abney, S. and Light, M. (1999). Hiding a semantic hierarchy in a Markov model. In *Proceedings of the Workshop on Unsupervised Learning in Natural Language Processing*, pages 1–8.

Allan, K. (2001). *Natural Language Semantics*. Wiley-Blackwell, Malden.

Antoniak, C. E. (1974). Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems. *The Annals of Statistics*, 2(6):1152–1174.

Ashburner, M. (2000). Gene ontology: Tool for the unification of biology. *Nature Genetics*, 25:25–29.

Banko, M., Cafarella, M. J., Soderland, S., Broadhead, M., and Etzioni, O. (2007). Open information extraction from the web. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Blei, D. M., Griffiths, T. L., and Jordan, M. I. (2007). The nested chinese restaurant process and hierarchical topic models.

Blei, D. M. and Jordan, M. I. (2003). Modeling annotated data. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research*

and development in informaion retrieval, pages 127–134, New York, NY, USA. ACM Press.

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

Blei, D. M. and Lafferty, J. (2009). *Text Mining: Theory and Applications*, chapter Topic Models. Taylor and Francis, London.

Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of International Conference of Machine Learning*, pages 113–120, New York, NY, USA. ACM Press.

Blei, D. M. and McAuliffe, J. D. (2007). Supervised topic models. In *Advances in Neural Information Processing Systems*. MIT Press.

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

Blunsom, P., Cohn, T., and Osborne, M. (2008). Bayesian synchronous grammar induction. In *Proc of NIPS*.

Boyd-Graber, J. and Blei, D. (2008). Syntactic topic models. In *Advances in Neural Information Processing Systems*.

Boyd-Graber, J. and Blei, D. M. (2007). PUTOP: Turning predominant senses into a topic model for WSD. In *4th International Workshop on Semantic Evaluations*.

Boyd-Graber, J. and Blei, D. M. (2009). Multilingual topic models for unaligned text. In *Uncertainty in Artificial Intelligence*.

Boyd-Graber, J., Blei, D. M., and Zhu, X. (2007). A topic model for word sense disambiguation. In *Empirical Methods in Natural Language Processing*.

Boyd-Graber, J., Fellbaum, C., Osherson, D., and Schapire, R. (2006a). Adding dense, weighted, connections to WordNet. In Sojka, P., Choi, K.-S., Fellbaum, C., and Vossen, P., editors, *Proc. Global WordNet Conference 2006*, pages 29–35, Brno, Czech Republic. Global WordNet Association, Masaryk University in Brno.

Boyd-Graber, J., Nikolova, S. S., Moffatt, K. A., Kin, K. C., Lee, J. Y., Mackey, L. W., Tremaine, M. M., and Klawe, M. M. (2006b). Participatory design with proxies: Developing a desktop-PDA system to support people with aphasia. In *Computer-Human Interaction*.

Boyd-Graber, J. and Resnik, P. (2010). Holistic sentiment analysis across languages: Multilingual supervised latent dirichlet allocation. In *Empirical Methods in Natural Language Processing*.

Brody, S. and Lapata, M. (2009). Bayesian word sense induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 103–111, Athens, Greece. Association for Computational Linguistics.

Buitelaar, P. and Sacaleanu, B. (2001). Ranking and selecting synsets by domain relevance. In *Proceedings of WordNet and Other Lexical Resources: Applications, Extensions and Customizations. NAACL 2001.* Association for Computational Linguistics.

Cai, J. F., Lee, W. S., and Teh, Y. W. (2007). NUS-ML:Improving word sense disambiguation using topic features. In *Proceedings of SemEval-2007*. Association for Computational Linguistics.

Carroll, S. E. (1992). On cognates. *Second Language Research*, 8(2):93–119.

Chang, J. and Blei, D. M. (2009). Relational topic models for document networks. In *Proceedings of Artificial Intelligence and Statistics*.

Chang, J., Boyd-Graber, J., and Blei, D. M. (2009a). Connections between the lines: Augmenting social networks with text. In *Refereed Conference on Knowledge Discovery and Data Mining*.

Chang, J., Boyd-Graber, J., Wang, C., Gerrish, S., and Blei, D. M. (2009b). Reading tea leaves: How humans interpret topic models. In *Neural Information Processing Systems*.

Chang, J., Boyd-Graber, J., Wang, C., Gerrish, S., and Blei, D. M. (2009c). Reading tea leaves: How humans interpret topic models. In *Neural Information Processing Systems*.

Charniak, E. (1997). Statistical techniques for natural language parsing. *AI Magazine*, 18:33–44.

Charniak, E. (2000). A maximum-entropy-inspired parser. In *Proceedings of the North Americal Association for Computational Linguistics*, pages 132–139.

Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on Information Theory*, 2:113–124.

Chomsky, N. and Miller, G. A. (1958). Finite state languages. *Information and Control*, 1(2):91–112.

Ciaramita, M. and Johnson, M. (2000). Explaining away ambiguity: Learning verb selectional preference with bayesian networks. In *COLING-00*, pages 187–193.

Collins, M. (2003). Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637.

Dapretto, M. and Bookheimer, S. Y. (1999). Form and content: Dissociating syntax and semantics in sentence comprehension. *Neuron*, 24(2):427–432.

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

Diebolt, J. and Ip, E. H. (1996). *Markov Chain Monte Carlo in Practice*, chapter Stochastic EM: method and application. Chapman and Hall, London.

Eisner, J. (1996). Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96)*, pages 340–345, Copenhagen.

Escobar, M. D. and West, M. (1995). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90:577–588.

Fergus, R., Fei-Fei, L., Perona, P., and Zisserman, A. (2005). Learning object categories from google's image search. In *Proceedings of the 10th International Conference on Computer Vision, Beijing, China*, volume 2, pages 1816–1823.

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

Finkel, J. R., Grenager, T., and Manning, C. D. (2007). The infinite tree. In *Proceedings of Association for Computational Linguistics*, pages 272–279, Prague, Czech Republic. Association for Computational Linguistics.

Fung, P. and Yee, L. Y. (1998). An ir approach for translating new words from nonparallel, comparable texts. In *Proceedings of the 17th international conference on Computational linguistics*, pages 414–420, Morristown, NJ, USA. Association for Computational Linguistics.

Galassi, M., Davies, J., Theiler, J., Gough, B., Jungman, G., Booth, M., and Rossi, F. (2003). *Gnu Scientific Library: Reference Manual*. Network Theory Ltd.

Geyken, A. (2007). The DWDS corpus: A reference corpus for the German language of the 20th century. In *Idioms and Collocations: Corpus-based Linguistic, Lexicographic Studies*. Continuum Press. Im Erscheinen.

Geyken, A. and Boyd-Graber, J. (2003). Automatic classification of multi-word expressions in print dictionaries. *Linguisticae Investigationes*, 26(2).

Goldwater, S. (2007). *Nonparametric Bayesian Models of Lexical Acquisition*. PhD thesis, Brown University.

Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, pages 5228–5235.

Griffiths, T. L. and Steyvers, M. (2006). Probabilistic topic models. In Landauer, T., McNamara, D., Dennis, S., and Kintsch, W., editors, *Latent Semantic Analysis: A Road to Meaning*. Laurence Erlbaum.

Griffiths, T. L., Steyvers, M., Blei, D. M., and Tenenbaum, J. B. (2005). Integrating topics and syntax. In Saul, L. K., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems*, pages 537–544. MIT Press, Cambridge, MA.

Griffiths, T. L., Steyvers, M., and Tenenbaum, J. (2007). Topics in semantic representation. *Psychological Review*, 114(2):211–244.

Gruber, A., Rosen-Zvi, M., and Weiss, Y. (2007). Hidden topic Markov models. In *Artificial Intelligence and Statistics*.

Haghighi, A., Liang, P., Berg-Kirkpatrick, T., and Klein, D. (2008). Learning bilingual lexicons from monolingual corpora. In *Proceedings of ACL-08: HLT*, pages 771–779, Columbus, Ohio. Association for Computational Linguistics.

Hajič, J. (1998). Building a Syntactically Annotated Corpus: The Prague Dependency Treebank. In Hajičová, E., editor, *Issues of Valency and Meaning. Studies in Honor of Jarmila Panevová*, pages 12–19. Prague Karolinum, Charles University Press.

Hall, D., Jurafsky, D., and Manning, C. D. (2008). Studying the history of ideas using topic models. In *Proceedings of Emperical Methods in Natural Language Processing*, pages 363–371. ACL.

Hamp, B. and Feldweg, H. (1997). GermaNet – a lexical–semantic net for German. In *In Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, pages 9–15.

Hays, D. G. (1964). Dependency theory: A formalism and some observations. *Language*, 40:511–525.

Hinton, G. (1999). Products of experts. In *Proceedings of the Ninth International Conference on Artificial Neural Networks*, pages 1–6, Edinburgh, Scotland. IEEE.

Hofmann, T. (1999). Probabilistic latent semantic analysis. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, Stockholm.

Hu, D. and Saul, L. K. (2009). A probabilistic model of unsupervised learning for musical-key profiles. In *International Society for Music Information Retrieval*.

Information Extraction and Synthesis Laboratory (2009). Rexa.

Isahara, H., Bond, F., Uchimoto, K., Utiyama, M., and Kanzaki, K. (2008). Development of the japanese wordnet. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odjik, J., Piperidis, S., and Tapias, D., editors, *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Jiang, J. J. and Conrath, D. W. (1997). Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*, Taiwan.

Johansson, R. and Nugues, P. (2007). Extended constituent-to-dependency conversion for English. In *Proceedings of the Nordic Conference on Computational Linguistics*.

Johnson, M. (2009). Grammars and topic models. Webpage.

Johnson, M., Griffiths, T. L., and Goldwater, S. (2006). Adaptor grammars: A framework for specifying compositional nonparametric Bayesian models. In *Advances in Neural Information Processing Systems*.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233.

Kilgarriff, A. (2000). Review of WordNet : An electronic lexical database. *Language*, (76):706–708.

Kilgarriff, A. and Rosenzweig, J. (2000). Framework and results for english senseval.

Kim, W. and Khudanpur, S. (2004). Lexical triggers and latent semantic analysis for cross-lingual language model adaptation. *ACM Transactions on Asian Language Information Processing (TALIP)*, 3(2):94–112.

Klein, D. and Manning, C. D. (2002). Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems 15 (NIPS 2002)*.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of Association for Computational Linguistics*, pages 423–430. Association for Computational Linguistics.

Knight, K. and Graehl, J. (1997). Machine transliteration. In Cohen, P. R. and Wahlster, W., editors, *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 128–135, Somerset, New Jersey. Association for Computational Linguistics.

Koehn, P. (2000). German-english parallel corpus "de-news".

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.

Koehn, P. and Knight, K. (2002). Learning a translation lexicon from monolingual corpora. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 9–16, Morristown, NJ, USA. Association for Computational Linguistics.

Kübler, S., McDonald, R., and Nivre, J. (2009). *Dependency Parsing*. Morgan and Claypool.

Kunze, C. and Lemnitzer, L. (2002). Standardizing wordnets in a web-compliant format: The case of germanet. In Christodoulakis, D., Kunze, C., and Lemnitzer, L., editors, *Proceedings of the Workshop on Wordnets Structures and Standardisation, and how these Affect Wordnet Applications and Evaluation*, pages 24–29.

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

Landauer, T. and Dumais, S. (1997). Solutions to Plato's problem: The latent semantic analsyis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, (104).

Lawler, E. (1976). *Combinatorial optimization - networks and matroids*. Holt, Rinehart and Winston, New York.

Li Fei-Fei and Perona, P. (2005). A Bayesian hierarchical model for learning natural scene categories. In *CVPR '05 - Volume 2*, pages 524–531, Washington, DC, USA. IEEE Computer Society.

Liang, P. and Klein, D. (2007). Structured Bayesian nonparametric models with variational inference (tutorial). In *Proceedings of Association for Computational Linguistics*.

Liang, P., Petrov, S., Jordan, M., and Klein, D. (2007). The infinite PCFG using hierarchical Dirichlet processes. In *Proceedings of Emperical Methods in Natural Language Processing*, pages 688–697.

Lin, D. (1998). An information-theoretic definition of similarity. In *Proceedings of International Conference of Machine Learning*, pages 296–304.

Ma, X., Boyd-Graber, J., Nikolova, S. S., and Cook, P. (2009). Speaking through pictures: Images vs. icons. In *ACM Conference on Computers and Accessibility*.

Magnini, B., Strapparava, C., Pezzulo, G., and Gliozzo, A. (2001). Using domain information for word sense disambiguation. In *In Proceedings of 2nd International Workshop on Evaluating Word Sense Disambiguation Systems*, Toulouse, France.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1994). Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19(2):313–330.

Maskeri, G., Sarkar, S., and Heafield, K. (2008). Mining business topics in source code using latent dirichlet allocation. In *ISEC '08: Proceedings of the 1st conference on India software engineering conference*, pages 113–120, New York, NY, USA. ACM.

McCarthy, D., Koeling, R., Weeds, J., and Carroll, J. (2004). Finding predominant word senses in untagged text. In *Proceedings of Association for Computational Linguistics*, pages 280–287. Association for Computational Linguistics.

McMahon, A. and McMahon, R. (2005). *Language Classification by Numbers*. Oxford University Press.

Mihalcea, R. (2005). Large vocabulary unsupervised word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the Joint Human Language Technology and Empirical Methods in Natural Language Processing Conference*, pages 411–418.

Mihalcea, R., Chklovsky, T., and Kilgarriff, A. (2004). The Senseval-3 English lexical sample task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 25–28.

Miller, G., Leacock, C., Tengi, R., and Bunker, R. (1993). A semantic concordance. In *3rd DARPA Workshop on Human Language Technology*, pages 303–308.

Miller, G. A. (1990). Nouns in WordNet: A lexical inheritance system. *International Journal of Lexicography*, 3(4):245–264.

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28.

Mimno, D. and McCallum, A. (2007). Mining a digital library for influential authors. In *JCDL '07: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 105–106, New York, NY, USA. ACM.

Mimno, D., Wallach, H., Naradowsky, J., Smith, D., and McCallum, A. (2009). Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 880–889, Singapore. ACL.

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

Neal, R. M. (2000). Markov chain sampling methods for Dirichlet process mixture models. *Journal of Computational and Graphical Statistics*, 9(2):249–265.

Ni, X., Sun, J.-T., Hu, J., and Chen, Z. (2009). Mining multilingual topics from Wikipedia. In *International World Wide Web Conference*, pages 1155–1155.

Nikolova, S. S., Boyd-Graber, J., and Cook, P. (2009a). The design of viva: A mixed-initiative visual vocabulary for aphasia. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pages 4015–4020, New York, NY, USA. ACM.

Nikolova, S. S., Boyd-Graber, J., and Fellbaum, C. (2011). *Collecting Semantic Similarity Ratings to Connect Concepts in Assistive Communication Tools*. Studies in Computational Intelligence. Springer Verlag, Heidelberg.

Nikolova, S. S., Boyd-Graber, J., Fellbaum, C., and Cook, P. (2009b). Better vocabularies for assistive communication aids: Connecting terms using semantic networks and untrained annotators. In *ACM Conference on Computers and Accessibility*.

Nivre, J. (2005). Dependency grammar and dependency parsing. Technical report, Växjö University: School of Mathematics and Systems Engineering.

Ordan, N. and Wintner, S. (2007). Hebrew WordNet: a test case of aligning lexical databases across languages. *International Journal of Translation, special issue on Lexical Resources for Machine Translation*, 19(1):39–58.

Padó, S. and Lapata, M. (2007). Dependency-based construction of semantic space models. *Computational Linguistics*, 33(2):161–199.

Patwardhan, S., Banerjee, S., and Pedersen, T. (2003). Using Measures of Semantic Relatedness for Word Sense Disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257.

Petrov, S. (2010). Products of random latent variable grammars. In *Proceedings of the North Americal Association for Computational Linguistics*.

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

Pritchard, J. K., Stephens, M., and Donnelly, P. (2000). Inference of population structure using multilocus genotype data. *Genetics*, 155:945–959.

Purver, M., Körding, K., Griffiths, T. L., and Tenenbaum, J. (2006). Unsupervised topic modelling for multi-party spoken discourse. In *Proceedings of Association for Computational Linguistics*.

Ramat, P. (1987). *Empirical approaches to language typology*. Mouton de Gruyter, Berlin.

Rapp, R. (1995). Identifying word translations in non-parallel texts. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 320–322, Morristown, NJ, USA. Association for Computational Linguistics.

Rayner, K., Carlson, M., and Frazier, L. (1983). The interaction of syntax and semantics during sentence processing — Eye-movements in the analysis of semantically biased sentences. *Journal of Verbal Learning and Verbal Behavior*, 22(3):358–374.

Resnik, P. (1995). Using information content to evaluate semantic similarity in a

taxonomy. In *International Joint Conferences on Artificial Intelligence*, pages 448–453.

Richter, F. (2008). Dictionary nice grep. In *http://www-user.tu-chemnitz.de/ fri/ding/*.

Robert, C. and Casella, G. (2004). *Monte Carlo Statistical Methods*. Springer Texts in Statistics. Springer-Verlag, New York, NY.

Rosen-Zvi, M., Griffiths, T. L., Steyvers, M., and Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494, Arlington, Virginia, United States. AUAI Press.

Sagot, B. and Fišer, D. (2008). Building a Free French WordNet from Multilingual Resources. In *OntoLex 2008*, Marrackech, Morocco.

Sapir, E. (1929). The status of linguistics as a science. *Language*, 5(4):207–214.

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*.

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

Sproat, R., Tao, T., and Zhai, C. (2006). Named entity transliteration with comparable corpora. In *ACL-44: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 73–80, Morristown, NJ, USA. Association for Computational Linguistics.

Tam, Y.-C. and Schultz, T. (2007). Bilingual lsa-based translation lexicon adaptation for spoken language translation. In *INTERSPEECH-2007*, pages 2461–2464.

Tanaka, K. and Iwasaki, H. (1996). Extraction of lexical translations from non-aligned corpora. In *Proceedings of the 16th conference on Computational linguistics*, pages 580–585, Morristown, NJ, USA. Association for Computational Linguistics.

Teh, Y. W. (2006). A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of Association for Computational Linguistics*.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2006). Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581.

Titov, I. and McDonald, R. (2008). A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, Ohio. Association for Computational Linguistics.

Toutanova, K. and Johnson, M. (2008). A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems 20*, pages 1521–1528. MIT Press, Cambridge, MA.

University of Oxford (2006). British National Corpus. http://www.natcorp.ox.ac.uk/.

Vossen, P., editor (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*. Springer.

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305.

Wallach, H. M. (2006). Topic modeling: beyond bag-of-words. In *Proceedings of International Conference of Machine Learning*, pages 977–984, New York, NY, USA. ACM.

Wang, C., Blei, D., and Fei-Fei, L. (2009). Simultaneous image classification and annotation. In *CVPR*.

Wang, C., Blei, D. M., and Heckerman, D. (2008). Continuous time dynamic topic models. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*.

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

Winn, J. (2003). Variational message passing and its applications.

Wolfe, J., Haghighi, A., and Klein, D. (2008). Fully distributed EM for very large datasets. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, pages 1184–1191, New York, NY, USA. ACM.

Yarowsky, D. (1995). Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196, Morristown, NJ, USA. Association for Computational Linguistics.

Zhai, K., Boyd-Graber, J., Asadi, N., and Alkhouja, M. (2012). Mr. LDA: A flexible large scale topic modeling package using variational inference in mapreduce. In *ACM International Conference on World Wide Web*.

Zhao, B. and Xing, E. P. (2006). Bitam: Bilingual topic admixture models for word alignment. In *Proceedings of Association for Computational Linguistics*, pages 969–976, Sydney, Australia. Association for Computational Linguistics.

Zhu, X., Blei, D. M., and Lafferty, J. (2006). TagLDA: Bringing document structure knowledge into topic models. Technical Report TR-1553, University of Wisconsin, Madison.

Chapter 7

Appendix: Variational Inference for Syntactic Topic Models

This appendix explains the derivation of the updates for variational inference for the Syntactic Topic Model (STM). After some mathematical preliminaries, we expand the expectations in the variational likelihood bound and then, having expanded the objective function, derive the updates which optimize the bound.

7.1 Dirichlet in the Exponential Family

A probability distribution is a member of the exponential family of distributions if it can be expressed using the exponential family form

$$p(x|\eta) = h(x)\exp\left\{g(\eta)^T u(x) - a(\eta)\right\}, \tag{7.1}$$

where $g(\eta)$ is the natural parameter vector, $u(x)$ is the natural statistic vector, $h(x)$ is the measure of the space, and $a(\eta)$ is the normalization. We can express the Dirichlet distribution (first discussed in Section 1.1) as an exponential family distribution,

rewriting the conventional density function,

$$\mathrm{Dir}(\boldsymbol{\theta}\Gamma\left(\alpha\right)_1,\ldots,\alpha_K) = \underbrace{\frac{\Gamma\left(\sum_k \alpha_k\right)}{\prod_k \Gamma\left(\alpha_k\right)}}_{\text{normalization}} \prod_k \theta_k^{\alpha_k - 1},$$

as an exponential family distribution

$$\mathrm{Dir}(\boldsymbol{\theta}|\alpha) = \exp\left\{ \begin{bmatrix} \alpha_1 - 1 \\ \vdots \\ \alpha_K - 1 \end{bmatrix}^T \begin{bmatrix} \log\theta_1 \\ \vdots \\ \log\theta_K \end{bmatrix} + \log\Gamma\left(\sum_{i=1}^K \alpha_i\right) - \sum_{i=1}^K \Gamma\left(\alpha_i\right) \right\}. \quad (7.2)$$

One property of the exponential family of distributions that we state without proof (Winn, 2003) is that the expectation of the natural statistic vector is the derivative of the log normalizer, with respect to the natural parameter. For a Dirichlet distribution,

$$\mathbb{E}_\theta\left[\log\theta_i\right] = \Psi\left(\alpha_i\right) - \Psi\left(\sum_{j=1}^K \alpha_j\right). \quad (7.3)$$

Thus, if we take the expectation of a Dirichlet distribution $p(\theta|\alpha)$ with respect to a variational Dirichlet distribution parameterized by γ, we have

$$\begin{aligned} \mathbb{E}_q\left[\log p(\theta|\alpha)\right] =& \mathbb{E}_q\left[\log\left(\frac{\Gamma\left(\sum_{i=1}^K \alpha_i\right)}{\sum_{i=1}^K \Gamma\left(\alpha_i\right)} \prod_{i=1}^K \theta^{\alpha_i - 1}\right)\right] \\ =& \mathbb{E}_q\left[\log\Gamma\left(\sum_{i=1}^K \alpha_i\right) - \sum_{i=1}^K \log\Gamma\left(\alpha_i\right) + \sum_{i=1}^K \left(\alpha_i - 1\right)\log\theta_i\right]. \end{aligned} \quad (7.4)$$

Only θ is governed by q, and $\log\theta_i$ is the natural statistic of the Dirichlet distribution when it as written as an exponential family distribution. In the variational distribution,

θ comes from a Dirichlet parameterized by γ, so using Equation 7.3, we have

$$\mathbb{E}_q\left[\log p(\theta|\alpha)\right] = \log\Gamma\left(\sum_{i=1}^{K}\alpha_i\right) - \sum_{i=1}^{K}\log\Gamma\left(\alpha_i\right) + \sum_{i=1}^{K}\left(\alpha_i - 1\right)\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right). \tag{7.5}$$

7.2 Expanding the Likelihood Bound for Document-Specific Terms

Recall that the objective function for the STM is

$$\begin{aligned}\mathcal{L}(\gamma,\nu,\phi;\tau,\theta,\pi,\beta) = \\ \mathbb{E}_q\left[\log p(\boldsymbol{\tau}|\alpha)\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\theta}|\alpha_D,\boldsymbol{\tau})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\pi}|\alpha_P,\boldsymbol{\tau})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{z}|\boldsymbol{\theta},\boldsymbol{\pi})\right] \\ +\mathbb{E}_q\left[\log p(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{\beta})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{\beta}|\sigma)\right] - \mathbb{E}_q\left[\log q(\boldsymbol{\theta}) + \log q(\boldsymbol{\pi}) + \log q(\boldsymbol{z})\right].\end{aligned} \tag{7.6}$$

In this section, we expand the terms in the $\mathcal{L}$ needed to perform document-specific expectations. This will provide the information needed to optimize document-specific variational parameters in the next section. We save the expansion of the remaining terms from $\mathcal{L}$ until Section 7.4.

7.2.1 LDA-like terms

The terms of equation 7.7 specific to a single document are

$$\begin{aligned}\mathcal{L}_d =& \mathbb{E}_q\left[\log p(\boldsymbol{\theta}_d|\alpha_D,\boldsymbol{\tau})\right] + \mathbb{E}_q\left[\log p(\boldsymbol{z}_d|\boldsymbol{\theta}_d,\boldsymbol{\pi})\right] \\ &+ \mathbb{E}_q\left[\log p(\boldsymbol{w}|\boldsymbol{z}_d,\boldsymbol{\beta})\right] - \mathbb{E}_q\left[\log q(\boldsymbol{\theta}_d) + \log q(\boldsymbol{z}_d)\right].\end{aligned} \tag{7.7}$$

We now expand each of these using the formula given in Equation 7.5. First, if we

consider the expectation over the distribution over latent topics,

$$\mathbb{E}_q\left[\log p(\boldsymbol{\theta}_d|\alpha_D,\boldsymbol{\tau})\right] = \log\Gamma\left(\sum_{j=1}^{K}\alpha_{D,j}\tau^*\right) - \sum_{i=1}^{K}\log\Gamma\left(\alpha_{D,i}\tau^*\right) + \sum_{i=1}^{K}\left(\alpha_{D,i}\tau^* - 1\right)\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right),$$

we can treat the truncated Dirichlet process as a Dirichlet distribution with a parameter that has been scaled by α_D. We postpone expanding the expectation over topic assignments $\boldsymbol{z}_d$ until the next section. For the expectation over the words, we note that the probability of the n^{th} word in document d taking topic k under the variational distribution is $\phi_{d,n,k}$ or (suppressing the document index) $\phi_{n,k}$ and given that assignment, the probability of the corresponding token $w_{d,n}$ being produced by topic k is $\beta_{k,w_{d,n}}$. Thus,

$$\mathbb{E}_q\left[\log p(\boldsymbol{w}|\boldsymbol{z},\boldsymbol{\beta})\right] = \sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\log\beta_{i,w_{d,n}}.$$

We are left with the entropy terms. First, the entropy for the per-document topic distribution is

$$-\mathbb{E}_q\left[\log q(\boldsymbol{\theta})\right] = -\log\Gamma\left(\sum_{j=1}^{K}\gamma_j\right) + \sum_{i=1}^{K}\log\Gamma\left(\gamma_i\right) - \sum_{i=1}^{K}(\gamma_i - 1)\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right),$$

which follows by the same reasoning used in equation 7.5. The entropy of a multinomial distribution is straightforward

$$\mathbb{E}_q\left[\log q(\boldsymbol{z})\right] = -\sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\log\phi_{n,i}.$$

7.2.2 The Interaction of Syntax and Semantics

We now move on to expanding $\mathbb{E}_q\left[\log p(\boldsymbol{z}|\boldsymbol{\theta},\boldsymbol{\pi})\right]$ from Equation 7.7. Rather than drawing the topic of a word directly from a multinomial, the topic is chosen from the renormalized point-wise product of two multinomial distributions. In order to handle the expectation of the log sum introduced by the renormalization, we introduce an additional variational parameter ω_n for each word via a Taylor approximation of the logarithm to find that $\mathbb{E}_q\left[\log p(\mathbf{z}|\boldsymbol{\theta},\boldsymbol{\pi})\right] =$

$$\begin{aligned}
&\mathbb{E}_q\left[\log\prod_{n=1}^{N}\frac{\theta_{z_n}\pi_{z_{p(n)},z_n}}{\sum_i^K\theta_i\pi_{z_{p(n)},i}}\right] = \mathbb{E}_q\left[\sum_{n=1}^{N}\log\theta_{z_n}\pi_{z_{p(n)},z_n} - \sum_{n=1}^{N}\log\sum_{i=1}^{K}\theta_i\pi_{z_{p(n)},i}\right] \\
\leq\ & \sum_{n=1}^{N}\mathbb{E}_q\left[\log\theta_{z_n}\pi_{z_{p(n)},z_n}\right] - \sum_{n=1}^{N}\mathbb{E}_q\left[\omega_n^{-1}\sum_{i=1}^{K}\theta_i\pi_{z_{p(n)},i}\right] + \log\omega_n - 1
\end{aligned}$$

$$\begin{aligned}
=\ & \sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\left(\Psi\left(\gamma_i\right) - \Psi\left(\textstyle\sum_{j=1}^{K}\gamma_j\right)\right) + \sum_{n=1}^{N}\sum_{i=1}^{K}\sum_{j=1}^{K}\phi_{n,i}\phi_{p(n),j}\left(\Psi\left(\nu_{j,i}\right) - \Psi\left(\textstyle\sum_{k=1}^{K}\nu_{j,k}\right)\right) \\
& - \left(\sum_{n=1}^{N}\omega_n^{-1}\sum_{i=1}^{K}\sum_{j=1}^{K}\phi_{p(n),j}\frac{\gamma_i\nu_{j,i}}{\sum_{k=1}^{K}\gamma_k\sum_{k=1}^{K}\nu_{j,k}} + \log\omega_n - 1\right). \qquad (7.8)
\end{aligned}$$

Combining this with the other expansions for a document gives us an individual

document's contribution to the objective function

$$
\begin{aligned}
\mathcal{L}_d &= \log\Gamma\left(\sum_{j=1}^{K}\alpha_{D,j}\tau^*\right) - \sum_{i=1}^{K}\log\Gamma\left(\alpha_{D,i}\tau^*\right) + \sum_{i=1}^{K}\left(\alpha_{D,i}\tau^* - 1\right)\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right) \\
&+ \sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right) + \sum_{n=1}^{N}\sum_{i=1}^{K}\sum_{j=1}^{K}\phi_{n,i}\phi_{p(n),j}\left(\Psi\left(\nu_{j,i}\right) - \Psi\left(\sum_{k=1}^{K}\nu_{j,k}\right)\right) \\
&\quad - \left(\sum_{n=1}^{N}\omega_n^{-1}\sum_{i=1}^{K}\sum_{j=1}^{K}\phi_{p(n),j}\frac{\gamma_i\nu_{j,i}}{\sum_{k=1}^{K}\gamma_k\sum_{k=1}^{K}\nu_{j,k}} + \log\omega_n - 1\right) \\
&+ \sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\log\beta_{i,w_{d,n}} \\
&- \log\Gamma\left(\sum_{j=1}^{K}\gamma_j\right) + \sum_{i=1}^{K}\log\Gamma\left(\gamma_i\right) - \sum_{i=1}^{K}(\gamma_i - 1)\left(\Psi\left(\gamma_i\right) - \Psi\left(\sum_{j=1}^{K}\gamma_j\right)\right) \\
&- \sum_{n=1}^{N}\sum_{i=1}^{K}\phi_{n,i}\log\phi_{n,i}. \qquad (7.9)
\end{aligned}
$$

Apart from the terms derived in Equation 7.8, the other terms here are very similar to the objective function for LDA. The expectation of the log of $p(\boldsymbol{\theta})$, $q(\boldsymbol{\theta})$, $p(\boldsymbol{z})$, $q(\boldsymbol{z})$, and $p(\boldsymbol{w})$ all appear in the LDA likelihood bound.

7.3 Document-specific Variational Updates

In this section, we derive the updates for all document-specific variational parameters other than $\boldsymbol{\phi}_n$, which is updated according to Equation 5.3.

Because we cannot assume that the point-wise product of of π_k and θ_d sums to one, we introduced a slack term ω_n in Equation 7.8; its update is

$$
\omega_n = \sum_{i=1}^{K}\sum_{j=1}^{K}\phi_{p(n),j}\frac{\gamma_i\nu_{j,i}}{\sum_{k=1}^{K}\gamma_k\sum_{k=1}^{K}\nu_{j,k}}.
$$

Because we couple $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$, the interaction between these terms in the normalizer prevents us from solving the optimization for $\boldsymbol{\gamma}$ and $\boldsymbol{\nu}$ explicitly. Instead, for each

$\boldsymbol{\gamma}_d$ we compute the partial derivative with respect to $\gamma_{d,i}$ for each component of the vector. We then maximize the likelihood bound for each $\boldsymbol{\gamma}_d$. In deriving the gradient, the following derivative is useful:

$$\begin{aligned} f(x) &= \sum_{i=1}^{N} \alpha_i \frac{x_i}{\sum_{i=j}^{N} x_j} \\ \Rightarrow \frac{\partial f}{\partial x_i} &= \frac{\alpha_i \sum_{j \neq i}^{N} x_j - \sum_{j \neq i}^{N} \alpha_j x_j}{\left(\sum_{i=1}^{N} x_i\right)^2}. \end{aligned} \tag{7.10}$$

This allows us to more easily compute the partial derivative of Equation 7.9 with respect to γ_i to be

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \gamma_i} &= \Psi'(\gamma_i)\left(\alpha_{D,i}\tau^* + \sum_{n=1}^{N}\phi_{n,i} - \gamma_i\right) - \Psi'\left(\sum_{j=1}^{N}\gamma_j\right)\sum_{j=1}^{K}\left[\alpha_{D,j}\tau^* + \sum_{n=1}^{N}\phi_{n,j} - \gamma_j\right] \\ &\quad - \sum_{n=1}^{N}\omega_n^{-1}\sum_{j=1}^{K}\left[\phi_{p(n),j}\frac{\nu_{j,i}\sum_{k\neq j}^{N}\gamma_k - \sum_{k\neq j}^{N}\nu_{j,k}\gamma_k}{\left(\sum_{k=1}^{N}\gamma_k\right)^2\sum_{k=1}^{N}\nu_{j,k}}\right] \end{aligned}$$

7.4 Global Updates

In this section, we expand the terms of Equation 7.7 that were not expanded in Equation 7.9. First, we note that $\mathbb{E}_q\left[\log \mathrm{GEM}(\boldsymbol{\tau}; \alpha)\right]$, because the variational distribution only puts weight on $\boldsymbol{\tau}^*$, is just $\log \mathrm{GEM}(\boldsymbol{\tau}^*; \alpha)$.

We can return to the stick-breaking weights by dividing each τ_z^* by the sum of all of the indices greater than z (recalling that τ sums to one), $T_z \equiv 1 - \sum_{i=1}^{z-1}\tau_i$. Using

this reformulation, the total likelihood bound, including Equation 7.9 as $\mathcal{L}_d$, is then[1]

$$\begin{aligned}
\mathcal{L} &= \sum_{d}^{M} \mathcal{L}_d \\
&+ (\alpha - 1) \log T_K - \sum_{z}^{K-1} \log T_z \\
&+ \log \Gamma \left(\sum_{j=1}^{K} \alpha_{T,j} \tau^* \right) - \sum_{i=1}^{K} \log \Gamma \left(\alpha_{T,i} \tau^* \right) + \sum_{i=1}^{K} \left(\alpha_{T,i} \tau^* - 1 \right) \left(\Psi \left(\nu_i \right) - \Psi \left(\sum_{j=1}^{K} \nu_j \right) \right) \\
&- \log \Gamma \left(\sum_{j=1}^{K} \nu_j \right) + \sum_{i=1}^{K} \log \Gamma \left(\nu_i \right) - \sum_{i=1}^{K} (\nu_i - 1) \left(\Psi \left(\nu_i \right) - \Psi \left(\sum_{j=1}^{K} \nu_j \right) \right). \qquad (7.11)
\end{aligned}$$

Variational Dirichlet for Parent-child Transitions

Like the update for $\boldsymbol{\gamma}$, the interaction between $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$ in the normalizer prevents us from solving the optimization for each of the ν_i explicitly. Differentiating the global likelihood bound, keeping in mind Equation 7.10, gives

$$\begin{aligned}
\frac{\partial L}{\partial \nu_{i,j}} &= \Psi' \left(\nu_{i,j} \right) \left(\alpha_{P,j} + \sum_{n=1}^{N} \sum_{c \in c(n)} \phi_{n,i} \phi_{c,j} - \nu_{i,j} \right) \\
&\quad - \Psi' \left(\sum_{k=1}^{K} \nu_{i,k} \right) \sum_{k=1}^{K} \left[\alpha_{P,k} + \sum_{n=1}^{N} \sum_{c \in c(n)} \phi_{n,i} \phi_{c,k} - \nu_{i,k} \right] \\
&\quad - \sum_{n}^{N} \phi_{n,i} \sum_{c \in c(n)} \left[\omega_c^{-1} \frac{\gamma_j \sum_{k \neq j}^{N} \nu_{i,k} - \sum_{k \neq j}^{N} \nu_{i,k} \gamma_k}{\left(\sum_{k=1}^{N} \nu_{j,k} \right)^2 \sum_{k=1}^{N} \gamma_k} \right].
\end{aligned}$$

Each of the $\boldsymbol{\nu}_i$ are then maximized individually using conjugate gradient optimization after transforming the vector to assure non-negativity.

[1]For simplicity, we present inference with the per-topic distribution β as a parameter. Inference for the complete model with β from a Dirichlet distribution requires adding an additional variational parameter. This is straightforward, but would further complicate the exposition.

Variational Top-level Weights

The last variational parameter is $\boldsymbol{\tau}^*$, which is the variational estimate of the top-level weights $\boldsymbol{\tau}$. Because τ_K^* is implicitly defined as $\left(1 - \sum_{i=0}^{K-1} \tau_i^*\right)$, τ_K^* appears in the partial derivative of $\boldsymbol{\tau}^*$ with respect to τ_k^* for $k < K$. Similarly, we must also use implicit differentiation with respect to the stick breaking proportions T_z, defined above. Taking the derivative and implicitly differentiating τ_K gives us

$$\begin{aligned}
\frac{\partial L_{\tau^*}}{\partial \tau_k^*} &= \left(\sum_{z=k+1}^{K-1} \frac{1}{T_z}\right) - \frac{\alpha - 1}{T_K} \\
&+ \alpha_D \sum_d^M \left(\Psi\left(\gamma_{d,k}\right) - \Psi\left(\textstyle\sum_{j=1}^K \gamma_{d,j}\right)\right) - \alpha_D \sum_d^M \left(\Psi\left(\gamma_{d,K}\right) - \Psi\left(\textstyle\sum_{j=1}^K \gamma_{d,j}\right)\right) \\
&+ \alpha_T \sum_z^K \left(\Psi\left(\nu_{z,k}\right) - \Psi\left(\textstyle\sum_{j=1}^K \nu_{z,j}\right)\right) - \alpha_T \sum_z^K \left(\Psi\left(\nu_{z,K}\right) - \Psi\left(\textstyle\sum_{j=1}^K \nu_{z,j}\right)\right) \\
&- K\left[\alpha_T \Psi\left(\alpha_T \tau_k^*\right) - \alpha_T \Psi\left(\alpha_T \tau_K^*\right)\right] \\
&- M\left[\alpha_D \Psi\left(\alpha_D \tau_k^*\right) - \alpha_D \Psi\left(\alpha_D \tau_K^*\right)\right] \qquad (7.12)
\end{aligned}$$

which we use with conjugate gradient optimization after appropriately transforming the variables to ensure non-negativity.