Foundations and Trends® in Information Retrieval
Vol. XX, No. XX (2016) 1–112
© 2016
DOI: 10.1561/XXXXXXXXXX

Applications of Topic Models

Jordan Boyd-Graber
University of Colorado
Jordan.Boyd.Graber@colorado.edu

Yuening Hu
Yahoo!
ynhu@yahoo-inc.com

David Mimno
Cornell University
mimno@cornell.edu

Contents

Abstract

How can a single person understand what's going on in a collection of millions of documents? This is an increasingly common problem: sifting through an organization's e-mails, understanding a decade worth of newspapers, or characterizing a scientific field's research. Topic models are a statistical framework that help users understand large document collections: not just to find individual documents but to understand the general themes present in the collection.

This survey describes the recent academic and industrial applications of topic models and reviews successful applications to help understand fiction, non-fiction, scientific publications, and political texts. In addition to these success stories on particular applications, this survey reviews technical innovations necessary for effective topic model applications spanning information retrieval, visualization, statistical inference, multilingual modeling, and linguistic understanding.

. *Applications of Topic Models*. Foundations and Trends$^{\text{®}}$ in Information Retrieval, vol. XX, no. XX, pp. 1–112, 2016.
DOI: 10.1561/XXXXXXXXXX.

1

The What and Wherefore of Topic Models

Imagine that you're an intrepid reporter with an amazing scoop: you have twenty-four hours of exclusive access three decades of e-mails sent within a corrupt corporation. You know there's dirt and scandal there, but it's been well-concealed by the corporation's political friends. How are you going to understand this haystack well enough to explain it to your devoted readers under such a tight deadline?

1.1 Tell me about your haystack

Unlike the vignette above, interacting with large text data sets is often posed as a needle in a haystack problem. The user is confronted with a large, undifferentiated collection of documents: the entire web, a hard drive, or all communications within a company. The poor user—faced with documents that would take a decade to read—is looking for a single needle: a document (or at most a handful of documents) that matches what the user is looking for. This could be a "smoking gun" e-mail, the document that best represents a concept [Salton, 1968] or the answer to question [Hirschman and Gaizauskas, 2001].

These questions are important. The sub-discipline of information retrieval is built upon systematizing, solving, and evaluating this problem. Google's empire is built on the premise of users typing a few keywords into a search engine box and seeing quick, consistent search results. However, this is not the only problem that confronts those interacting with large text datasets.

A different, but related problem is *understanding* large document collections, common in science policy [Talley et al., 2011], journalism, and the humanities [Moretti, 2013a]. There isn't just one precious silver needle in the haystack. At the risk of abusing the metaphor, *sometimes you care about the straw.* Instead of looking for a smoking gun alerting to you some crime an evil company committed, perhaps you're looking for a sin of omission: did this company never talk about diversity in it's workforce? Instead of a single answer to a question, perhaps you're looking for a diversity of responses: what are the different ways that people account for rising income inequality. Instead of looking for one document, perhaps you want to answer population level statistics: what proportion of Twitter users have ever talked about gun violence?

At first, might seem that answering these questions require building an extensive ontology or categorization scheme. For every new corpus, define all of the buckets that a document could fit into, have some librarians and archivists put each document into the correct buckets, perhaps automate the process with some supervised machine learning, and then collect summary statistics when you're done.

Obviously, such laborious processes are possible—they've been done for labeling congressional speech [1] and understanding emotional state Wilson and Wiebe [2005]—and remain an important part of social science, information science, library science, and machine learning. But these processes aren't always possible, fast, or even the optimal outcome [illegible]e had infinite resources. First, they obviously require a significant investment of time and resources. Even creating the *list* of categories is a difficult task and requires careful deliberation and calibration. It's possible that a particular question does not warrant the time or effort: the œvre of a minor author (only of interest to a few), or the tweets of a day (not relevant tomorrow).

[1] www.congressionalbills.org/

Topic 3	Topic 6	Topic 9	Topic 14	Topic 22
trading	gas	state	ferc	group
financial	capacity	california	issue	meeting
trade	deal	davis	order	team
product	pipeline	power	party	process
price	contract	utilities	case	plan

Figure 1.1: Five topics from a twenty-five topic model fit on Enron e-mails. Example topics concern financial transactions, natural gas, the California utilities, federal regulation, and planning meetings.

Topic models allow us to answer these big picture questions quickly and cheaply. Topic models provide a framework for understanding these document collections for both humans or computers. This survey explores the ways that humans and computers make sense of document collections through tools called topic models. For readers already comfortable with topic models, feel free to skip this chapter; we'll mostly cover the definitions and implementations of topic models.

1.2 What is a Topic Model

Returning to our motivating example, consider the e-mails from Enron, the prototypical evil corporation of the turn of the century. Imagine that you're an investigative reporter, the first to get your hands on Enron e-mails. You know that wrongdoing happened, but you don't know who did it or how it was planned and carried out. You have suspicions (e.g., around the California spot market), but you are curious to know if there are other skeletons in the closet, and you're highly motivated to find them.

So you run a topic model on the data. True to its name, a topic model gives you "topics", collections of words that make sense together. Looking at the Enron e-mails, we can see topics about gas contracts, California regulators, and stock prices (Figure 1.1).

When we call a topic about X, that's a *post hoc* label applied by humans (more on this in Chapter 3.1). A topic only gives you a jumbled

Yesterday, SDG&E filed a motion for adoption of an electric procurement cost recovery mechanism and for an order shortening time for parties to file comments on the mechanism. The attached email from SDG&E contains the motion, an executive summary, and a detailed summary of their proposals and recommendations governing procurement of the net short energy requirements for SDG&E's customers. The utility requests a 15-day comment period, which means comments would have to be filed by September 10 (September 8 is a Saturday). Reply comments would be filed 10 days later.

Topic	Probability
9	0.42
11	0.05
8	0.05

Figure 1.2: Example document from the Enron corpus and its association to topics. Although it doesn't contain the word "California", it discusses a single California utility's dissatisfaction with how much it is paying for electricity.

"bag of words"; it remains the responsibility of the human consumer of topic models to go further and make sense of these piles of straw (we discuss labeling the topics more in Chapter 3).

Making sense of one of these piles requires actually reading the document, which topic models also provide. Each topic is associated with individual documents. For example, the document in Figure 1.2 is about a California utility's reaction to the short-term electricity market and exemplifies Topic 9 from Figure 1.1. If we get a sense that Topic 9 is of interest, we can explore deeper to find other documents.

1.3 Foundations

Topic models began with a linear algebra approach [Deerwester et al., 1990] called latent semantic analysis (LSA): find the best low rank approximation of a document-term matrix (Figure 1.3). While these approaches have seen a resurgence in recent years [Anandkumar

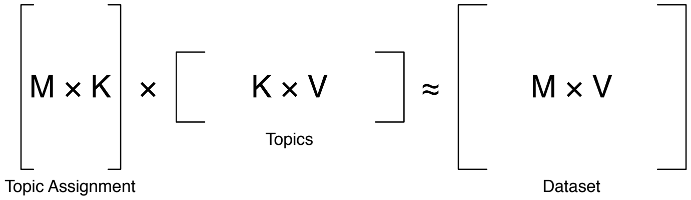


Figure 1.3: A matrix formulation of finding K topics for a dataset with M documents and V unique words. While the original formulation of topic modeling approaches such as latent semantic analysis (LSA), we focus on probabilistic techniques in the rest of this survey.

Distribution	Density	Parameter Examples	Draw Examples
Dirichlet	$\frac{\prod_{i=1}^{K}\Gamma(\alpha_i)}{\Gamma(\sum_{i=1} K\alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$	$\alpha = (1.1, 0.1, 0.1)$	$\theta = (0.8, 0.15, 0.05)$
Gaussian	$\frac{1}{\sqrt{2\sigma^2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$	$\mu = 2, \sigma = 1.1$	$x = 2.2$
Discrete	$\prod_i \phi_i^{\mathbb{1}[w=i]}$	$\phi = (0.1, 0.6, 0.3)$	$w = 2$ (second index)

Figure 1.4: Examples of probability distributions used in the generative stories of topic models.

et al., 2012a, Arora et al., 2013], we focus on probabilistic approaches, which are intuitive, work well, and allow for easy modification (as we see later in many of our later chapters).

1.3.1 Probabilistic Building Blocks

Probabilistic models begin with a generative story: a recipe listing a sequence of random events that creates the dataset you're trying to explain. Figure 1.4 lists some of the key players in these stories, how they're parameterized and what their draws look like. Let's briefly discuss them, as we'll use them to build a wide variety of topic models later.

Gaussian If you know any probability distribution already, it's the Gaussian. It does not have a role in the most basic topic models that we'll discuss here, but it will later (e.g., Chapter ??). We do include it because it's a useful point of comparison against the other distributions we're using (since it's perhaps the easiest to understand and best known).

A Gaussian distribution is a distribution over all real numbers (e.g., $0.0, 0.5, -4.2, \pi, \ldots$). You can ask it to spit out a number, and it will give you some real number between negative infinity and positive infinity. But not all numbers have equal probability. Gaussian distributions are parameterized by a mean μ and variance σ. Most samples from the distribution will be near the mean μ; how close is determined by the variance: higher variances will cause the samples to be more spread out.

Discrete While Gaussian distributions are over a continuous space, documents are combinations of discrete words.[2] Thus, we need a distribution over discrete sets.

A useful metaphor for thinking about discrete distributions is a weighted die. The number of faces on the die is its dimension, and each face is a distinct outcome. Each outcome has its own probability of how likely that outcome is; this is the parameter of a discrete distribution (Figure 1.4).

Topic models are described by discrete distributions (sometimes called multinomial distributions) over words and topics. The distribution over words is called the topic distribution; each of the topics gives higher weights to some words more than others (e.g., in Topic 9 from the Enron corpus, "state" and "california" have higher probability than other words). Each document also has an "allocation" for each topic: documents are about a small handful of topics, and most documents have very low weights for most of the possible topics.

Dirichlet Although discrete distributions are the star players in topic models, they are the end of the story. The story actually begins with Dirichlet distributions. Dirichlet distributions are distributions that produce discrete distributions. Like the Gaussian distribution, they

[2] An emerging trend in natural language processing research is to view words as embedded in a continuous space. We discuss these "representation learning" approaches and their connection to topic modeling in Chapter 9.

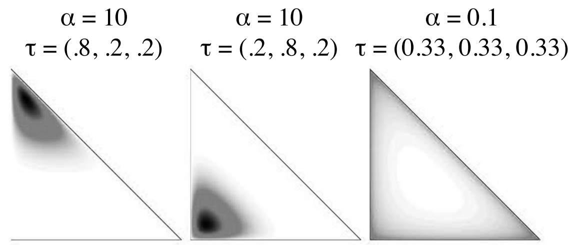


Figure 1.5: Given different Dirichlet parameters, the Dirichlet distribution can either be informative (left, middle) or sparse (right). Sparse distributions encourage distributions to favor only a small number of elements but don't care which ones. This is consistent with our intuitions of how documents are written: they are only about a few things, and topics contain only a handful of words.

have parameters analagous to a mean and variance. The mean is called the "base measure" τ and is the expected value of the Dirichlet: what you'd get if you averaged many draws from the Dirichlet. The concentration parameter α controls how far away draws are from the base measure.

If α is very large, then the draws from a Dirichlet will be very close to τ (Figure 1.5, left). If α is small, however, something more interesting happens: the discrete distributions become sparse (Figure 1.5, right). A sparse distribution is a distribution where only a few values have high probability and most are small.

Because topic models are meant to reflect real documents, modeling sparsity correctly is important. When a person sits down to write a document, they only write about a handful of topics. They don't write about every possible topic under the sun, and the sparsity of Dirichlet distributions is the probabilistic tool that encodes this intuition.

1.4 Latent Dirichlet Allocation

We now have all the tools we need to tell the complete story of the prototypical modern topic model: latent Dirichlet allocation [Blei et al., 2003]. Latent Dirichlet allocation[3] posits a "generative process" about how the data came to be. We assemble the probabilistic pieces to tell this story about generating topics and how those topics are used to create diverse documents.

For example, consider the document in Figure 1.2. To generate it, we choose a distribution over all of the topics. This is θ. For this document, the distribution favors Topic 9 about California. It's higher for this topic than any other topic. For each word in the document, the generative process chooses a topic assignment z_n. For this document, most of those will be Topic 9.

Then, for each word, we need to choose which word will appear. This comes from Topic 9's distribution over words (multiple topics' have word distributions shown in Figure 1.1). Each is a discrete draw from the topic's word distribution, which makes words like "California", "state", and "Sacramento" more likely.

Generating Topics The first part of the story is to create the topics. The user specifies that there are K distinct topics. Each of the K topics is drawn from a Dirichlet distribution with a uniform base distribution and concentration parameter λ: $\phi_k \sim \mathrm{Dir}(\lambda \boldsymbol{u})$. The discrete distribution ϕ_k has a weight for every word in the vocabulary.

Document Allocations Document allocations are distributions over topics for each document. This encodes what a document is about; the sparsity of the Dirichlet distribution's concentration parameter α ensures that the document will only be about a few topics. Each document has a discrete distribution over topic: $\theta_d \sim \mathrm{Dir}(\alpha \boldsymbol{u})$.

[3] The name LDA is a play on LSA, its non-probabilistic forerunner (latent semantic analysis). Latent because we use probabilistic inference to infer missing probabilistic pieces of the generative story. Dirichlet because of the Dirichlet parameters encoding sparsity. Allocation because the Dirichlet distribution encodes the prior for each document's allocation over topics.

Words in Context Now that we know what each document is about, we need to actually create the words that appear in the document. We assume[4] that there are N_d words in document d. For each word n in the document d, we first choose a **topic assignment** $z_{d,n} \sim \text{Discrete}(\theta_d)$. This is one of the K topics that tells us which topic the word token is from, but not what the word is.

To select which word we'll see in the document, we draw from the a discrete distribution again. Given a words topic assignment $z_{d,n}$, we draw from that topic to select the word: $w_{d,n} \sim \phi_{z_{d,n}}$. The topic assignment tells you what the word is about, and then this selects which distribution over words we use to generate the word.

It goes without saying that the generative story is a fiction [Box and Draper, 1987]. Nobody is sitting down with dice to decide what to type in on their keyboard. We use this story because it is *useful*. This fanciful story about randomly choosing a topic for each word can help us because if we assume this generative process, we can work backwards to find the topics that explain how a document collection was created: every word, every document, gets associated with these underlying topics.

This fantasy helps us order our document collection: by assuming this story, we can discover *topics* (which certainly don't exist) so we can understand the common themes that people use to write documents. As we'll see in later chapters, slight tweaks of this generative story allow us to tell uncover more complicated structures: how authors prefer specific topics, how topics change over time, or how topics can be used across languages.

1.5 Inference

Given a generative model and some data, the process of uncovering the hidden probabilistic pieces of the generative story is called *inference*. More concretely, it is a recipe for generating algorithms to go from data to *topics that explain a dataset*.

[4] It's possible to model this in the generative story as well, e.g., with a Poisson distribution. However, we often do not care about document *lengths*—only what the document is about—so we can usually ignore this part of the story.

There are many flavors of algorithms for posterior inference: message passing [Zeng et al., 2013], variational inference [Blei et al., 2003], gradient descent [Hoffman et al., 2010], and Gibbs sampling [Griffiths and Steyvers, 2004]. All of these algorithms have their advocates and reasons you should use them. However, that discussion is better left for other venues. In this survey, we focus on Gibbs sampling, which is simple, intuitive, and fast [Yao et al., 2009].

We present the results of Gibbs sampling without derivation, which—along with the history of its origin in statistical physics—are well described elsewhere.[5] We use a variety of Gibbs sampling called *collapsed* Gibbs sampling, which ignores some of the pieces of the generative story: it ignores the topics and the document allocations to only focus on the topic assignments. Some relatively straightforward math allows you to recreate the topics and document allocations if you only know the topic assignments.

1.5.1 Random Variables

Topic Assignments Recall that every individual token gets a topic assignment. For example, an instance of the word "compilation" might be in a business topic in one document and in an arts topic in another document. It's even possible that the same word might be assigned to different topics in the same document, as each instance of the word has its own topic assignment. Because topic models care about *global* properties, we'll use aggregate statistics derived from these topic assignments.

Document Allocation The document allocation is a distribution over the topics for each document; in other words, it says how popular each topic is in a document. If we count up how often a document uses a topic, this gives us an idea of the popularity. Let's define $N_{d,i}$ as the number of times document d uses topic i. Clearly, this is larger for more popular topics; however, it's not a probability because it is larger than one. We can make it a probability by dividing by the number of words

[5] We recommend Resnik and Hardisty [2009] for additional information on derivation.

in a document

$$\frac{N_{d,i}}{\sum_k N_{d,k}}, \tag{1.1}$$

but this is problematic because it can sometimes give us zero and ignores the influence of the Dirichlet distribution; a better estimate is[6]

$$\theta_{d,i} \approx \frac{N_{d,i} + \alpha_i}{\sum_k N_{d,k} + \alpha_k}. \tag{1.2}$$

It's important that this is never zero because we don't want it to be impossible for a topic to get used in a particular document. This helps the sampler explore more of the possible combinations.

Topics Each topic is a distribution over words. To understand what a topic is about, we look at the profile of all of the tokens that have been assigned to that topic. We estimate the probability of a word in a topic as

$$\phi_{i,v} \approx \frac{V_{i,v} + \beta_v}{\sum_w V_{i,w} + \beta_w}, \tag{1.3}$$

where β is the Dirichlet parameter for the topic distribution.

1.5.2 Algorithm

The algorithm for learning a topic model is only based on the topic assignments, but we'll use our estimates for the topics ϕ_k and the documents θ_d discussed above. We begin by setting these topic assignments randomly: if we have K topics, each word has equal chance to be associated with any of the topics.

These topics will be quite bad, but we'll improve them one word at a time. You then change the topic assignments for each word in a way the reflects the underlying probabilistic model of the data. Each pass over the data makes the topics slightly better. Eventually, the topics will "converge" to something reasonable and you can consider yourself done.

[6] To be technical, Equation 1.1 is a maximum likelihood estimate and Equation 1.2 is the maximum *a posteriori*, which incorporates the influence of both the prior and the data.

The equation for the probability of assigning a word to a particular topic combines these two factors[7]

$$p(z_{d,n} = j \,|) = \theta_d \phi_j = \left(\frac{N_{d,i} + \alpha_i}{\sum_k N_{d,k} + \alpha_k} \right) \left(\frac{V_{i,w_{d,n}} + \beta_v}{\sum_w V_{i,w} + \beta_w} \right). \tag{1.4}$$

Computing this value will eventually give you a vector equal to the number of topics you chose. The next step is to randomly choose one of those indices with probability proportional to the vector value. You now assign that word to the topic, update $N_{,}$ and $V_{,}$, and move on to the next word and repeat. These topics will be quite bad, but we'll improve them one word at a time.

At the very end of the algorithm, we can use the estimates of each topic (Equation 1.3) to summarize the main themes of the corpus and the estimates of each document's topic distribution (Equation 1.2) to start exploring the collection automatically (Chapter 2) or with a human in the loop (Chapter 3).

The algorithm that we have sketched here is the foundation of many of the more advanced models that we'll discuss later in the survey. While we won't describe the algorithms in detail, we will occasionally make reference to this sketch to highlight challenges or difficulties in implementing topic models.

1.5.3 Implementations

Hopefully the previous algorithm sketch has convinced you that implementing topic models is not a Herculean task; most skilled programmers can complete a reasonable implementation of topic models in less than a day. However, we would suggest not trying to implement topic models yourself, as there are many solid implementations that help users get to useful results more quickly, particularly as topic models often require extensive preprocessing.

[7] To be theoretically correct, it is important not to include the count associated with the token you are currently sampling in these counts, which becomes more clear if the probability is written as $p(z_{d,n} = j \,|\, z_{d,1} \ldots z_{d,n-1}, z_{d,n+1} \ldots z_{d,N_d}, w_{d,n})$ to show the dependence on the topic assignments of *all other* token but not this token.

Mallet is fast and is a widely used implementation in Java [McCallum, 2002]. This is where you should probably start, in our biased opinion. Java uses highly-optimized Gibbs sampling implementations and can work from a variety of text inputs. It's well documented, mature, and runs well on a multi-core machine, allowing it to process up to millions of documents. Variational inference is the other major option [Blei et al., 2003, Langford et al., 2007], but often requires a little more effort for new users to get a first result.

However, if your corpus is truly large, it may be worthwhile considering techniques that can be parallelized over large computer clusters. These techniques can be based on variational inference [Narayanamurthy, 2011, Zhai et al., 2012] or on sampling [Newman et al., 2008].

While these implementations allow you to run *specific* topic models, other frameworks allow you to specify arbitrary generative models. This allows for quick prototyping of topic models and integrating topic models with other probabilistic frameworks like regression or collaborative filtering. Examples of these general frameworks include Stan [Stan Development Team, 2014], Infer.net, and Church.

1.6 Rest of this Survey

In each of the following chapters, we focus on an application of topic models, gradually increasing the complexity of the underlying models. The chapters do occasionally refer to each other, but a reader should be able to read each of the chapters independently.

The next chapter returns to the distinction between high level overviews and finding a needle in a haystack. We show how a high level overview can help users and algorithms find documents of interest. We show how a high level overview can help algorithms (Chapter 2) and users (Chapter 3) find documents of interest.

These tools help enable new applications of topic models: how understanding newspapers (Chapter 4) reveals the march of history, how the corpus of writers of fiction (Chapter 5) illuminates societal norms, how the writings of science reveal innovation (Chapter 6), or how politicians' speeches (Chapter 7) reveal schisms in political organizations.

2

Information Retrieval

Topic models explore and summarize document collections, without knowing users' information need. In the contrary, traditional information retrieval (IR) systems aim to retrieve relevant documents given users' information need. Users start with their information need in the form of queries, and early IR systems treat both the query and documents as "bag of words", retrieve and rank the documents by measuring the word overlap between queries and documents.

However, the ability of this direct and simple matching is always limited. Words with similar meaning or in different forms should also be considered as matched instead of being ignored. **Language modeling** has been one of the most popular frameworks to capture such semantic relationships. Also, humans would also use background knowledge to interpret and understand the queries and "add" missing words [Wei, 2007], which provides another way often referred as **query expansion** to improve the retrieval and ranking results.

Both directions can be pursued by learning and discovering the semantic relations between words and further the semantic relations between queries and documents. Topic models, which describe each topic using weighted words and model each document as a distribution over all topics, provides a semnatic relations between query words and documents [Deerwester et al., 1990, Hofmann, 1999b]. Such semantic relations can be applied to smoothing the language models, or introducing related words in query expansion. This chapter focuses on how to apply topic models in document language modeling [Lu et al., 2011, Wei and Croft, 2006] and query expansion [Park and Ramamohanarao, 2009, Andrzejewski and Buttler, 2011] to further improve the ranking results of information retrieval. We start with a brief review of the traditiona information retrieval framework.

2.1 Document Language Modeling in IR

The language modeling approach [Croft and Lafferty, 2003, Ponte and Croft, 1998, Song and Croft, 1999] is one of the main frameworks for using topic models in IR systems, since it has been shown to be effective probabilistic framework for studying information retrieval problem [Ponte and Croft, 1998, Berger and Lafferty, 1999].

A statistical language model is to estimate the probability of word sequences, denoted as $p(w_1, w_2, \cdots, w_n)$. In practice, the statistical language model is often approximated by N-gram models. A unigram model assumes each word in the sequence is independent, and is denoted as,

$$p(w_1, w_2, \cdots, w_n) = p(w_1)p(w_2)\cdots p(w_n) \tag{2.1}$$

A trigram model assumes the probability of the current word only depends on the previous two words, and it is represented as,

$$p(w_1, w_2, \cdots, w_n) = p(w_1)p(w_2|w_1)p(w_3|w_1, w_2)\cdots p(w_n|w_{n-2}, w_{n-1}) \tag{2.2}$$

In the application of information retrieval, the queries are generated by a probabilistic language model based on a document [Zhai and Lafferty, 2001a]. More specifically, each document is viewed as a language sample, and a language model for each document is estimated based on document terms. Then the probability of generating the query is estimated by each document language model. The probability of a query is computed by multiplying the probabilities of generating each query term using different document language model, and the documents are ranked based on the probability.

Given a document sample d, a straightforward way to estimate the probability of generating a word w is to use maximum likelihood estimation

$$p_{\mathtt{ml}}(w|d) = \frac{n_{d,w}}{n_{d,\cdot}} \tag{2.3}$$

where $n_{d,w}$ is the term frequency of word w in document d, and $n_{d,\cdot}$ is the total number of tokens in document d. Then the probability of the given query q can be computed by,

$$p(q|d) = \prod_{w \in q} p(w|d) = \prod_{w \in q} \frac{n_{d,w}}{n_{d,\cdot}} \tag{2.4}$$

Then the documents are ranked based on this probability. Higher probability implies the corresponding document is more relevant to the given query [Song and Croft, 1999]. However, a document is often too small to cover all the terms in the query. The probability of a missing term is zero, which means the probability of the whole query is zero and causes problems for ranking documents.

This data sparsity problem can be fixed by smoothing, which allocates some non-zero probability to the missing terms. In fact, topic models provide a unique way to extract the word probabilities given the corpus, which can be used to smooth doucument language models. In this section, we summarize two simple smoothing methods, and then introduce how topic models are applied to smooth document language models in the next section.

There are two major directions for smoothing: **interpolation** [Jelinek and Mercer, 1980, Mackay and Petoy, 1995, Ney et al., 1994, Ponte and Croft, 1998, Zhai and Lafferty, 2001a] and **backoff** [Katz, 1987, Song and Croft, 1999]. The interpolation-based method discount the counts of the seen words and distribute the extra counts to both seen words and unseen words. An alternative backoff smoothing strategy trusts the maximum likelihood estimation for high count words, discounts and redistributes mass only for the less common words [Zhai and Lafferty, 2001a].

Here we introduce two popular and simple interpolation smoothing methods, which are further used by topic models to smooth document language models.

Jlinek-Mercer method The Jlinek-Mercer method [Jelinek and Mercer, 1980] is a linear interpolation of the maximum likelihood model in a document with the model based on the whole corpus, and a coefficient λ is used to combine the two parts.

$$p(w|d) = (1 - \lambda)p_{\mathtt{ml}(w|d)} + \lambda p(w|\mathcal{C}) \tag{2.5}$$

where $\mathcal{C}$ denotes the whole corpus. This simple mixture solves the data sparsity problem. For terms that occur in the document d, the maximum likelihood estimator (Equation 2.3) is not accurate given the limited size of a document, thus it is smoothed with the more reliable corpus level probability. For the missing terms in the document d, the probability is not zero any more, but fall back to the corpus level probability. This smoothing method has been explored by Ponte and Croft [1998] and Song and Croft [1999] in information retrieval task, except Ponte and Croft [1998] explored a weighted product version instead of a linear interpolation.

$$p(w|d) = p_{\mathtt{ml}}(w|d)^{(1-\lambda)} \times p(w|\mathcal{C})^{\lambda} \tag{2.6}$$

But in general, the linear interpolation is more popular since the resulting probability is still normalized [Song and Croft, 1999].

Bayesian Smoothing using Dirichlet Priors A language model can be viewed as a multinomial distribution, thus it can be smoothed by applying the Dirichlet distribution as the conjugate prior [Mackay and Petoy, 1995]. The smoothing model is adding extra prior count for each word to smooth the probability of unseen words,

$$p(w|d) = \frac{n_{d,w} + \mu p(t|\mathcal{C})}{\sum_{v \in V} n_{d,v} + \mu} \tag{2.7}$$

where the Dirichlet prior is decided by parameter μ and the corpus-level probabilities $p(v|\mathcal{C})$ as follows,

$$(\mu p(v_1|\mathcal{C}), \mu p(v_2|\mathcal{C}), \cdots, \mu p(v_n|\mathcal{C})) \tag{2.8}$$

2.2 Applying Topic Models to Document Language Models

Topic models, which model each document as a mixture of topics and each topic as a mixture of words, offer an interesting framework to model documents in information retrieval. Wei and Croft [2006] introduce Latent Dirichlet Allocation (LDA) to learn the relationship between query words and documents, and the conditional probability of a query word w given a document d is computed as marginalizing all topics,

$$p_{\mathtt{lda}}(w|d) = \sum_{z=1}^{K} p(w|z, \hat{\phi}) p(z|\hat{\theta}, d) \tag{2.9}$$

where $\hat{\theta}$ is the posterior estimate for the document-topic distribution θ; $\hat{\phi}$ is the posterior estimate for the topic-word distribution ϕ; K is the total number of topics.

Because topic models learn the semantic relationships between words and documents through learning the hidden topics, and their posterior estimates of document-topic distribution and topic-word distribution are smoothed by the Dirichelet priors, topic models learn a better and smoothed semantic relationship between words and documents. However, the topic models based on documents only lose the direct connection between query words and documents as in the original document language models, but it is a good approach to complement the original document language models. Thus Wei and Croft [2006] further propose to combine the LDA-based document model with the original smoothed document model (Equation 2.6) through a linear interpolation,

$$p(w|d) = \omega((1 - \lambda)p_{\mathtt{ml}(w|d)} + \lambda p(w|\mathcal{C})) + (1 - \omega)p_{\mathtt{lda}}(w|d) \tag{2.10}$$

where ω is the coefficient which combines the LDA-based document model with the general smoothed language model.

Following Wei and Croft [2006], Lu et al. [2011] further evaluate the performance of applying topic models into the document language model framework. Instead of combining with the language model with Jlinek-Mercer smoothing (Equation 2.6), Lu et al. [2011] smooth the document language model with the Bayesian smoothing (Equation 2.8), and the final linear combination with topic models is as,

$$p(w|d) = \omega \frac{n_{d,w} + \mu p(t|\mathcal{C})}{\sum_{v \in V} n_{d,v} + \mu} + (1 - \omega) p_{\texttt{lda}}(w|d) \tag{2.11}$$

While using different smoothing strategies, both approaches ([Wei and Croft, 2006] and [Lu et al., 2011]) apply topic models to connect the query words with documents through hidden topics. As a result, better semantic relationships between query words and documents can be learned to complement the document language models.

2.3 Query Expansion in IR

The document language models in Information Retrieval [Ponte and Croft, 1998] have been attempting to model the query generation process based on the document models. However, a big problem is that these models abandon modeling the query-document relevance explicitly [Lavrenko and Croft, 2001], which is very important in traditional Information Retrieval task.

In fact, queries, which are normally brief and using informal languages from users, diverse significantly from the language in documents [Müller and Gurevych, 2009]. This semantic gap or lexical gap can result in poor query-document relevance, even though the document is quite relevant from the users' view point. This is due to users, who add "missing" or potentially related words implicitly in their minds and then consider the query-document relevance.

Query expansion simulates the similar process. Query expansion normally analyzes the relationships between the query words and other words, and tries to find potential related words so that the original query is better represented and better query-document relevance can be obtained. This section reviews the classic query expansion frameworks in information retrieval and then we introduce the related works about using topic models for query expansion in the next section.

2.3.1 Learning Query-Word Relationships for Query Expansion

There are two main steps for query expansion. The first step is to find the relationships between queries and words and select the top related words to expand the query. The second step is to apply the expanded queries for ranking and compute the final ranking relevance scores. We start with the first step. Two major directions have been explored: query language model [Zhai and Lafferty, 2001b] and relevance model [Lavrenko and Croft, 2001].

Query Language Model To learn the query-word relationship, Zhai and Lafferty [2001b] build up a query language model to estimate the probability $p(w|q)$ of a word w given a query q. However, it is not easy to learn a good query language model since the query content is too limited.

Zhai and Lafferty [2001b] propose to use both the query content and the relevant documents (sometimes referred as feedback documents or clicked documents) to estimate the query language model. Let $\hat{\theta}_{\mathcal{F}}$ be the estimated query language model based on the relevant documents, the combined query model $\hat{\theta}_{Q'}$ can be computed by,

$$\hat{\theta}_{Q'} = (1-\mu)\hat{\theta}_Q + \mu\hat{\theta}_{\mathcal{F}} \tag{2.12}$$

A simple way to estimate the feedback query model $\hat{\theta}_{\mathcal{F}}$ is a unigram language model θ which generates each word in $\mathcal{F}$ independently. However, most documents contain not only the query relevant information, but also the background information. As a result, Zhai and Lafferty [2001b] propose to generate a relevant document by a mixture model, which combines a query model $p(w|\theta_Q)$ with a collection language model $p(w|\mathcal{C})$. Thus the log-likelihood of the relevant document is,

$$\log p(\mathcal{F}|\theta) = \sum_i \sum_w c(w, d_i) \log((1-\lambda)p(w|\theta_Q) + \lambda p(w|\mathcal{C})) \tag{2.13}$$

where $c(w, d_i)$ is count of word w in document d. The related parameters θ are estimated by the EM algorithm (more details can be found in Zhai and Lafferty [2001b]). By combing with the information from the relevant documents, the query language model is more robust, thus query-word relationships are better represented.

Relevance Model Unlike the query language model approach [Zhai and Lafferty, 2001b], Lavrenko and Croft [2001] assume both the query and the relevant documents are random samples from an unknown relevance model R. Given the query q, they approximate the probability $p(w|R)$ based on the observed query q as the following,

$$p(w|R) \approx p(w|q) = \frac{p(w,q)}{p(q)} \tag{2.14}$$

To estimate the joint probability $p(w,q)$, Lavrenko and Croft [2001] assumes the word w and the query q are sampled independently from the same distribution, e.g., from a unigram distribution, then the joint probability can be computed as,

$$p(w,q) = \sum_{D \in \mathcal{C}} p(D)p(w,q|D) = \sum_{D \in \mathcal{C}} p(D)p(w|D)p(q|D) \tag{2.15}$$

Then the multinomial distribution $p(w|q)$ for a given query q can be computed as follows,

$$p(w|q) = \frac{p(w,q)}{p(q)} = \sum_{D \in \mathcal{C}} p(w|D)p(D|q) \tag{2.16}$$

Both the query language model and the relevance model capture the relationship between the query and other words, based on which the top related words can be selected for query expansion.

2.3.2 Ranking Relevance with Query Expansion

Given the related words for query expansion, the next question is how to apply the expanded queries for computing the final ranking relevance score. The combination can happen either before or after the relevance score is computed.

Zhai and Lafferty [2001b] combine the expanded query language model $\hat{\theta}_{\mathcal{F}}$ with the original query language model $\hat{\theta}_Q$ as one query language model $\hat{\theta}_{Q'}$ (Equation 2.12). Given a query q generated from the expanded query model $p(q|\hat{\theta}_{Q'})$, and a document d generated from a document model $p(d|\hat{\theta}_D)$, they compute the KL-divergence of document d with respect to query q as the final relevance score:

$$D(\hat{\theta}_{Q'}||\hat{\theta}_D) = -\sum_w p(w|\hat{\theta}_{Q'}) \log p(w|\hat{\theta}_D) + \texttt{cons}(q) \tag{2.17}$$

Again, in contrast to Zhai and Lafferty [2001b], Lavrenko and Croft [2001] compute the relevance score using the original query and the expanded query respectively, and then linearly combine the two scores. Thus the final query-document relevance $\hat{s}_d(q)$ is computed as,

$$\hat{s}_d(q) = \mu s_d(e) + (1 - \mu) s_d(q) \tag{2.18}$$

where $s_d(q)$ is the relevance between the original query q and documents d, and $s_d(e)$ is relevance between the expanded query terms e and document d.

2.4 Applying Topic Models For Query Expansion

Topic models capture the semantic relationships of words through learning the latent topics, which are presented as distributions over different words. Such semantic relationships among words provide a unique way to match or expand words in the semantic level rather than a direct spelling matching. As a result, topic models have been successfully applied into query expansion [Yi and Allan, 2009, Park and Ramamohanarao, 2009].

Smoothing Query Language Model The most intuitive way to apply topic models for query expansion is to extract the words' relevance from topics directly as Yi and Allan [2009]. They train a topic model, from which the probability $p_{\texttt{TM}}(k|q)$ of a topic k in a query q is learned. Then the query-word relevance $p(w|q)$ is computed based on topics as follows,

$$p(w|q) = \sum_k p_{\texttt{TM}}(w|k) p_{\texttt{TM}}(k|q) \tag{2.19}$$

This query-word relevance $p(w|q)$ from topic models can be used to smooth the original query language model through linear interpolation.

However, queries are normally too short to learn meaningful topics, thus the quality of query-word relevance is relatively limited. To improve the quality of extracted topics, Yi and Allan [2009] also train topic models from the relevant documents (e.g., top documents retrieved by a query), and extract the query-word relationships based on the Equation 2.19 for query expansion.

Improving Relvence Model In addition to this direct approach, Yi and Allan [2009] also apply topic models to improve the relevance model in Equation 2.16 for query expansion. In this approach, topic models are applied to capture the document-word relationship $p(w|D)$ given the query q as follows,

$$p_{\mathtt{TM}}(w|D,q) = \sum_k p(w|k)p(k|D,q) \tag{2.20}$$

where,

$$\begin{aligned} p(k|D,q) &= \frac{p(k,D,q)}{p(D,q)} = \frac{p(k)p(D|k)p(q|k)}{p(D,q)} \\ &= \frac{p(k|D)p(q|k)}{p(q|D)} \approx p(k|D)p(q|k) \end{aligned} \tag{2.21}$$

where $p(k|D)$ is the topic probability in document D, and $p(q|k)$ is the probability of generating a query q given the topic k. Then the topic-based document-word relationship $p_{\mathtt{TM}}(w|D,q)$ is applied to smooth the document-word relationship $p(w|D)$ in the relevance model (Equation 2.16) through a linear interpolation,

$$p(w|q) = \sum_{D \in \mathcal{C}} (\omega p(w|D) + (1-\omega)p_{\mathtt{TM}}(w|D,q))p(D|q) \tag{2.22}$$

where ω is a constant weight to combine the original relevance model and the topic-based relevance model. Because the topic models capture the word relationships on a sematic level, this improved relevance model better captures the query-word relationships and improve query expansion.

Learning Pair-wise Word Relationships Park and Ramamohanarao [2009] also apply topic models for query expansion but in a different way. They model the pair-wise relationship between words through topic models and then apply it for query exapansion. More specifically, based on the topics extracted from topic models, they compute the probabilistic relationships of each word pair (w_x, w_y),

$$\begin{aligned} p(w_x|w_y,\alpha) &= \sum_i p(w_x, z_i|w_y,\alpha) \\ &= \sum_i p(w_x|z_i, w_y,\alpha) p(z_i|w_y,\alpha) \\ &= \sum_i p(w_x|z_i,\alpha) p(z_i|w_y,\alpha) \end{aligned} \tag{2.23}$$

where $p(w_x|z_i,\alpha)$ is the topic-word distribution which can be learned from topic models, and $p(z_i|w_y,\alpha)$ can be computed as,

$$\begin{aligned} p(z_i|w_y,\alpha) &= \frac{p(w_y|z_i,\alpha)p(z_i|\alpha)}{p(w_y|\alpha)} \\ &= \frac{p(w_y|z_i,\alpha)p(z_i|\alpha)}{\sum_j p(w_y, z_j|\alpha)} \\ &= \frac{p(w_y|z_i,\alpha)p(z_i|\alpha)}{\sum_j p(w_y|z_i,\alpha)p(z_i|\alpha)} \end{aligned} \tag{2.24}$$

where $p(w_y|z_i,\alpha)$ is the topic-word distributions, and Park and Ramamohanarao [2009] also show that $p(z_i|\alpha) = \frac{\alpha_i}{\sum_k \alpha_k}$. As a result, we have,

$$\begin{aligned} p(z_i|w_y,\alpha) &= \frac{p(w_y|z_i,\alpha)\frac{\alpha_i}{\sum_k \alpha_k}}{\sum_j p(w_y|z_i,\alpha)\frac{\alpha_j}{\sum_k \alpha_k}} \\ &= \frac{p(w_y|z_i,\alpha)\alpha_i}{\sum_j p(w_y|z_i,\alpha)\alpha_j} \end{aligned} \tag{2.25}$$

The final probabilistic relationships of each term pair can be represented as,

$$p(w_x|w_y,\alpha) = \frac{\sum_i p(w_x|z_i,\alpha)p(w_y|z_i,\alpha)\alpha_i}{\sum_j p(w_y|z_j,\alpha)\alpha_j} \tag{2.26}$$

Once this term relationship is obtained, they choose the top related terms as the expanded terms e for the given query q, and the final document ranking score is computed as Equation 2.18.

Building Bilingual Topic Models Gao et al. [2012] further introduce the bilingual topic models [Gao et al., 2011] for query expansion. They assume the search query and its relevant Web documents share a common distribution of topics, but use different vocabularies to express these topics. Thus in their models, queries and documents share the same document-topic distributions θ^Q, but have different topic-word distributions ϕ_z^Q and ϕ_z^D respectively.

To generate a query q, a document-topic distribution θ^Q is drawn from a Dirichlet prior, and a topic z is sampled from θ^Q, then a query term q_i is sampled from ϕ_z^Q. To generate a document term, a topic z is firstly sampled from θ^Q, the same document-topic distribution as the query, and then a document term d_i is sampled from document topic-word distribution ϕ_z^D. In this way, documents and queries are conntected through the hidden topics, even though their vocabularies (topic-word distributions) are different. By summing over all possible topics, the relationship between document term e and query q can be computed as,

$$p(e|q) = \sum_z p(e|\phi_z^D)p(z|\theta_q) \tag{2.27}$$

The paramters can be learned by standard EM algorithm. The top related terms in the relevant documents can be used for query expansion. This method aligns queries and documents throught the document-topic distribution on a semantic level, which is very similar to the polylingual topic models [Mimno et al., 2009], except the latter aligns parallel documents in different languages. More details will be introduced in Chapter 8.

Getting Interactive Feedback Relevance feedback involves the users in the retrieval process to improve the ranking result set [Rocchio, 1971]. The basic idea is to ask users to give feedback on the relevance of documents in an initial set of retrieval results, and users' feedback on relevance is further used to improve the ranking results. This process can go through one or more iterations.

Andrzejewski and Buttler [2011] present a new framework for obtaining and exploiting user feedback at the latent topic level. They learn the latent topics from the whole corpus and construct meaningful topic representations. At query time, they decide which latent topics are potentially relevant and present the topic representations along keyword search results. When a user select a latent topic, the original query is expaned with the top words in this selected topic, and the search results are refined.

This direction is actually more related with topic labeling and interative visulization for topic models, which will be further discussed in Chapter 3.

2.5 Conclusion

Becuase topic models analyze documents on a semantic level, it offers an interesting and unique framework for modeling the document-word relations and word-word relations. As a result, topic models have been successfully applied in smoothing language models and query expansion, two main directions in information retrieval. This successful application on inforation retrieval is based on a general good understanding of the outputs of topic models. In the next chapter, we will introduce how to visualize and label the topics in general to help the users better understand the topics.

3

Visualization

While the previous two chapters have focused on algorithmic uses of topic models, one of the reasons for using topic models is that they produce human-readable summaries of the themes of large document collections. However, for users to use the results of topic models, they must be able to understand the models' output. This depends on *visualization* and *interaction* with the model.

We begin this chapter with a discussion of how best to show individual topics to users. From these foundations, we move to how we can display entire models—with many topics—to users. Finally, we close with how users can provide feedback through these interfaces to improve the underlying model.

3.1 Displaying Topics

Recall from the previous chapters that topics are distributions over words; the words with the highest weight in a topic best explain what the topic is about. While the simplest answer—just show the most probable words—is a common solution, there are possible refinements that can improve a user's understanding of a dataset by showing the

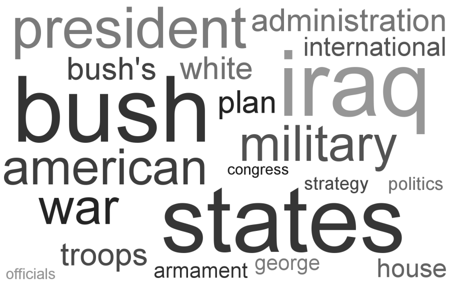


Figure 3.1: Word clouds use a 2D layout to show which words appear in a topic. Word size is related to its probability in the topic, showing which words are more prominent.

relationships between words or explicitly showing words' probability.

Word Lists Just showing a list of the most common words (a visualization that we'll call "word list") is very simple, and it also works well. Users can quickly understand what's going on, and it is an efficient use of space. represented horizontally Gardner et al. [2010b], Smith et al. [2015] or vertically Eisenstein et al. [2012], Chaney and Blei [2012b], with or without commas separating the individual words, or using set notation Chaney and Blei [2012b]. Smith et al. [2015] go further by adding bars representing the probabilities of the word.

Word Clouds Word clouds (e.g., Figure 3.1) are another popular approach for displaying topics. Unlike word lists, they also use the size of words to convey additional information. Word clouds typically use the size of words to reflect the probability of the words. This uses more of a given visualization area to be used to display a topic.

However, word clouds have been criticized for providing poor support for visual search Viégas and Wattenberg [2008] and lacking contextual information between words Harris [2011]; users can sometimes draw false connections between words that are placed next to each other randomly in a word cloud. Another alternative is to use word associations to layout words [Smith et al., 2014]; Figure 3.2 shows places

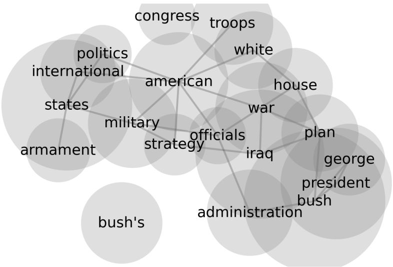


Figure 3.2: A topic-in-a-box visualization for topics—like a word cloud—shows words in a 2D context. However, it uses local co-occurence to decide which words to place next to each other.

words that appear together next to each other in the visualization.

3.2 Labeling Topics

Throughout this survey, we've been referring to topics about information technology or about the arts. These are convenient labels, but completely removed from the raw distribution over words. Thus, it's often useful to assign labels to topics within an interface.

In contrast to the previous *visualization* approaches, labeling focuses son showing not the original words of a topic but rather a clearer label more akin to what a human summary of the data would provide.

Approaches for automatic labeling can be divided into those that only use internal information from the topic model against those that also use external knowledge resources. While purely internal methods are more robust and consistent with the philosophy of unsupervised topic models, external resources often produce higher quality labels.

Of the techniques that use external resources, we further separate those that use direct supervision for labeling (i.e., knowing what constitutes a good labeling) from those that use general knowledge resources such as Wikipedia or knowledge bases.

Internal Labeling Mei et al. [2007] propose an internal labeling method that takes prominent phrases from the topic and compares how consistent the phrase's context is with the topic distribution. Phrases whose contexts closely resemble the topic often appear in regions of text that summarize the document, making them good candidates for labels. Mao et al. [2012] extend the technique to hierarchies, using the insight that parents' labels should be consistent with their children's.

Labeling with Supervised Labels Lau et al. [2010] use a supervised approach to rerank the words in a topic to ensure that the "best" word in the topic is shown to a user. Each candidate word forms a feature vector consisting of features such as the following:

- the conditional probability of a word given the other words in a topic (which implies topic coherence, as discussed in Section 3.4);
- whether the word is a hypernym of other words in the topic (e.g., "dog" in a topic that also contains "terrier" and "poodle"); and
- the original probability of the word in the topic.

While these can be used alone as an unsupervised reranking, Lau et al. [2010] use user-selected best topic words to weight which of these features are most important for selecting the best topic word. These weights are learned using support vector regression. Lau et al. [2011] extend their technique by adding candidates from Wikipedia to the set and show that models learned on different domain corpora are still effective.

Labeling with Knowledge Bases Mao et al. [2012] align topic models with an external ontology of labels. They argue that labels should match topic words (as labeling with flat topics); a topic's words should be consistent with a labels' children in the hierarchy; and the topic's labels should be unique.

Aletras et al. [2014a] instead query the whole web and then build a graph that includes the words that make up the titles of the retrieved webpages. The edges between the words is the NPMI computed on a reference corpus. The intuition is that words that are "central" in this graph will be a good title for the topic. They find the central words

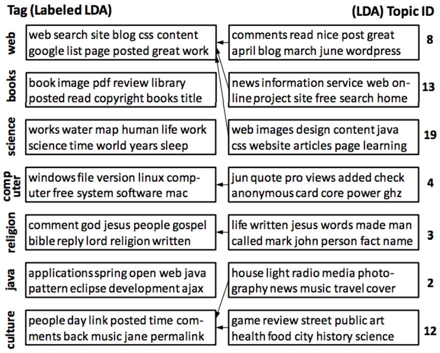

Tag (Labeled LDA)
(LDA) Topic ID
web
web search site blog css content google list page posted great work
comments read nice post great april blog march june wordpress
8
books
book image pdf review library posted read copyright books title
news information service web on-line project site free search home
13
science
works water map human life work science time world years sleep
web images design content java css website articles page learning
19
comp uter
windows file version linux comp-uter free system software mac
jun quote pro views added check anonymous card core power ghz
4
religion
comment god jesus people gospel bible reply lord religion written
life written jesus words made man called mark john person fact name
3
java
applications spring open web java pattern eclipse development ajax
house light radio media photo-graphy news music travel cover
2
culture
people day link posted time com-ments back music jane permalink
game review street public art health food city history science
12

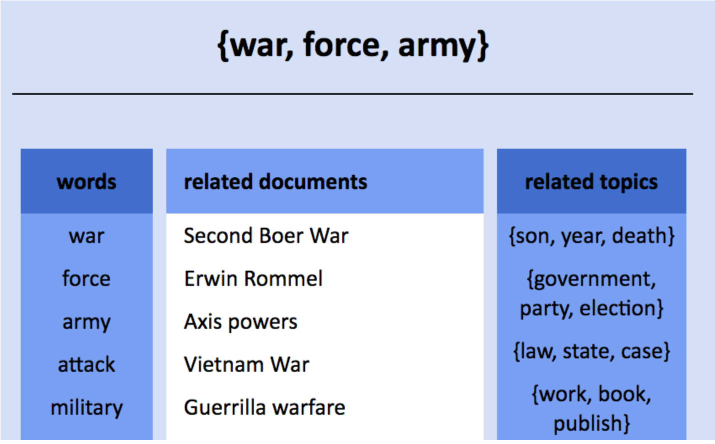
Figure 3.3: Example of topics learned by labeled LDA (Figure from Ramage et al. [2009]). Each topic in labeled LDA is associated with a label, which encourages the topics to be consistent with the ontology of labels. LDA, in contrast, uses the empirical frequency of topics to divide the dataset, resulting in three topics (8, 13, 19) associated with the labeled LDA web topic.

by using the PageRank [Page et al., 1999] algorithm. This finds words that are highly probable in the topic and appears often with many other words in the topic.

Using Labeled Documents The task of associating labels with topics becomes much easier if many of your documents are themselves labeled. Labeled LDA [Ramage et al., 2009] associates topics to each of the labels and forces labeled documents to only use the topics associated with the document. This constraint forces the topics to be consistent with the original labels (Figure 3.3). Bakalov et al. [2012] extend this to hierarchical label sets (e.g., NY Times subjects that place Russia under International), while Nguyen et al. [2014] extend it to learning hierarchies of topics from unorganized labels, learning that ska topics are a kind of music without provided links.

3.3 Displaying Models

However, topics are not the end of the story. Users often want to use topics to find relevant documents within the collection. Going back



{war, force, army}

words	related documents	related topics
war	Second Boer War	{son, year, death}
force	Erwin Rommel	{government, party, election}
army	Axis powers	{law, state, case}
attack	Vietnam War	{work, book, publish}
military	Guerrilla warfare	

Figure 3.4: The Topic Model Visualization Engine [Chaney and Blei, 2012a] shows the most related documents to a topic along with related topics.

to our example in the previous chapter, a user may want to find the "smoking gun" in the Enron dataset, not just use topics to understand the main themes in a dataset.

Thus, a good topic model visualization must also show the documents associated with a topic. The Topic Model Visualization Engine [Chaney and Blei, 2012a] shows the top documents associated with a topic (Figure 3.4). Recall that each document has a distribution over topics θ_d, which is a vector with an entry for each topic. We focus the dimension associated with a particular topic and then sort the documents based on that topic coordinate from largest to smallest.

The topical guide [Gardner et al., 2010a] extends this approach by enriching topic views with additional metadata. For instance, if the collection has dollar amounts or sentiment Pang and Lee [2008] associated with a document, it provides a histogram of the metadata associated with the topic. It also provides *in context* examples of topic words, allowing to see how a word is used within a topic (helping to address topic model's bag of words assumptions).

TOME [Eisenstein et al., 2014] focuses on a specific type of metadata: time. It allows users to view the evolution of topics over time

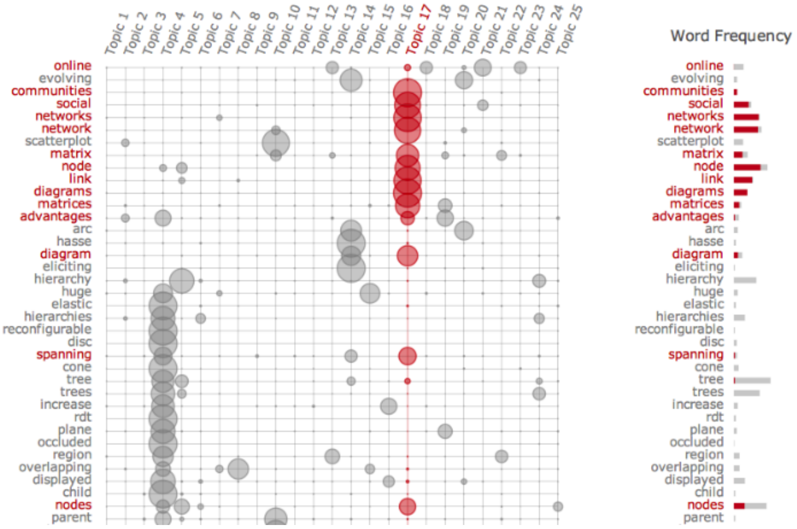


Figure 3.5: The Termite visualization of topics helps reveal which topics use similar words and are thus likely talking about similar things.

to understand, for example, how the issue of slavery is reframed from an economic argument to an argument over human rights. It supports filtering to specific topics or to see how words are used over time across topics.

In contrast to showing how topics related to metadata, Chuang et al. [2012] focus on how topics relate to *each other*. Their "Termite" topic visualization (Figure 3.5) shows the term-by-term similarity between topics. By presenting topic-term probabilities on a grid with topics as the columns and terms as the rows, users can see when topics share words or when topics are only about a handful of words.

3.4 Quality, Stability, and Repair

However, not all topic models are perfect. Chang et al. [2009] showed that held-out likelihood, the traditional measure of probabilistic model quality, emphasizes *complexity* rather than interpretability, what humans presumably care about. Thus, automated techniques may not be able to tell you whether a topic model is good or not.[1]

Visualizations can help show users where topic models have issues. Showing the relationships between multiple models can also help distinguish stable from spurious topics [Chuang et al., 2015], and adjusting the "hyperparameters" of distributions (the Dirichlet parameters of models discussed in Chapter 1) can have a large effect of what the final models are Wallach et al. [2009].

Interactive topic modeling—in conjunction with visualizations—can help correct the problems of topic models. A user first get an overview of the dataset using a visualization of the topics and documents. Then, the user can see instances where the model errs and then correct those mistakes.

For example, Figure 3.6 shows a topic learned from abstracts of grants funded by the American National Institutes of Health (NIH, discussed more in Chapter 6.1). Most topics were "good"; they summarized the data and told a story about a coherent slice of research supported by the NIH. However, this topic is more problematic; it combines words about the central nervous system with words about the urinary system. Such a topic (as discussed in Mimno et al. [2011]) does not give a clear understanding of the documents it should represent.

Hu et al. [2014a] address this problem by allowing a user to add probabilistic constraints to the model [Boyd-Graber et al., 2007, Andrzejewski et al., 2009]. For example, the user might say that "bladder" and "spinal cord" don't belong in the same topic together. Figure 3.6 shows how the topic is more focused after the user provides this feedback. In contrast to probabilistic constraints, Choo et al. [2013] add matrix factorization constraints, which are in practice much faster.

3.5 Conclusion

Much of what topic models are used for is to help users understand corpora. However, the output of topic models don't give insights to users without the helpmeet of interactive visualizations which allow users

[1] Automated measurements [Newman et al., 2010, Mimno et al., 2011, Aletras et al., 2014b] of topic quality may serve as a proxy for human interpretability ratings.

Topic Words (before)	Topic Words (after)
bladder, sci, spinal_cord, spinal_cord_injury, spinal, urinary, urinary_tract, urothelial,injury, motor, recovery, reflex, cervical, urothelium, functional_recovery	sci, spinal_cord, spinal_cord_injury, spinal, injury, recovery, motor, reflex, urothelial, injured, functional_recovery, plasticity, locomotor, cervical, locomotion

Figur[illegible]: Before and after topics with iteractive topic modeling from Hu et al. [2014a]. Initially, this topic conflates two topics (urinary and central nervous system), which is undesierable. Adding a constraint that the words “bladder” and “spinal cord” shouldn’t appear together in a topic makes the topic more coherent and discovers concepts that weren’t present before.

to discover and refine insights. In the next chapters we’ll talk about specific applications of these insights, but these insights are often built on the initial understanding of a model offered by the visualizations discussed in this chapter.

4

Historical Documents

Topic models play an important role in the analysis of historical documents. Historical records tend to be extensive and difficult to manage without intense and time-consuming organization. Records are complicated: they resist categorization, and may even lack standard spelling and formatting. But there is more to history than simply the management of documents. The task of a historian is not only to absorb the contents of historical records, but to generalize; to find patterns and regularities that are true to the documents, but also beyond any single piece of evidence. Topic models are useful because they address these issues. They are scalable, robust to variability, and able to generalize while remaining grounded in observation.

Automated methods are an especially valuable counterpoint to traditional close reading methods. Studying history is about encountering the unexpected, often in contexts that seem familiar. We don't necessarily know how people in the past talked about particular issues, or how they organized their lives. Perhaps more dangerously, we assume that we know these things, and that our ancestors saw the world in the same way we do. Topic models give us a perspective that is interpretable but at the same time alien, based on patterns in documents and not on our own conceptions of how things should be.

Time is a critical variable in the study of historical documents. Although many modern collections have a significant aspect of time variation (see for example scientometrics), time is a defining element of historical research. Collections of historical documents are necessarily situated in a time other than our own, but also tend to cover long periods — decades or even centuries. As a result, many of the examples cited in this chapter organize documents along a temporal axis. The associated analysis is particularly concerned with how language, as reflected in topic concentrations and topic contents, changes over time.

This chapter is organized around different formats for historical documents. A recurring focus is the desire to plot events and discourses against time. We begin with historical newspapers, which are relatively close to the modern news articles that are a more familiar use case in topic modeling. We then consider other forms of historical records, such as annals and diaries. These demonstrate the flexibility of topic modeling, including a corpus not in English and corpus in English with irregular spelling. Finally, we consider studies of historical scholarly literature.

4.1 Newspapers

Newman and Block [2006] present an example of topic modeling on historical newspapers, in a collection of articles from the *Pennsylvania Gazette* from 1728 to 1800.[1] These articles comprise 25 million word tokens in articles and advertisements, and cover several generations of everyday life before, during, and after the founding of the United States of America. The authors contrast their study to manually created keyword-based indexes, which focus on specific terms and can be applied inconsistently across large corpora. Spurious patterns in index term use could complicate historical research. They cite an example of the tag *adv*, which is used extensively in the early and late decades of the corpus, but not in the middle. The topic-based approach is at-

[1] http://www.accessible-archives.com/

tractive because it is consistent across the collection (as long as the terms used in the documents are themselves consistent) and because it operates at a more abstract semantic level, reducing the chance that modern historians miss key terms.

They compare three methods for finding semantic dimensions, latent semantic analysis [Deerwester et al., 1990], k-means clustering, and a topic model [Hofmann, 1999a]. The difference between these methods can be described in terms of expressivity. LSA is effective at embedding word types and documents in a low-dimensional space, but the individual dimensions of this space are not interpretable as themes. LSA is too expressive: it places no constraints, such as positivity, on the learned dimensions, and therefore produces uninterpretable results that nevertheless fit the document set well. The k-means clustering is more similar to the topic model, and more successful at finding recognizable themes. But it is also prone to repeating similar clusters with small variations. Because of the single-membership assumption (a document can only belong to one cluster), the clustering model cannot represent documents with varying combinations of somewhat independent themes. The k-means model is therefore insufficiently expressive: it it forced to "waste" clusters on frequent combinations of simpler themes. The topic model, in contrast, has both modeling flexibility along with sufficient constraints to support interpretable results.

The authors find that the learned topics are a good representation of dynamics in the corpus, although not always in a direct manner. There is a large increase in discussions of politics in the period immediately around the American Revolution (*state government constitution law united power*). There is also evidence of economic factors: a topic relating to descriptions of cloth (*silk cotton ditto white black linen*) rises in the 1750s, but then declines as Americans turned to domestic "homespun" cloth production in response to British trade policies. Other topics point to more subtle changes in language. A topic that is less immediately interpretable (*say thing might think own did*) corresponds to a series of long "public letters" that contain more academic "argument making".

Nelson[2] studies topics in Civil War-era newspapers, including the Confederate paper of record, the Richmond Daily Dispatch. Like Block and Newman, Nelson's goal is to organize the collection into themes and to measure the variation in prevalence of those themes over time. The web interface highlights a temporal view of the collection as a series of topic-specific time series. The mode of analysis is neither fully automated nor manual, but rather combines the two approaches. Nelson manually labels the topics and groups them into larger categories such as "slavery", "nationalism and patriotism", "soldiers", and "economy".

He validates the model by comparing topics to a known and previously annotated category, the "fugitive slave ads". These documents were pre-photographic descriptions of runaway slaves, and have a specific language consisting of aspects of personal appearance and possible locations where enslaved people might have hidden. He finds a near perfect correspondence between the prevalence over time of manually labeled fugitive slave ads and documents that have a high concentration of a specific topic, which places high probability on terms such as *reward, years,* and *color* (manual labels were not used in training the model). Nelson notes that few if any of these documents are assigned completely to this topic: he uses a cutoff of 21.5% as a criterion.

Nelson's larger-scale groupings of topics pick out threads of discourse that may or may not be correlated over time. The model identifies three topics that have similar temporal distribution, peaking at the beginning of the war in 1861 and largely disappearing afterwards. These are related but distinct themes: anti-Northern sentiment expressed in poetic form, anti-Northern sentiment expressed in vitriolic prose, and discussion of secession. All three form aspects of the same process, the rhetorical push for war. Other related topics have slightly different temporal distributions. Nelson groups six topics related to soldiers, and displays them in the order of their maximum concentration over time. They move from "military recruitment" and "orders to report" to later topics related to "deserters", "casualties", and "war prisoners." Again, these are related themes but rather than comprising a single event they trace the development of the increasingly dire military situation of the

[2] Mining the Dispatch, http://dsl.richmond.edu/dispatch/

Confederacy.

Yang et al. [2011] model a collection of historical newspapers from Texas spanning from the end of the Civil War to the present day. The goal is both exploratory, to find out about the interests of Texans through the 19th and 20th centuries, and *semi-exploratory*, to find out more about the history and context of specific, pre-specified themes such as cotton production. In the topic model setting, semi-exploratory analysis starts by identifying one or more topics that seem to correspond to the theme of interest, and then using those topics as a axis of investigation into the corpus. For example, a historian considered documents that exhibit topics related to *cotton*, and the topics that cooccur in those documents. The study also led to more fully exploratory results. A topic related to the battle of San Jacinto, the final conflict in the Texas Revolution that led to separation from Mexico, appeared earlier than expected. Further investigation suggested that the significance of the pivotal battle of San Jacinto was established much earlier than historians had previously anticipated.

The Texas newspaper study raises several interesting methodological issues relating to pre-processing and iterative modeling. The authors put considerable work into dealing with the quality of digitization. There are many factors that affect the quality of digitized historical newspapers, from the quality of the original printing to scanning, article segmentation, and optical character recognition (OCR). For this study extensive work was applied to automated spelling correction. Another notable factor in this study is its prominent use of multiple topic models. In many cases there is a tacit assumption that a single corpus should result in a single model, but in practice modeling is often iterative, and intimately bound to the development of pre-processing systems. Yang et al. [2011] train different models on different temporal slices of the corpus. Although there is some advantage to maintaining a consistent topic space over time, dividing the corpus into separate sections has certain advantages. In this case, historians were interested in the context of specific historic periods, such as the full run of a a newspaper in one of several pivotal years, that are smaller than the full corpus but yet too large to be read easily. The authors also describe an iterative workflow that involves comparing topic model output after each of several pre-processing steps. Topic models are often effective at identifying consistent data-preparation errors, such as end-of-line hyphenation and consistent OCR errors.

4.2 Historical Records

Other types of records besides newspapers are of interest, and present their own challenges. In this section we consider two case studies, in which the simplicity of the bag-of-words document model is an asset because it allows for substantial variability in spelling and language, both in English and in other languages.

Miller [2013] uses Chinese records to investigate the meaning of the word *zei*, or "bandit" in Qing dynasty China. The word by itself can imply several different forms of anti-social behavior, which are difficult to distinguish from word frequencies alone. A topic model uses contextual information to separate these effects.

The application of topic models in Chinese highlights the importance of tokenization. We usually receive documents in the form of long strings, but we are interested in identifying *tokens* that are short strings with a specific meaning. Breaking a document into distinct tokens is an often-overlooked part of the document analysis process. In European languages we can achieve good results simply by separating strings of letter characters from sequences of non-letter characters, although there are many special cases. Tokens may contain non-letter characters such as apostrophes and hyphens, and may span multiple words (*Queen Victoria*, *black hole*). In many East Asian writing systems we cannot rely on orthographic conventions to identify tokens. Miller argues that in Classical Chinese a single character can be treated as a token without a strong negative impact on modeling, but for Japanese and modern Chinese we must often rely on pre-processing tools that are themselves potentially unreliable.

Cameron Blevins[3] models the diary of Martha Ballard, a revolutionary war-era midwife who recorded entries over 27 years. The model

[3] http://www.cameronblevins.org/posts/topic-modeling-martha-ballards-diary/

provides a useful way to discover connections between words and repeated discourses. As with other historical corpora, Blevins focuses on the connection between topics and time. Specific events, like a birth, can be highlighted by looking at spikes in a certain topic in the day-to-day time series. But larger trends are also evident. As a calibration experiment, Blevins measures the association of a topic that appears to refer to cold weather (*cold, windy, chilly, snowy, air*) to months of the year. As expected, the concentration of this topic is lowest from May to August, rises from September to January, and falls from February to April.

Blevins identifies several other topics that appear to change in their concentration over time. Two topics involving house work, focusing roughly on cleaning and cooking, respectively, appear to be correlated in time, and rise over the decades. Blevins connects this finding to suggestions that as Ballard grew older and her children moved away, she had less help from family members. A more subtle topic involves descriptions of fatigue and illness. This topic also increases over time, and appears to correlate with the housework topics, except in the last year of the diary, where fatigue and illness reach their highest concentration and housework declines.

This analysis exemplifies the exploratory nature of topic modeling: by themselves, these observations are not conclusive, but they are suggestive and point to areas of further analysis. A scholar might take the diary entries that score high on an individual topic as a reading list, and determine how well a particular automatically detected discourse maps to themes in Ballard's personal experience. For example, one might check whether Ballard's references to fatigue and illness are referring to herself or to patients. The model does not tell the whole story, but it points to where stories might lie.

Blevins argues that characteristics of the diary form make it well-suited for topic analysis: "Short, content-driven entries that usually touch upon a limited number of topics appear to produce remarkably cohesive and accurate topics." In addition, the topic model's lack of linguistic sophistication is actually an asset in this case. The diary is written in a terse style with many abbreviations and with irregular, 18th century spelling: “mrss Pages illness Came on at Evng and Shee was Deliverd at 11h of a Son which waid 12 lb.” Models trained on modern text corpora might not even recognize this example as English, but the topic modeling algorithm is still capable of finding semantically meaningful groups of words.

4.3 Scholarly Literature

The historical record of scholarship is a valuable source for intellectual history. Many users make use of the JStor “Data for Research” API.[4] The DFR API is an important example, because it provides access to articles that have been scanned by JStor and may be under copyright. Access to the underlying documents in their original form as readable sequences of words may be restricted for legal or commercial reasons. DFR provides a simple view into selected articles by only providing the frequency of word unigrams. While the bag-of-words assumption used by topic models is restrictive, in this case it can be an advantage, because the original sequence of words is not used for inference anyway.

Mimno [2012] studies a collection of Classics journals digitized by JStor to detect changes in the field over the 20th century. A distinctive aspect of this study is the use of a *polylingual* topic model Mimno et al. [2009]. An English-language journal is compared to a German-language journal by learning a common set of topics that each have a vocabulary in both languages. In other words, a topic has two “modes”, one in which it emits words drawn from a distribution over English terms, and another in which it emits words drawn from a distribution over German terms. The linkage between English and German words is constructed using Wikipedia articles. Wikipedia articles exist in many different languages, and articles in one language often link to comparable articles in another language. The author first selects English Wikipedia articles matching key terms in the English-language journals, and then collects the German Wikipedia articles that are listed as being comparable to the selected English-language articles.

By training the topic model jointly on the combined corpus of the

[4] http://dfr.jstor.org/

original journal articles and the comparable Wikipedia articles, the model provides insight into the relative concentration of scholarly interests across the two language communities. The German-language journal articles contain relatively more work on law and oratory, themes that are present in the English-language articles but less prevalent. The model also shows a large increase in interest in poetry in the German journal in the period following the second world war. In the English journals there is a large increase starting in the 1980s in cultural and economic studies along with critical theory, which does not appear in the German journals.

Goldstone and Underwood [2014] use a topic model as a tool to structure an exploration of a corpus that spans more than a century. They are interested both in changes at the topic level and at the level of word use within topics. For these authors the appeal of topic modeling is that models are better able to represent contextual meaning than simple lists of keywords. They write that "[t]he meanings of words are shifting and context-dependent. For this reason, itâĂŹs risky to construct groups of words that we imagine are equivalent to some predetermined concept."

They analyze the proceedings of the Modern Language Association to find shifts in focus in the field of English literature. A model trained with 150 topics on 21,000 articles identifies a topic associated with descriptions of violence: *power, violence, fear, blood, death, murder, act, guilt.* Using a temporal plot they argue that the concentration of this topic is greater in the second half of the 20th century than during the first half. They contextualize this finding by comparing the frequency of these words in a more general corpus from Google ngrams; there is no comparable change. This approach holds topic fixed and searches for associated words. They then pivot and hold the word "power" fixed and search for associated topics. In this case the *violence* topic actually appears to be relatively stable in its association with the target word. The largest increase is in a different topic characterized by the words *own power text form*, in which context it appears almost exclusively after 1980.

In this chapter we have focused on works in which the temporal axis is of primary concern. When we consider newspapers, historical records, and historical scholarly journals we are looking not just for the topical foci of each time period, but how those topics shift in concentration as they are influenced by historical events. Our consideration of scholarly journals leads directly into our next chapter on the study of a much larger and temporally variable literature, the study of science.

5

Fiction and Literature

This chapter considers documents that are valued not just for their information content but for their artistic expression. There are many different ways to read fiction, poetry, and rhetoric, and the choice of how we read affects the type of conclusions that we are able to make. Scholars have traditionally focused on a “close reading” approach, in which the goal is to identify the specific features of a passage that convey a more general meaning, or emotion, or atmosphere. These features might include nuances of word selection, echoes of sound through rhyme or alliteration, or prosodic features like rhythm or cadence.

5.1 The Role Topic Models Play in the Humanities

While close reading is a foundational tool in the study of literature, it is necessarily limited by its scale. We value literature because it is one of the best ways to capture the spirit of an age, and the experiences of those who lived through it. But standard close reading methods require narrow focus and thorough interpretation. Topic models complement close reading in two ways, as a survey method and as a means for tracing and comparing large-scale patterns.

The survey method is relatively simple, linking passages that a reader may not have known about. Close reading is the best way to analyze a short passage of text, but which short passages of text do we want to analyze? Because no one can read — much less close read — all the available material from a culture or time period, scholars are often left trying to make large-scale arguments about the history of literature from small-scale evidence. And this small-scale exploration is not randomly selected: the same small canon is studied in detail while the vast proportion make up the "great unread" [Moretti, 2000], works that are never studied. Identifying broad themes and then mapping those themes to their realization in different contexts may reveal works or sections of works that are "hiding in plain sight," unknown to modern scholarship simply through obscurity.

An alternative, and less traditional, mode of analysis is often referred to as "distant reading" [Moretti, 2013a]. This approach uses computer-assisted methodologies. Topic modeling has emerged as a central tool in distant reading, as a way to organize our reading of large scale patterns.

Topic analysis, viewed as a way of identifying repetitions of language or discourse through multiple works, resonates with many more familiar approaches to the study of literature. At the broadest scale, to define a genre or a literary period is to separate a corpus into sections based on some observable criterion. We posit a "gothic" literature characterized by atmospheric descriptions of castles, or a "cyberpunk" literature characterized by conflicted relationships with information technology. At a smaller scale, themes or tropes reappear in different contexts. At the most detailed level, scholars identify repeated phrases, such as the descriptive epithets used in Homeric oral poetry.

Statistical topic analysis has a similar goal, but pursues it through different means. Rather than rigid boundaries specified by date of publication or nationality, algorithms identify genre through the repeated words that form the traces of those themes. Topics do not represent themes themselves, but rather identify the implicit statistical regularities in word use brought about by the presence of genres, themes, and discourses.

Applying topic models to fiction, however, brings new challenges. Jockers [2013] trains a 500-topic model on a corpus of 4000 English-language novels. Several issues emerge from this corpus. These are present in other contexts, but they are much more readily apparent in fiction.

5.2 What is a Document?

In most literature about topic models, the term "document" is used on the implicit assumption that users have things called documents. In the canonical LDA article Blei et al. [2003], this word is used 143 times, but never defined. In many cases, the meaning of a "document" is fairly clear: a news article, or a scientific abstract. What was not clear in this earlier work was that this definition can be problematic, especially for documents longer than a few pages of text.

Treating novels as a single bag of words, for example, does not work. Topics resulting from this corpus treatment are overly vague and lack thematic coherence. We should not be surprised by this finding. The assumption of a topic model is that the concentration of topics over a document is fixed and unchanging from the beginning of a document to the end. Natural writing rarely fits the topic model assumption, and a novel that had no thematic variation over its entire length is unlikely to have been published.

We need to find a good segmentation into shorter contexts. We assume that themes are expressed in different sections of a long document like a novel. If a segmentation does a good job of identifying the boundaries between these sections, each resulting segment should have relatively few themes. If a segmentation does not do a good job of identifying boundaries, we should see segments that contain more themes on average, because our segments combine fragments of multiple thematic segments.

Jockers [2013] chooses to avoid relying on structural markers such as chapter divisions and divides novels into 1000-word chunks. This treatment results in coherent, tightly focused topics that can be reasonably used as proxies for recognizable themes.

Although fixed-length segmentation is effective, it is not necessarily ideal. Algee-Hewitt et al. [2015] compare varying fixed-length segmentations to segmentation based on paragraphs. They evaluate the difference between treatments by measuring the concentration of topics in each segment of text after modeling. The Herfindahl index is a measure of concentration in discrete probability distributions, calculated as the sum of the squared probabilities of each possible value. When a corpus of 19th-century novels is divided by paragraphs, the Herfindahl index over concentration of topics within each segment is consistently larger than the same index calculated when the same corpus is divided into evenly sized 200-word slices. Setting the slice size to the average length of paragraphs in the corpus, 82 words, increases the Herfindahl concentration metric, but the resulting value is still smaller than the value based on paragraphs.

This result is reassuring, in that it suggests that paragraphs do indeed have some consistent meaning, at least in this collection of 19th-century novels.

5.3 People and places

Because most works of fiction are set in imaginary worlds that have no existence outside the work itself, they are often characterized by words such as character names that are extremely frequent locally but never occur elsewhere. This word co-occurrence pattern is problematic for topic models because they can be thought of as machines for finding groups of words that occur frequently together and not in other contexts. Character names are — by that criterion — a perfect topic. Modeling these documents can result in topics that are essentially lists of character names.

As an example, consider a model of 14 novels by Charles Dickens. The top 20 words from a selection of topics ($K = 50$) are shown in Table 5.1. Upper-case letters are not reduced to lower-case in order to emphasize the presence of proper names. Several topics are dominated by capitalized names, with individual novels clearly identifiable: Topic 4 is *Oliver Twist*, Topic 5 is *Nicholas Nickleby*, Topic 6 is *The Pickwick*

Papers and Topic 7 is *A Tale of Two Cities*. In fact, exactly half of the distinct words in the top 20 words for all topics are capitalized, and almost all of these are proper names.

Focusing on characters is not always uninformative, and can in some cases highlight structure within works. Topics 1–3 all refer primarily to *Bleak House* (with the exception of *Scrooge*), but focus on different interlocking subplots. The first focuses on Lady Dedlock, the second on Mr. Jarndyce and his two wards, Richard and Ada, and the third on the investigations of the detective Mr. Bucket. The plot centers around the revelation of the connections between these apparently unrelated groups.

Jockers [2013] approaches this problem by constructing a stopword list that removes all character names before modeling. There are many ways to construct such lists. Lists of common names are a good start, but may not be aligned with a specific corpus. Some languages mark proper names with orthographic conventions like capitalization, but these tend to be noisy. A useful heuristic in English is to identify terms that appear capitalized in more than 90% of instances. Even then, names that are also common words, such as *daisy* and the aforementioned Mr. Bucket, or words that appear capitalized for other reasons, such as *god*, may lead to unintended results. Furthermore, some languages do not differentiate letter cases (Hebrew, Korean) and others use it for other purposes (all nouns in German). Named-entity recognition tools scan text for patterns of language that indicate personal names, and may result in greater precision than simpler methods. Nevertheless, there is no known way to avoid careful consideration of the meaning of words in context.

Novels describe people and places, but they are also created by people (authors) who are influenced by their cultural setting. Jockers and Mimno [2013] perform a post-hoc analysis on Jockers' earlier 500-topic model to determine whether there is a connection between the use of specific topics and metadata variables such as author gender, author nationality, and year of publication. They find that the concentration of many topics is strongly correlated with author gender, and that these correlations are statistically significant. Such significance testing can be

Table 5.1: Sample topics from Charles Dickens novels, without removal of character names (ordered manually).

1	Lady Leicester Scrooge Dedlock Rouncewell ladyship Wold Chesney Ghost Volumnia Christmas Tulkinghorn family Spirit Baronet nephew Rosa Scrooge's housekeeper Lady's
2	Richard Jarndyce guardian Ada Charley Caddy dear Skimpole Miss Summerson Esther Jellyby miss Vholes Kenge Woodcourt quite myself Guppy Chancery
3	says George Bucket Snagsby Guppy returns Smallweed Bagnet comes Tulkinghorn looks takes trooper does makes friend goes asks cries Chadband
4	Oliver replied Bumble Sikes Jew Fagin boy girl Rose Brownlow dear gentleman Monks Noah doctor Giles Dodger lady Nancy Bill
5	Nicholas Nickleby Ralph Kate Newman replied Tim Mulberry Mantalini Creevy brother N oggs Madame Gride Linkinwater Smike Arthur rejoined Wititterly Ned
6	Pickwick Winkle replied Tupman Wardle gentleman Snodgrass Pickwick's Perker fat boy Bardell dear Jingle inquired Fogg Dodson friends friend lady
7	Lorry Defarge Doctor Manette Pross Carton Darnay Madame Lucie Monseigneur Cruncher Jerry Stryver prisoner Charles Monsieur Tellson's Marquis father Paris
8	coach uncle gentleman lady box coachman gentlemen landlord get London guard inside horses waiter boys mail passengers large better hat
9	street door streets windows houses room window few iron walls wall rooms dark within shop doors corner small stood large
10	money letter paper business read pounds papers five hundred office thousand clerk paid years pen next law desk letters week

carried out by randomization and bootstrap tests. Both methods create "fake" corpora that are similar to the real corpus but different in specific ways. Randomization or permutation tests randomly shuffle the assignment of labels (such as author gender). If an observed correlation between a topic and an external variable is within the range of the correlations generated by randomly assigning documents to labels, there is little statistical evidence that that observed correlation is meaningful. Bootstrap tests preserve the relationship between documents and metadata variables, but resample documents with replacement. This test indicates whether a result depends on the presence or absence of a specific document. If there is wide variation between randomly generated corpora, the observed correlation may be the result of unusual outliers rather than a consistent pattern.

While the use of statistical hypothesis testing methods is potentially valuable in the context of large-scale distant reading, a literary analysis is not — and should not be — like a clinical trial. It is important to note some differences between their use in a scholarly context and their use in more typical scientific studies. First, the presence of unusual outliers or singular examples can in fact be a positive result. The suggestion that a particularly work may be radically different from supposedly similar examples could be the beginning of a new perspective. At the very least, it can identify editing and curation issues. Second, a critical variable in an analysis of statistical significance is sample size. Unlike a designed experiment, this sample size is usually not within our control: we have the literature that we have. Finally, it is vitally important to avoid the impulse to treat a significance score as a binary valid/invalid result. If numeric scores should be used at all, they should be presented as a "level of support" given the documents that are available. Humanists may also be fundamentally more comfortable with dubious hypotheses: an observed association with a 10% chance of being purely random could still be a very strong result.

As an example, Jockers and Mimno [2013] evaluate an intriguing hypothesis, that a topic about religious foundations (*Convents and Abbeys*) is used more by unknown authors[1] than by either (known) male

[1] Authors unknown to modern scholarship, not authors publishing under known

or female authors. The conjecture is authors were choosing to remain anonymous in order to write about politically and religiously touchy subjects. This correlation, however, showed large variability under a bootstrap test, and indeed one of the supposedly anonymous works turned out to be an abridgment of an Anne Radcliffe novel. The pattern is still present without the effect of these works, but there is not a clear and undeniable association between anonymity and the questioning of religious authority.

In some cases fiction is set in the context of real places. Tangherlini and Leonard [2013] look at nested models of sub-corpora within Danish literature in a way that highlights connections between real-world events and cultural movements and fictional echoes. Their method, which they describe as a topical "trawl line," uses a user-specified subcorpus as a query and then searches the remainder of the corpus for works that match to that query. As examples, they find works influenced by the translation of Charles Darwin into Danish, works influenced by the "Modern Breakthrough", and works influenced by folklore and regional literature.

5.4 Beyond the Literal

One of the hallmarks of fiction and literature is the use of figurative language. It is not obvious that unintelligent machines with no cultural understanding would have any ability to process such metaphors. However, Rhody [2012] demonstrates on a corpus of poetry that although topics do not represent symbolic meanings, they are a good way of detecting the concrete language associated with repeated metaphors.

Specifically, Rhody explores a corpus of 4500 poems that describe works of art (or *ekphrastic* poems). She trains a 60-topic model, and highlights several particularly interesting topics. One of these topics places high probability on *night, light, moon, stars, day, dark, sun, sleep, sky, wind, time, eyes, star, darkness, bright.* The apparent meaning of the topic is clear, and well summarized by the single top word: night. But Rhody finds that when she explores the *context* of this topic,

pseudonyms.

the poems are all using a consistent metaphor relating night and sleep to death. The concept of death does not appear in the top words — poets are not addressing the issue directly. Nevertheless, the model has identified an example of non-literal, figurative language even though, because it is grounded in the actual words, it has no ability to represent what the poets actually mean. The model is able to do this because the poets are using a consistent "surface" language to represent a consistent metaphor. The metaphor is not detectable directly, but a poet's use of a metaphor has a signature that is observable.

Rhody highlights a second topic that provides an example of a different type of non-literal meaning. This topic places high probability on *death, life, heart, dead, long, world, blood, earth, man, soul, men, face, day, pain, die.* Unlike the previous topic, the topic directly references death and life, but it also lacks what Rhody calls the "unambiguous comprehensibility" of the *night* topic. But examining the context of poems that contain the topic reveals a different pattern. These poems have a consistent *form* that Rhody describes as elegiac. She writes that "Paul Laurence Dunbar's 'We Wear the Mask' never once mentions the word 'death,' the discourse Dunbar draws from to describe the erasure of identity and the shackles of racial injustice are identified by the model as drawing heavily from language associated with death, loss, and internal turmoil — language which 'The Starry Night' indisputably also draws from."

5.5 Comparison to stylometric analysis

In addition to discussing what researchers have done in literary analysis with topic models, it is useful to consider how other technologies have been used in the same setting. One of the most established applications of computation in the study of literature is stylometry, or more specifically the question of authorship attribution Juola [2006]. It is illustrative to contrast the goals and methods of stylometry with those of topic modeling.

The critical insight of modern stylometry is that it is easy for authors to shift the focus of their work, but much more difficult to alter the semi-conscious style of their language Mosteller and Wallace [1964]. The implication is that content-bearing words, such as nouns and adjectives, are a relatively poor indicator of authorship or at least authorial style, while functional words, such as determiners, conjunctions, and prepositions, carry more information about authorship. Therefore, measures such as Burrows' delta Burrows [2002] restrict attention to the most frequent words in a corpus.

The contrast to topic modeling is clear: stylometric analysis focuses on frequent, low-information words and ignores content-bearing words, while topic modeling generally does the exact opposite. We generally remove high frequency words using a stop list, and in fiction go even further in removing words that are overly distinctive of a particular work. An assumption of topic modeling is therefore that the goal is to find thematic components that are *not* specific to one author, but rather exhibit themselves, with more or less variation, across multiple works. Where stylometry seeks to see past what authors are saying and focus on how they are saying it, a use of topic modeling is to find instances where different authors are writing about the same thing.

5.6 Operationalizing "theme"

The use of topic modeling in the study of literature has been beneficial both for humanities scholars and for machine learning researchers. For scholars, these models offer the possibility of a more precise approach to concepts that have traditionally been vague and impressionistic, such as theme, genre, and motif. At the same time, and somewhat paradoxically, literary documents present such a radically different mode of language than news articles or scientific publications that they lead us to question the apparent precision of statistical approaches.

Topic models provide a way of operationalizing the concept of distant reading. Moretti [2013b] defines this term as "Taking a concept, and transforming it into a series of operations." He attributes this definition to Bridgman [1927], who introduces the term in the context of measurement in physics: "To find the length of an object, we have to perform certain physical operations. The concept of length is therefore fixed when the operations by which length is measured are fixed: that is, the concept of length involves as much as and nothing more than the set of operations by which length is determined." While topic models are an imperfect tool for measuring theme in literature, they do provide a much more powerful approximation of theme than anything that we have had previously.

But applying statistical models to literature also brings forward a series of challenges that highlight the amount of human interpretive work that must go into successful topic modeling. Literary documents are of varied lengths, describe self-contained imaginary worlds, and are suffused with symbolic language. We can address these issues through corpus curation and through interpretive reading of models, but in doing so we must necessarily confront the fact that we are not simply applying Bridgman's fixed set of operations.

6

Understanding Scientific Publications

In Chapter 4, we discuss how scholars use topic models to understand non-fiction documents. This chapter focuses on a particular subgenre of non-fiction: scientific documents. Scientific documents deserve their own chapter because these documents are unique: they use very specialized vocabulary, they are the vehicles for innovation, and they shape important policy decisions. We discuss each of these aspects in turn.

Specialized Vocabularies Define Fields of Study First, scientific documents are unique because unlike general documents, their vocabulary is precise and carefully measured. “Resistance”, “splice”, “utilization”, and “demand” are common words with radically different meanings when used in specialized, technical contexts. Their use is a shibboleth whose use shows that you a member of a specific discipline (Figure 6.1). Thus, the ability of topic models to capture patterns of word usage also captures community and affiliation; this goes well beyond the thematic uses of topic models described in previous chapters.

Scientific Documents Innovate As any researcher will tell you, not every scientific publication is innovative; the sad truth is that most are

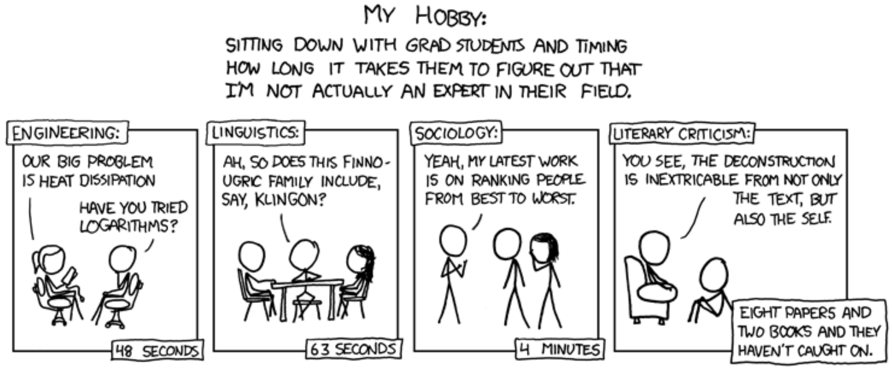

MY HOBBY:
SITTING DOWN WITH GRAD STUDENTS AND TIMING HOW LONG IT TAKES THEM TO FIGURE OUT THAT I'M NOT ACTUALLY AN EXPERT IN THEIR FIELD.
ENGINEERING:
OUR BIG PROBLEM IS HEAT DISSIPATION
HAVE YOU TRIED LOGARITHMS?
48 SECONDS
LINGUISTICS:
AH, SO DOES THIS FINNO-UGRIC FAMILY INCLUDE, SAY, KLINGON?
63 SECONDS
SOCIOLOGY:
YEAH, MY LATEST WORK IS ON RANKING PEOPLE FROM BEST TO WORST.
4 MINUTES
LITERARY CRITICISM:
YOU SEE, THE DECONSTRUCTION IS INEXTRICABLE FROM NOT ONLY THE TEXT, BUT ALSO THE SELF.
EIGHT PAPERS AND TWO BOOKS AND THEY HAVEN'T CAUGHT ON.

Figure 6.1: Using the appropriate language is a prerequisite for being part of a field (but not sufficient). Topic models use this to automatically discover fields of study.

not. However, some scientific monographs are Earth shattering (hopefully just figuratively). Unlike the other domains we've discussed, scientific documents are not just *reports* of news or events; they actually *are the news.*

What makes the analysis of scientific document collections both challenging and interesting is that innovation is hard to detect and hard to attribute. Einstein's groundbreaking 1905 papers were not fully recognized until many years later; important ideas are often proposed by an obscure researcher but only accepted once popularized and supported by other research; which document (researcher) in this case was the true source of the innovation? As we will see in this chapter, topic models can help answer this question.

Policy Makers Understanding Scientific Documents Understanding scientific publications is important for funding agencies, lawmakers, and the public. Government funding of science can create jobs, improve culture, and is an important form of international "soft power". However, knowing which research to fund is difficult, as the nature of science means that fields constantly change, which precludes rigid classifications [Szostak, 2004]. One challenge of modeling scientific documents is modeling how fields change over time; the static models we've discussed thus far are not always appropriate.

6.1 Understanding Fields of Studies

One of the first uses of topic models was to understand the "fields of science". Griffiths and Steyvers [2004] found that they were able to reconstruct the official PNAS topic codes automatically using topic models (Figure 6.2). This is a useful sanity check: yes, topic models correlate with what we often think of as scientific disciplines. They use distinct language for methods, subjects of study, and have different key players.

However, Griffiths and Styvvers were not in a position to do anything with their understanding of the fields of science. In contrast, Talley et al. [2011] sought to understand the American National Institutes of Health's funding priorities from within the organization.

The National Institutes of Health (NIH) are America's premiere funding agency for biological and health research. The NIH consists of several institutes that focus on particular diseases, research techniques, or body systems; each of these institutes manages its own independent funding portfolio, sometimes making it difficult to understand the "big figure" of funding.

Talley et al. [2011] used topic models to help create this big picture, in contrast to more labor-intensive techniques (e.g., keywords from a meticulously organized ontology). Their analysis discovered unexpected overlaps in research priorities across institutes. For example, many institutes study angiogenesis, the formation of new blood vessels; as a treatment for cancer, in heart imaging, the molecular basis of angiogenesis in the eye, and how angiogenesis might signal complications in diabetes.

6.2 How Fields Change over Time

One way that science is unique from the fields discussed in the previous chapters is that it is part of a continuous dialog. Each paper in its own way stands on the shoulders of giants. Topic models for science thus need to be aware of the connections between documents over time.

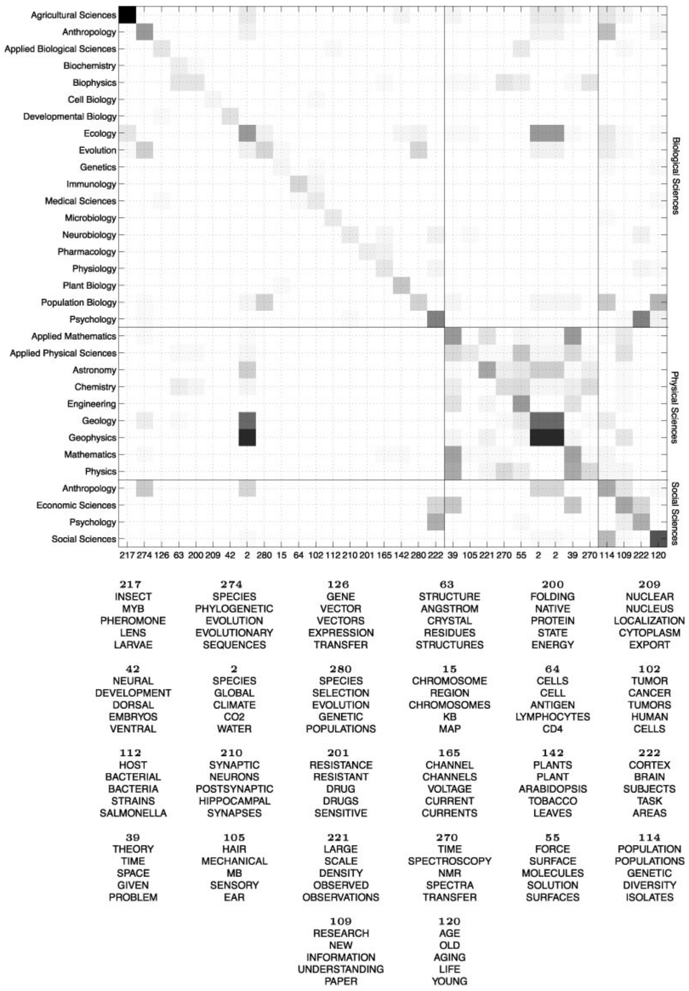


Figure 6.2: After running a topic model on PNAS, Griffiths and Steyvers [2004] found topics (x-axis) that could recreate the manually defined fields of study covered by PNAS (y-axis).

Another way that science is different is that the documents themselves introduce new ideas (we discuss detecting these innovative ideas in the next section).

One of the first techniques to do this viewed topics as subtly changing each year with a Dynamic Topic Model [Blei and Lafferty, 2006, DTM]. Within the generative model, this views each year's topic distribution to be distinct. That is, the physics topic has separate distributions over words for each year. Of course, we don't want the topics to be completely different every year—we want topics to change, but not *too much*.

The DTM views topics as changing through *Brownian motion*: the topic at year t is drawn from a Gaussian distribution with mean at the topic for year $t - 1$ (a separate variance parameter controls how much topics can vary each year). At this point, you may object given our discussion of distributions from Chapter 1.3.1: Gaussian produce continuous observations, while topics are multinomial distributions over discrete outcomes.

To move from Gaussian draws from $\vec{x} \in \mathbb{R}^d$ to a discrete distributions over d outcomes, Blei and Lafferty [2006] use the logistic normal form to create multinomial distribution

$$p(w = k \,|\, \vec{x}) = \frac{x_k}{\sum_i x_i}. \tag{6.1}$$

This greatly complicates inference, but allows the topics to change gradually from year to year.

With this model, the DTM discovers how fields change over time. At the start of the twentieth century, the language of physics focused on understanding how the "ether" propagates waves and the fundamental forces; by midcentury, understanding "quantum" effects took precedence; by the end of the century, experimental physics with large particle accelerators lead the search for ever more exotic members of the subatomic menagerie. While the final topic is nearly unrecognizable given the first, they all are clearly physics; the modeling assumptions of the DTM capture these nearly imperceptible changes in each year.

The flipping of a calendar page does not rule science, however; changes can happen at any time. Wang et al. [2008] captures changes in topics in continous time; each document gets its "own" view of a topic that can change slightly from the previous version of a topic. This can help capture sudden changes in scientific topics, e.g. from an innovative contribution.

6.3 Innovation

The changes to fields happen because of *innovation*. Scientists develop new techniques, new terminologies, and new understanding of the world. These concepts require new words which are reflected in their scientific publications. Unlike other fields, where documents merely report the changing world, scientific documents are themselves the force that can change the world: from Darwin's *Origin of Species* to Einstein's papers on relativity.

Thus, it is interesting to try to find out where this change is happening. From a historical perspective, it's interesting to learn who introduced groundbreaking research first. For policy makers [Largent and Lane, 2012], learning what research collaborations, conditions, and teams lead to breakthroughs can help direct new initiatives to recreate the magic that lead to these findings.

From a topic perspective, this amounts to detecting *who* was responsible for changing topics. From an institutional perspective, Ramage et al. [2010b] take a *post hoc* perspective: after fitting a standard LDA topic model, find the distribution over research topics in the entire research community at time t and th[illegible]ok back at time $t-1$ at the places who look like the future. They hypothesis is that these places "lead" other places to adopt their ideas.

Capturing more nuanced effects at either the individual or lab level requires refining the model. Gerrish and Blei [2010] adapt the random walk model of Wang et al. [2008] for scientific change. Instead of topic randomly bumbling into new concepts, Gerrish and Blei [2010] posit that innovative article "nudge" topics to look more like them in the future. This model is called the "Dynamic Influence Model" (DIM).

For example, the Penn Treebank [Marcus et al., 1993] revolutionized natural language process and helped enabled the statistical revolution in computational linguistics. Among its many effects is that people started using the word “treebank” much more than they had in the past. DIM captures this by explicitly modeling the influence $\delta_{k,d}$ of a document d in topic k. This document also has a distribution over words τ_d (for example, the article introducing the Penn Treebank uses “treebank” much more than “potato”). Recall that each topic is a distribution over words at time t, $\phi_{t,k}$.

Documents that don’t make a splash have zero influence, while influential documents are absorbed by other scientists who adopt the influential ideas and language,

$$\phi_{t+1,k} \propto \sum_d \delta_{k,d} \tau_d. \tag{6.2}$$

Each of these terms are random variables; inference in the model discovers the settings of the random variables that best explain the data. The DIM’s estimates of influence correlate well with the number of citations an article gets (the traditional measure of influence). Unlike citations, however, the DIM can be used in more informal settings to detect influential documents.

While science communication is ostensibly about promoting ideas and new understanding, topic models can also help understand less objective communications where influence isn’t just about facts but also is about emotion, beliefs, and relationships. The next chapter discusses how topic models can understand these messy, interesting properties of text.

7

Computational Social Science

While the previous chapters were mostly retrospective analyses, computational social science is mostly in the “here and now”. It is focus on data being generated in the past hours, days, or weeks to inform intelligence analysts, brand monitors, journalists, or social scientists. The underlying problem is the same, however: these stakeholders are interested in what people have to say but cannot read all of the data at their disposal.

Historically, social science asks questions such as what candidates is preferred in a particular part of the country or whether people like a new restaurant or product. These questions are often answered by polling: social scientists would head out into the world, gather a statistically significant survey sample, and extrapolate to the broader population.

These techniques are still valuable, but they still take time. A company needs to know if it has an issue with a product immediately, particularly if its good name is being dragged through the mud on social media Bowen [2016]. However, the reason for the acute time pressure can also be the solution: if a company is able to quickly see that it has a social media problem, it can more quickly intervene and correct the issue.

Traditional social science methods are labor intensive, take a long time, or are impossible for sensitive subjects. For instance, surveys of influenza take too long to be useful compared to the life cycle of influenza's progression Broniatowski et al. [2015]; using twitter and Google searches results in more accurate information faster.

Monitoring pollution in China or drug use in teens requires access to populations that may be difficult. Using social media presents an alternative Wang et al. [2015], as individuals share information more freely than official news agencies (which may suffer from official censorship in the case of Chinese pollution) or in school-administered surveys (which can suffer from self-censorship in the case of drug use). Topic models and other large-data approaches that can look at vast quantities of text help overcome some of the obstacles to fast-response social science.

7.1 Sentiment Analysis

In industrial settings, this problem is often called sentiment analysis [Pang and Lee, 2008]. Here, the goal is to determine the "sentiment"—e.g., positive or negative opinions—associated with a piece of text. For example, "Chipotle is great!" would be associated with positive sentiment, while "Chipotle made me sick" would be associated with negative sentiment.

While industrial applications of sentiment analysis are mostly for identifying whether people like a product or company, there are wider social science applications of examining large corpora to determine authors' *internal state*. For example, determining whether they they are politically liberal or conservative based on their online commentary.

Topic models can help these tasks by dividing a problem into topics. For example, "Apple" can appear in tech news as well as a food ingredient; someone monitoring the seller of iPods and iPhones would not want to be confused by social media commentary complaining about a bad apple pie. "Surprising" in an automotive review is likely associated with negative sentiment, while it's a good thing in a book review.

Thus, topics can help differentiate different kinds of discussion in broad corpora.

However, topic models lose their value if you want to *contrast* sentiment within a topic. While a topic model can find people discussing Chipotle burritos online, it cannot separate the lovers from haters. Thus, *distinguishing* topics based on their sentiment can help a user better understand how topics and sentiment interact in a dataset. This requires modifying the topic model to make it aware of the underlying sentiment.

7.2 Upstream and Downstream Models

To distinguish topics based on their sentiment, the model must be aware of what sentiment is. In the language of probabilistic models, sentiment and topic are modeled *jointly*. That is, there is a probability distribution over both the sentiment of a document y and the topics that is uses z.

There are two general kinds of joint models that incorporate metadata such as sentiment: upstream and downstream models. The distinction is based on the generative story of topic models (Chapter 1): is sentiment before (upstream) or after (downstream) topics in the generative story?

Upstream models assume that sentiment comes first in the generative story. That is, there will be different topics given the underlying sentiment. This can come in the form in a prior learned from observed sentiment [Mimno and McCallum, 2008] or from a latent variable that can serve as a proxy for sentiment [Lin and He, 2009]. Upstream models are often easier to implement and are more flexible [Roberts et al., 2014].

In contrast, downstream models explicitly predict sentiment *given* text. If the goal is the later predict sentiment given raw text with the help of topic models, downstream models can work better than upstream models. These models are often called "supervised" topic models after supervised LDA [Blei and McAuliffe, 2007], which use a document's topics to predict the downstream sentiment regression: a

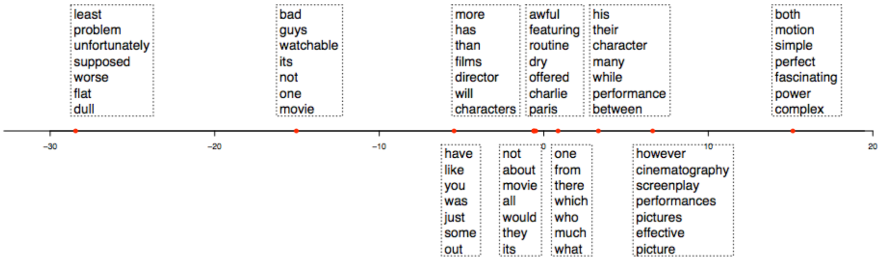


Figure 7.1: Example topics learned by supervised LDA from Blei and McAuliffe [2007]. Each topic is not just a collection of words but also has a regression score η that explains whether it is associated with positive sentiment (right) or negative sentiment (left).

document's sentiment y_d is assumed to come from a normal distribution with mean $\eta^\top \bar{z}$, where $\bar{z}$ is a normalized vector of all of the topics that a document uses and η is a regression parameter that describes the sentiment of each topic.

During inference, the words and sentiments work together to find combinations of topic and sentiment that make sense. While "vanilla" topic models seek to find clusters of words that make sense together, if a topic is associated with documents with many different sentiment values, it will have low probability.

Consider Figure 7.1. If a topic has an inconsistent sentiment values (for example, a negative sentiment document in a positive sentiment topic), inference will try to move the negative sentiment documents to topics with consistent sentiment η **and** consistent words.

These models form the foundation for the models and problems we discuss in the rest of this section.

Aside: Prediction or Interpretation A common theme in using topic models is the emphasis on whether models should prioritize *prediction* or *interpretation*. Our previous chapters have focused on interpretation: can a user understand the output of a model? But for supervised models, there's a question of how well the model can predict y_d (sentiment or another prediction of interest).

To some extent, these are not always in conflict. Ramage et al. [2010a] show that topic model features can improve tweet categorization, as do Blei and McAuliffe [2007] for supervised LDA. However, changing the objective function can further improve predictions [Zhu et al., 2009].

However, sometimes improved interpretability hampers the ability of the model to predict content. This is true of both words within a document and document labels. Chang et al. [2009] showed that complicated topic models do a better job of predicting held-out documents but make less sense to a user. Nguyen et al. [2015a] show that supervised models offer better predictions with additional topics but the topics are less interpretable.

7.3 Understanding Stance and Polarization

Another form of internal state is *stance*: which side does a person take on an issue. This can take many forms: are you for or against a proposal, are you a Democrat or a Republican, or are you a fan of the original Star Trek or the new version?

Upstream models can discover these sides by incorporating stance into the generative model. For example, Paul and Girju [2010] posit that each "side" had a distribution over words that it uses generally *and* that each side had its own take on how it discusses a topic. Within a document, each word is chosen either from a side's background distribution, a side's version of the topic, or from the topic's "neutral" words. For instance, Israelis and Palestinians both use "attacks", "civilians", and "military" in discussing unrest in Israeli-occupied Palestine, but the Israeli side uses "terrorist" and "incitement", while the Palestinian side focuses on "resistance" and "occupation".

Downstream models can also capture these divisions as well. Nguyen et al. [2013] predict whether a speaker is Republican or Democrat based on the versions of topics they discuss. For example, Republicans are more likely to discuss taxes in general than Democrats, but Democrats focus on the good that comes out of taxes (Figure 7.2).

However, there are not always two sides to an issue. A probabilistic solution to this model is the nested Dirichlet process [Blei et al., 2010]. These hierarchies induce a non parametric hierarchy over an unbounded number of topics. This corresponds to agenda setting from political science [Nguyen et al., 2015b].

7.4 Social Networks

We have talked about meta data that are independent for each user. Sometimes, however, we are interested in meta data that describe the relationships *between* documents: which users follow each other on Twitter, which scientific papers cite each other, or which webpages link to each other. This makes modeling more difficult, but we still see the same division between upstream and downstream models: upstream models assume that the communities form before we see words, while downstream models use the words to explain which links we see.

The stochastic block model [Holland et al., 1983] and its mixed-membership descendant [Airoldi et al., 2008] are prototypes for upstream models. They posits that there are intrinsic groups of documents and links are more likely inside the group than outside the group. These groups are analagous to the topics in topic models, except that the links are "shared" between documents.

However, the first probabilistic models of network structure ignored the words in documents. Link LDA is the exemplar for downstream models Nallapati and Cohen [2008] and include the text in the documents. It uses a regression on the topic allocations (θ) rather than topic assignments (z), in contrast to supervised LDA above.

Conditioning on the topic assignments can improve the algorithm's ability to predict links on held-out documents, however [Chang and Blei, 2009]. This is because a regression based on the allocations alone can use topics to explain links that aren't in the document. For example, if the model thinks there's a link between documents because they both use Topic 14 but no words in the document are assigned ($z_n = 14$), then the model is unable to recreate this prediction in a held-out document.

However, as social networks increasingly span the entire globe, assuming that a topic model is only in a single language is often a poor assumption. Indeed, even within a single country, topic models can discover regional variation [Eisenstein et al., 2010]. In the next chapter, we discuss how to cope with multilingual datasets and still discover reasonable topics.

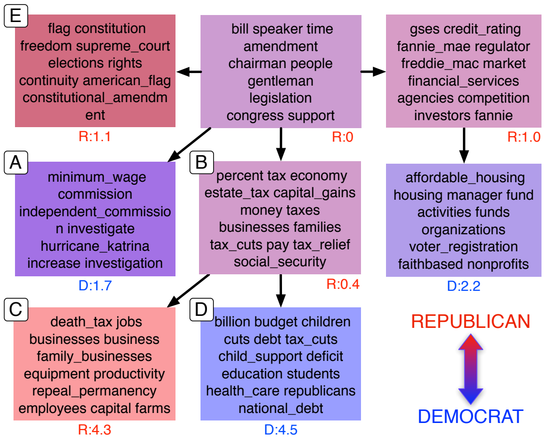

E
flag constitution freedom supreme_court elections rights continuity american_flag constitutional_amendm ent
R:1.1
bill speaker time amendment chairman people gentleman legislation congress support
R:0
gses credit_rating fannie_mae regulator freddie_mac market financial_services agencies competition investors fannie
R:1.0
A
minimum_wage commission independent_commissio n investigate hurricane_katrina increase investigation
D:1.7
B
percent tax economy estate_tax capital_gains money taxes businesses families tax_cuts pay tax_relief social_security
R:0.4
affordable_housing housing manager fund activities funds organizations voter_registration faithbased nonprofits
D:2.2
C
death_tax jobs businesses business family_businesses equipment productivity repeal_permanency employees capital farms
R:4.3
D
billion budget children cuts debt tax_cuts child_support deficit education students health_care republicans national_debt
D:4.5
REPUBLICAN
DEMOCRAT

Figure 7.2: Topics discovered from Congressional floor debates using a downstream model to capture speaker's ideology. Many first-level topics are bipartisan (purple), while lower level topics are associated with specific ideologies (Democrats blue, Republicans red). For example, the "tax" topic (B) is bipartisan, but its Democratic-leaning child (D) focuses on social goals supported by taxes ("children", "education", "health care"), while its Republican-leaning child (C) focuses on business implications ("death tax", "jobs", "businesses"). The number below each topic denotes the magnitude of a learned regression parameter associated with that topic. Colors and the numbers beneath each topic show the regression parameter η associated with the topic. From Nguyen et al. [2013].

8

Multilingual Data and Machine Translation

So far, we have been focusing on monolingual topic models and their applications. Multilingual topic models have also been developed to analyze and understand the corpus in multiple langauges, with statistical machine translation (SMT) is one of the major applications.

Given a text input in one language (source language), statistical machine translation tries to find a similar piece of text in another language (target language). Modern machine translation systems [Koehn, 2009] use millions of training examples to learn the translation rules and apply these rules on the test data. While the translation rules are learned in local context, these systems work best when the training corpus has consistent global context or the same genre (e.g., sports, business) from a similar style (e.g., newswire, blog-posts). These are called *domains*.

Translations within one domain are better than translations across domains since they vary dramatically in their word choices and style. A correct translation in one domain may be inappropriate in another domain. For example, “潜水” in sports domain usually means “underwater diving”, but in social media domain, it means a non-contributing “lurker”. To avoid such translation errors caused by the domain change, domain knowledge is needed to train translation systems that are robust to such systematic variation in the training set, which are said to exhibit *domain adaptation.*

To train such SMT systems with domain adaptation, early efforts focus on building separate models given the hand-labeled domains [Foster and Kuhn, 2007, Matsoukas et al., 2009, Chiang et al., 2011]. However, this setup is at best expensive and at worst infeasible for large data. Topic models provide a promising solution where domains can be automatically discovered. Each extracted topic is treated as a soft domain. [1] Thus the normal monolingual topic models on the source documents only have been applied to extract domain knowledge for machine translation [Eidelman et al., 2012].

However, the source language the and target language can complement each other to build up more accurate topic models. For example, if we only know the Chinese phrase “潜水”, it is hard to decide whether it is a sport domain or it is a social media domain. However, with the help of the aligned English translation “luker”, it is easy to identify the “social media” domain. Thus multilingual topic models [Mimno et al., 2009, Boyd-Graber and Resnik, 2010] have been applied to extract domain knowledge for machine translation [Hu et al., 2014b].

This chapter first reviews the basic components of statistical machine translation, then introduces how to apply the monolingual and multilingual topic models in domain adaptation to improve each component of the statistical machine translation systems.

8.1 Statistical Machine Translation

Statistical machine translation casts machine translation as a probabilistic process [Koehn, 2009]. Here we briefly introduce the standard phrase translation model introduced in [Koehn et al., 2003]. Given the source sentence $\mathbf{f}$, the best translation in target language $\mathbf{e}_{\texttt{best}}$ is modeled as,

$$\mathbf{e}_{\texttt{best}} = \mathbf{argmax}_{\mathbf{e}} p(\mathbf{e}|\mathbf{f}) = \mathbf{argmax}_{\mathbf{e}} p(\mathbf{f}|\mathbf{e}) p(\mathbf{e}) \tag{8.1}$$

[1] Henceforth we will use the term “topic” and “domain” interchangeably: “topic” to refer to the concept in topic models and “domain” to refer to SMT corpora.

which is split to a *translation model* $p(\mathbf{f}|\mathbf{e})$ and a *language model* $p(\mathbf{e})$.

The source sentence $\mathbf{f}$ is segmented into multiple source phrases $\bar{f}_n$ during the decoding, which are translated to a set of target phrases $\bar{e}_n$. Thus the translation probability $p(\mathbf{f}|\mathbf{e})$ can be further decomposed to the phrase translation probability $p(\bar{f}_n|\bar{e}_n)$. Besides, the target phrases may need to be *reordered* to get the best translation result, and this part is captured by a relative distortion probability distribution $d(a_n - b_{n-1})$, where a_i denotes the start position of the source phrase that was translated to the nth target phrase, and b_{n-1} denotes the end position of the source phrase translated into the $(n-1)$th target phrase. As a result, the translation model can be decomposed as,

$$p(\mathbf{f}|\mathbf{e}) = \prod_n p(\bar{f}_n|\bar{e}_n) d(a_n - b_{n-1}) \tag{8.2}$$

In phrase-based SMT, the phrase probability $p(\bar{f}_n|\bar{e}_n)$ can be further estimated by combining lexical translation probabilities of words contained in that phrase [Koehn et al., 2003], which is normally referred as *lexical weighting*. Lexical conditional probabilities $p_w(f|e)$ are maximum likelihood estimates from relative lexical frequencies,

$$p_w(f|e) = c(f,e) \Big/ \textstyle\sum_f c(f,e) \tag{8.3}$$

where $c(f,e)$ is the count of observing lexical pair (f,e) in the training dataset. Given a word alignment a, the lexical weight for this phrase pair $p_w(\bar{f}|\bar{e};a)$ is the normalized product of lexical probabilities of the aligned word pairs within that phrase pair:

$$p_w(\bar{f}|\bar{e};a) = \prod_i \frac{1}{\{|j|(i,j) \in a\}|} \sum_{\forall (i,j) \in a} p_w(f_i|e_j) \tag{8.4}$$

where i and j are the word positions in target phrase $\bar{e}$ and source phrase $\bar{f}$ respectively.

Next we introduce how to apply topic models to improve translation models, language models and reordering models respectively.

8.2 Topic Models in Translation Models

To Train such SMT systems with domain adaptation, early efforts focus on building separate models based on the hand-labeled domains [Foster and Kuhn, 2007, Matsoukas et al., 2009, Chiang et al., 2011]. For example, for all the training examples that are labeled as sports domain, one translation model is trained. As a result, in any test example that is labeled as sports, “潜水” is always translated to “underwater diving”, and the probability of translating “潜水” to “lurker” is zero. In fact, such hard domain labels are not only expensive and time consuming to obtain, but also unsmoothed and sensitive to labeling erros.

Unlike the manual hard domain labels, topic models provide a promising solution for automatically discovering the soft domain assignments. While the topic-word distributions generated by topic models show what each topic (domain) is about, the document-topic distributions provide such soft domain assignments for each document. Let’s continue to use the previous example and assume there are two topics sports and social media. Given a test example that is most likely about sports, it may have a soft domain distribution as 99% for sports domain and 1% for social media. These automatically obtained soft domain labels are well smoothed, and they are not only cheap to obtain but also much more robust to topic errors. Next, we introduce the details about applying monolingual and multilingual topic models to improve translation models [Eidelman et al., 2012, Hu et al., 2014b].

8.2.1 Translation Domain Adaptation with Topic Models

Topic models take the number of topics K and a collection of documents as input, where each document is a bag of words. They output two distributions: a distribution over topics for each document d; and a distribution over words for each topic. If each topic defines a SMT domain, the document’s topic distribution is a soft domain assignment for that document.

Topical Lexical Features Eidelman et al. [2012] use such soft domain assignments to build up translation models. The document-topic distribution $p(k|d)$ is used to smooth the expected count $\hat{c}_k(f, e)$ of a word translation pair under topic k,

$$\hat{c}_k(f, e) = \textstyle\sum_d p(k|d) c_d(f, e), \tag{8.5}$$

where $c_d(\bullet)$ is the number of occurrences of the word pair in document d. The lexical probability conditioned on topic k is the unsmoothed probability estimate of those expected counts

$$p_w(f|e;k) = \hat{c}_k(f,e) / \textstyle\sum_f \hat{c}_k(f,e), \tag{8.6}$$

from which we can compute the lexical weight of this phrase pair $p_w(\bar{f}|\bar{e};a,k)$ given a word alignment a[Koehn et al., 2003]:

$$p_w(\bar{f}|\bar{e};a,k) = \prod_{i=1}^{n} \frac{1}{\{|j|(i,j) \in a\}|} \sum_{\forall(i,j)\in a} p_w(f_i|e_j;k) \tag{8.7}$$

where i and j are the word positions in target phrase $\bar{e}$ and source phrase $\bar{f}$ respectively. Comparing Equation 8.6 against Equation 8.3, and Equation 8.7 against Equation 8.4, we can clearly see how the topics are encoded as soft domains and how these topics influence the probability.

For a test document d, the document topic distribution $p(k|d)$ is inferred based on the topics learned from training data. The lexical weight feature of a phrase pair $(\bar{f},\bar{e})$ is,

$$f_k(\bar{f}|\bar{e}) = -\log\left\{p_w(\bar{f}|\bar{e};k) \cdot p(k|d)\right\}, \tag{8.8}$$

a combination of the topic dependent lexical weight and the topic distribution of the document, from which we extract the phrase.

In fact, Eidelman et al. [2012] introduces the two direction topic-adapted probabilities instead of a single direction: $p_w(\bar{f}|\bar{e};a,k)$ and $p_w(\bar{e}|\bar{f};a,k)$. This is equivalent to introduce $2K$ new word translation tables, thus it results in introducing $2K$ lexical weight features $f_k(\bar{f}|\bar{e})$ and $f_k(\bar{e}|\bar{f})$. These topic-adapted features are combined with the other standard SMT features including the standard $f(\bar{f}|\bar{e})$ and $f(\bar{e}|\bar{f})$, and the feature weights are optimized through using the *Margin Infused Relaxed Algorithm* [Crammer et al., 2006, MIRA].

These adapted features allow us to bias the translations according to the topics. For example, if topic k is dominant in a test document, the feature $f_k(\bar{f}|\bar{e})$ will be large, which may bias the decoder to a translation that has small value of the standard feature $f(\bar{f}|\bar{e})$. In addition, combining the adapted features with the standard features makes this model more flexible. For a test document with less clear topics, the topic distribution will tend toward being fairly uniform. In this case, the topic features will contribute less to the translation results and the standard features will dominate the translation results.

Topical Lexical and Phrasal Features Hasler et al. [2012] also apply topic models for domain adaptation to SMT in a similar framework as Eidelman et al. [2012], except they introduce different features, which they call sparse word pair features and phrase pair features. The topics on source documents are integrated as a source side trigger for a particular word pair or phrase pair as sparse features. For example, given a word w_f with topic k, for an aligned word pair (w_f, w_e) which is observed c times in the aligned sentence pair, the sparse word pair feature wp with topics is represented as $wp\_k\_w_f \sim w_e = c$, while the original word pair feature is $wp\_w_f \sim w_e = c$. The topic phrase pair features are also defined in a similar way: given an aligned phrase pair (p_f, p_e) with count c in the same sentence, the sparse phrase pair feature pp with topics is represented as $pp\_k\_p_f \sim p_e = c$. Both the phrase pair and word pair features are extracted from the aligned training sentence pairs, and then MIRA is used to learn the feature weights.

One difference in Hasler et al. [2012] from Eidelman et al. [2012] is that they are using *hidden topic Markov models* [Gruber et al., 2007, HTMM] instead of LDA to learn topics. While LDA assumes that each word is generated independently in a document, HTMM models the word topic in a document as a Markov chain where all words in a sentence are assigned with the same topic. HTMM computes $p(z_n, \Phi_n | d, w_1, \cdots, w_N)$ for each sentence, where z_n is the topic of sentence n in document d, $w_1, \cdots, w_N$ are the words in the sentence n, Φ_n is the topic transition between words. Φ_n is only non-zero at sentence boundaries. The advantage of using HTMM is that each sentence gets the same topic assignment, thus the topic for each phrase pair in the aligned sentence is consistent and can be used for topical features directly.

Topical Phrase Probability via Topic Mapping Su et al. [2012] also apply HTMM to SMT based on a similar idea as Eidelman et al. [2012].

Given the bilingual translation training data without any specific domain information (referred as out-of-domain bilingual data), they incorporate the topic information from the source language into translation probability estimation, and decompose the phrase probability $p(\bar{e}|\bar{f})$ as,

$$p(\bar{e}|\bar{f}) = \sum_{k_{out}} p(\bar{e}, k_{out}|\bar{f}) = \sum_{k_{out}} p(\bar{e}|\bar{f}, k_{out}) \cdot p(k_{out}|\bar{f}) \tag{8.9}$$

where $p(\bar{e}|\bar{f}, k_{out})$ is the translation probability given the source side topic k_{out}, and $p(k_{out}|\bar{f})$ denotes the phrase probability in topic k_{out}.

In addition, Su et al. [2012] assume that a monolingual corpus in the same domain as the test sentence(referred as in-domain monolingual data) is available. Thus they also apply HTMM to estimate the in-domain topic k_{in} and $p(k_{in}|\bar{f})$. However, the in-domain topics k_{in} and the out-of-domain topics k_{out} may not be in the same space, so Su et al. [2012] introduce the topic mapping probability $p(k_{out}|k_{in})$ to map the in-domain topic to the out-of-domain topic:

$$p(k_{out}|\bar{f}) = \sum_{k_{in}} p(k_{out}|k_{in}) \cdot \cdot p(k_{in}|\bar{f}) \tag{8.10}$$

As a result, the final phrase probability can be refined as,

$$p(\bar{e}|\bar{f}) = \sum_{k_{out}} \sum_{k_{in}} p(\bar{e}|\bar{f}, k_{out}) \cdot p(k_{out}|k_{in}) \cdot p(k_{in}|\bar{f}) \tag{8.11}$$

Comparing with the approaches in Eidelman et al. [2012] and Hasler et al. [2012], this approach incorprates the topic information into the phrase probability directly, rather than through the word translation probability. The topics and topic mapping relationship between the training data and test data can be built offline, thus the whole process brings no additional burden to the translation system.

Problems However, Eidelman et al. [2012], Hasler et al. [2012], Su et al. [2012] ignore a wealth of information that could improve topic models and help machine translation. They rely solely on monolingual source-side models. In contrast, machine translation uses inherently multilingual data: an SMT system must translate a phrase or sentence from a *source* language to a different *target* language, so these approaches are ignoring available information on the target side that could aid domain discovery. This can be improved by introduction multilingual topic models, and the details are introduced in the next section.

8.2.2 Multilingual Information for Domain Adaptation

While the previous approaches focus on applying topic models on the monolingual source language of tranlsation data, the bilingual data indeed can complement each other to reduce topic ambiguity. For example, "木马" in a Chinese document can be either "hobbyhorse" in a children's topic, or "Trojan virus" in a technology topic. A short Chinese context obscures the true topic. However, these terms are unambiguous in English, revealing the true topic. From this example, we can see that that topic models on multilingual corpus do improve the quality of the extracted topics.

There are various ways to build up the multilingual topic models. Different languages can be connected on the word-level [Boyd-Graber et al., 2007, Andrzejewski et al., 2009, Hu et al., 2014a] or the document levels [Mimno et al., 2009]. Hu et al. [2014b] further combine the two types of connections to build up multilingual topic models. Given the improved multilingual topic models, the extracted topics are further applied in the SMT framework from Eidelman et al. [2012] to improve the machine translation results.

Word-level Correlations *Lexical information*, such as orthographic similarity [Boyd-Graber and Blei, 2009] and multilingual dictionaries [Boyd-Graber and Resnik, 2010], can be very helpful to induce better topics from multilingual corpora. For instance, tree-based topic models [Boyd-Graber et al., 2007, Andrzejewski et al., 2009, Hu et al., 2014a, tLDA] incorporate the positive correlations between words in the same or different languages by encouraging words that appear together in a **concept** to have similar probabilities given a topic. These concepts can come from WordNet [Boyd-Graber and Resnik, 2010], domain experts [Andrzejewski et al., 2009], or user constrains [Hu et al., 2014a]. If these concepts are in the same language, it is the same backend model for interactive topic modeling introduced in Chapter 3. However, when we gather concepts from bilingual resources, these concepts can connect different languages. For example, if a bilingual dictionary defines “电脑” as “computer”, we combine these words in a concept.

These concepts (positive correlations) are organized into a **prior tree** structure. As Figure 8.1 shows, words in the same concept share a common parent node, and then that concept becomes one of many children of the root node. Words that are not in any concept—**uncorrelated words**—are directly connected to the root node. Thus a topic becomes a distribution over all paths in this prior tree and each path is associated with a word.

The probability of a path in a topic depends on the transition probabilities in a topic. Each concept i in topic k has a distribution over its children nodes is governed by a Dirichlet prior: $\pi_{k,i} \sim \mathrm{Dir}(\beta_i)$. Each path ends in a word (i.e., a leaf node) and the probability of a path is the product of all of the transitions between topics it traverses. Topics have correlations over words because the Dirichlet parameters can encode positive or negative correlations [Andrzejewski et al., 2009].

As a result, to sample a word w_{dn} given a topic z_{dn}, a path y_{dn} from the topic tree of topic z_{dn} is sampled: we start from the root n_0 and first sample a child node n_1 of the root; if node n_1 is a concept node, we continue to sample a word node n_2 and generate the word assciated with n_2; if node n_1 is a word node already, we generate the word directly.

When this tree serves as a prior for topic models, words in the same concept are positively correlated in topics. For example, if “电脑” has high probability in a topic, so will “computer”, since they share the same parent node. With the tree priors, each topic is no longer a distribution over word types; instead, it is a distribution over paths, and each path is associated with a word type. The same word could appear in multiple paths, and each path represents a unique sense of this word.

Document-level Alignments Lexical resources connect languages and help to guide the topics. However, these resources are sometimes

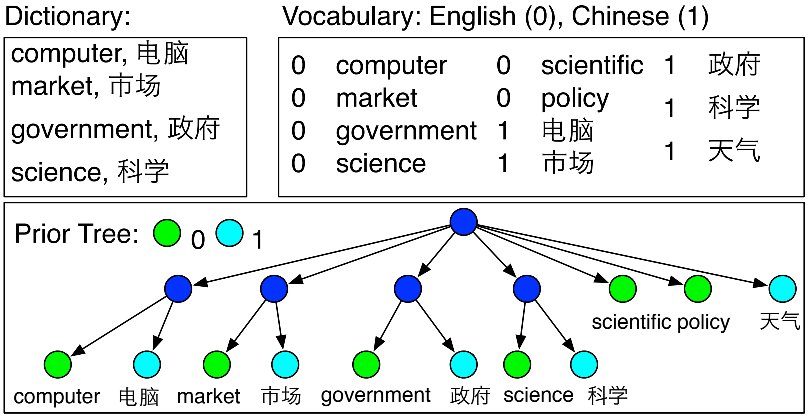


Figure 8.1: An example of constructing a prior tree from a bilingual dictionary: word pairs with the same meaning but in different languages are concepts; a common parent node is created to group words in a concept, and then is connected to the root; uncorrelated words are connected to the root directly.

brittle and may not cover the whole vocabulary. Aligned document pairs provide a more corpus-specific, flexible association across languages.

Landauer and Littman [1990] connect aligned documents in different languages by projecting both documents to a shared latent semantic indexing space. Similarly, polylingual topic models [Mimno et al., 2009, pLDA] assume that the aligned documents in different languages share the same topic distribution and each language has a unique topic distribution over its word types.

Thus the generative process of polylingual topic model is as follows: given a document pair (d_{l_1}, d_{l_2}), we first sample a document-topic distribution θ_d; for a document d_{l_i} in language l_i, we then sample a topic z_{dn} from θ_d, and generate a word from topic ϕ_{z_{dn},l_i} in language l_i.

This connection between languages is flexible: instead of requiring the exact matching or translations on words and sentences, only a coarse document alignment is necessary, as long as the documents discuss the same topics, e.g., wikipedia articles in different languages. Such connection between languages is also helpful to infer more robust topics, since different languages can complement each other to reduce ambiguity.

Combine Words and Documents These two approaches are not mutually exclusive, however; they reveal different connections across languages. Hu et al. [2014b] bring existing tree-based topic models (tLDA) and polylingual topic models (pLDA) together and create the polylingual tree-based topic model (ptLDA) that incorporates both word-level correlations and document-level alignment information.

To build up the prior tree structure, Hu et al. [2014b] consider two resources that correlate words across languages. The first is multilingual dictionaries, which match words with the same meaning in different languages together. The other is the word alignments extracted from aligned sentences in a parallel corpus. These relations between words are used as the concepts [Bhattacharya, 2006] in the prior tree (Figure 8.1).

Given the prior tree structure, the generation of documents is a combination of tLDA and pLDA. For each aligned document pair (d_{l_1}, d_{l_2}), we first sample a distribution over topics θ_d from a Dirichlet prior $\text{Dir}(\alpha)$. For each token in the aligned document d_{l_i}, we first sample a topic z_{dn} from the multinomial distribution θ_d, and then sample a path y_{dn} along the tree of topic z_{dn}. Because every path y_{dn} leads to a word w_{dn} in language l_{dn}, we append the sampled word w_{dn} to document $d_{l_{dn}}$ in language l_{dn}.

If a flat symmetric Dirichlet prior is used instead of the tree prior, pLDA is recovered; and if all documents are monolingual (i.e., with distinct distributions over topics θ), tLDA is recovered. ptLDA connects different languages on both the word level (using the word correlations) and the document level (using the document alignments), thus it learns better topics by considering more information from both languages.

Topic Projection While the above approaches try to model the source and target languages simultaneously to extract topics, Xiao et al. [2012] apply topic models on the source documents and target documents respectively to learn the document-topic distributions $p(k_f|d_f)$ and $p(k_e|d_e)$, and then estimate the phrase-topic probability $p(\bar{e}, k_f|\bar{f})$ and $p(\bar{e}, k_e|\bar{f})$ respectively. They further compute the topic simiality scores between the phrase topic distribution and document topic distribution as features for decoding to improve SMT results.

For test data, they estimate the document-topic distribution $p(k_f|d_f)$ for source document only, and compute the topic similarity between the document-topic distribution $p(z_f|d_f)$ and the phrase-topic distribution $p(\bar{e}, z_f|\bar{f})$ as one similarity feature for decoding,

$$\texttt{Similarity}(p(z_f|d_f), p(\bar{e}, z_f|\bar{f})) = \sum_{k=1}^{K}(\sqrt{p(z = k_f|d_f)} - \sqrt{p(z = k_f, \bar{e}|\bar{f})})^2 \tag{8.12}$$

Similarly, they also consider a second similarity feature $\texttt{Similarity}(p(z_f|d_f), p(\bar{e}, z_e|\bar{f}))$. However, the source topics z_f may not be in the same space as the target topics z_e. Thus they learn the topic projection probability $p(k_f|k_e)$ by normalizing the co-occurrence count in the aligned training sentences, then the target phrase-topic probability $p(\bar{e}, z_e|\bar{f})$ is projected to the source topic space as,

$$P(p(\bar{e}, z_e|\bar{f})) = p(\bar{e}, z_e|\bar{f}) \cdot p(k_f|k_e) \tag{8.13}$$

Thus the second feature $\texttt{Similarity}(p(z_f|d_f), P((\bar{e}, z_e|\bar{f})))$ is computed.

This topic projection idea is similar to the topic mapping by Su et al. [2012], but it is applied between the source language and the target language. Compared to the lexical features in Eidelman et al. [2012] and Hu et al. [2014b], Xiao et al. [2012] introduce a new framework to apply topic information on phrase directly and present two topic similarity features for decoding. These two approaches can be combined to further improve SMT.

8.3 Topic Models in Language Modeling

A critical component of machine translation systems is the language models, which provide local constraints and preferences to make translations more coherent. A language model describes the probability of a word w occurring given the previous context words, which is also mentioned as the history h. We have introduced the application of language models in information retrieval in Chapter 2.1. In fact, it also helps to choose the correct or more proper word during the statistical machine translation. For example, the English translation “I am going home” is preferred than “I am going house”.

Domain adaptation for Language models [Bellegarda, 2004, Wood and Teh, 2009] uses extra knowledge to adjust this probability $p(w|h)$ to reflect the content change, which is an important avenue for improving machine translation. As Bellegarda [2004] points out that “an adaptive language model seeks to maintain an adequate representation of the current task domain under changing conditions involving potential variations in vocabulary, syntax, content, and style”.

Topics from topic models can be one of the resources to provide such knowledge for language model adaptation. For example, the Chinese phrase “很多粉丝” is translated to “a lot of vermicelli” in a food domain, but means “a lof of fans” in an entertainment domain. Such ambiguity can be reduced by considering the topics (domain knowledge)from topics. If the entertainment topic is extracted based on the previous context, this Chinese phrase will be translated to “a lof of fans” without any ambiguity. Next, we introduce the details about applying topic models for language model adaptation.

8.3.1 Monolingual Topic Models for Language Model Adaptation

Early work Clarkson and Robinson [1997], Seymore and Rosenfeld [1997], Kneser and Peters [1997], Iyer and Ostendorf [1999] focus on partitioning the training data to multiple topic-specific subsets and building up language models for each subset. Then the topic-specific language models $p_k(w|h)$ are linearly combined with a general language model $p_g(w|h)$ built from all training data as Equation 8.14. The weights λ_k can be tuned based on the topics of the test documents.

$$p_{\text{adapted}}(w|h) = \sum_k \lambda_k p_k(w|h) + \lambda_g p_g(w|h) \tag{8.14}$$

Seymore et al. [1998] further identify the most qualified topic for each word in the vocabulary and choose a topic-specific language model or the general language model. The intuition is that the general language model provides the most reliable estimation for general words, and the topic language model estimates the probability more accurately for topic words. As a result, they split the vocabulary words into three groups: the general subset, on-topic subset and off-topic subsets. For general subset and off-topic subset, the general language model provides the word probability; and the topic-specific language model provides the word probability for the on-topic subset.

While these early attempts introduce the topics to improve the language models, these topic-specific language models are just like the traditional n-gram models, which can model the limited history. In addition, these models assume each document or history belongs to exactly one topic cluster.

To fix these problems, models with topic mixtures, such as *Latent Semantic Analysis* [Deerwester et al., 1990, LSA] and its probabilistic interpretation—probabilistic latent semantic indexing [Hofmann, 1999a, PLSI], have been introduced to learn large-span language models [Bellegarda, 1997, Coccaro and Jurafsky, 1998, Gildea and Hofmann, 1999]. Gildea and Hofmann [1999] decomposes the language model based on topics,

$$p(w|h) = \sum_k p(w|k)p(k|h) \tag{8.15}$$

where the topics are learned from the training corpus by optimizing the log probability,

$$l(\theta; N) = \sum_w \sum_d n(w,d) \log \sum_k p(w|k)p(k|d) \tag{8.16}$$

where d is the training documents, and $n(w,d)$ is the word frequency of w in document d. $p(w|k)$ and $p(k|d)$ are learned through the EM algorithm. For test documents, they fix $p(w|k)$ to estimate $p(k|h)$ and then compute $p(w|h)$ using Equation 8.15.

These approaches of applying topic models to language models are very similar to the document language modeling for information retrieval, which have been introduced in Chapter 2.1. However, unlike information retrieval, two different languages are involved in the process of SMT, and they can complement each other to learn more accurate topics. Next, we introduce the multilingual topic models for language model adaptation.

8.3.2 Multilingual Topic Models for Language Model Adaptation

As we explained in Section 8.2.2, the information from different languages can complement each other to extract better topics. In order to introduce multilingual information to topic models for language model adaptation, various approaches such as Tam et al. [2007], Ruiz and Federico [2011], Yu et al. [2013] etc., have been explored. More details are introduced in this section.

Bilingual Latent Semantic Analysis Tam et al. [2007] introduces the bilingual latent semantic analysis (BLSA) to learn the topics for both source language and target language, and apply the learned topics into language model adaptation for SMT. BLSA transfers the inferred topics from the source language to the parallel target language. In fact, the idea is similar to polylingual topic models [Mimno et al., 2009].

More specifically, Tam et al. [2007] assume the aligned source document and the target document share the same document-topic distribution, thus they first learn a LSA model on the source language. Then they use this document-topic distribution from the source document as the document-topic distribution for the aligned target document, and then infer the topic-word distribution on the target side. The topics on the target language are not learned iteratively, thus the topics in parallel corpus can be learned very efficiently.

In order to apply the topics for language adaptation, the word marginal distribution $p_{lsa}(w)$ for document d is computed,

$$p_{lsa}(w) = \sum_{k=1}^{K} p(w|k)p(k|d) \tag{8.17}$$

Then this word marginal distribution is integrated into the target background language model by minimizing the KL divergence between the adapted language model and the background language model [Kneser et al., 1997]:

$$p_a(w|h) \propto \Big(\frac{p_{lsa}(w)}{p_{bg}(w)}\Big)^{\beta} \cdot p_{bg}(w|h) \tag{8.18}$$

Ruiz and Federico [2011] also apply the similar idea for language model adaptation. In stead of using BLSA, Ruiz and Federico [2011] simply merge the aligned source and target document as one document, and perform the probabilistic latent semantic analysis. Both ideas are based on the assumption that the aligned source document and target document share the same document-topic distribution. The final adapted language model combines the topic-based language model with the general background language model, thus it is more robust in improving the results of SMT.

Topic Projection Yu et al. [2013] introduces the hidden topic markov model to improve the language model in SMT. They build up a topic model on the source side and target side respectively, and learn a topic-specific language model based on the target side by estimating the maximum-likelihood. To smooth the sharply distributed probabilities, back-off probabilities are also considered as follows:

$$\begin{aligned} p(w_i|w_{i-n+1}^{i-1}, k_e) = & \lambda_{w_{i-n+1}^{i-1}} p_{MLE}(w_i|w_{i-n+1}^{i-1}, k_e) \qquad (8.19) \\ & + (1 - \lambda_{w_{i-n+1}^{i-1}}) p(w_i|w_{i-n+2}^{i-1}, k_e) \qquad (8.20) \end{aligned}$$

where λ is the normalization parameter, calculated by,

$$\lambda_{w_{i-n+1}^{i-1}, k_e} = \frac{N_{1+}(w_{i-n-1}^{i-1}, k_e)}{N_{1+}(w_{i-n-1}^{i-1}, k_e) + \sum_{w_i} c(w_{i-n+1}^{i}, k_e)} \tag{8.21}$$

where $N_{1+}(w_{i-n-1}^{i-1}, k_e)$ is the number of words following w_{i-n-1}^{i-1} in topic k_e, and $c(w_{i-n+1}^{i}, k_e)$ is the count of n-gram w_{i-n+1}^{i} in k_e.

During the decoding, since no target sentence is available, they use the topic model on the source side to predict the topics for the testing data, and then project the source topic to the target topic, and then estimate the probability as following:

$$p(e) = \sum_{k_e} p(e|k_e) p(k_e) = \sum_{k_e} p(e|k_e) \cdot \sum_{k_f} p(k_e|k_f) p(k_f) \tag{8.22}$$

where $p(k_e|k_f)$ is the topic projection probability, estimated by the co-occurrence of the source-side and the target-side topic assignment.

Topic	Type	Example
Economy	Source	··· **比五** 月份下降3.8% ···
	Target	··· down 3.8% from May ···
Sports	Source	··· **五比**一3.8% ···
	Target	··· five to one ···

Table 8.1: Topics influence the word orders: the Chinese words in bold are in different orders in different topics. (Example from Wang et al. [2014])

8.4 Reordering with Topic Models

In addition to translation models and languages models, a third important component of a phrase-based SMT systemt is the reordering models, which learn how the order of words in the source sentences influences the order of words in the target sentences and how to make the translations in the right order.

While a topic model helps the domain adaptation of translation models and language models, it is not obvious how it helps the reordering model, but it does. The word orders in different domains of the same language may be different, and this is where the topic models come to help. As the example shown in Table 8.1 [Wang et al., 2014], in economy topic, Chinese word 比 is on the left of 五; but in sports topic, 比 is on the right of 五. As a result, it is necessary to introduce domain knowledge (topics) to model such order variance.

This section introduces two different reordering models and how to apply topic models to improve the reordering model respectively.

8.4.1 Lexicalized Reordering Models with Topics

A lexicalized reordering model is to learn the orientation probabilities for a given phrase pair with respect to the previous phrase pair and the following phrase pair [Chen et al., 2013]. The orientation o typically includes three types: monotone (M), swap (S) and discontinuous (D). The M orientation means the current phrase pair is immediately to the right of the previously translated phrase in the source sentence, S orientation occurs when the current phrase is immediately to the left of the previous phrase, and all other cases all belong to D orientation.

Formally, the reordering model is defined to estimate the corresponding probabilities $p(o|f,e)$ using a recursive MAP smoothing:

$$p(o|f,e) = \frac{c(o,f,e) + \alpha_f p(o|f) + \alpha_e p(o|e)}{c(f,e) + \alpha_f + \alpha_e} \tag{8.23}$$

$$p(o|f) = \frac{c(o,f) + \alpha_g p(o)}{c(f) + \alpha_g} \tag{8.24}$$

$$p(o|e) = \frac{c(o,e) + \alpha_g p(e)}{c(e) + \alpha_g} \tag{8.25}$$

$$p(o) = \frac{c(o) + \alpha_u/3}{c(\cdot) + \alpha_u} \tag{8.26}$$

where $c(o,f,e)$ is the orientation counts obtained from the word-aligned corpus; the parameters α_f, α_e, α_g and α_u are learned by minimizing the perplexity of the resulting model on the held-out data.

This type of reordering model is normally referred as lexicalized since the estimated orientations depend on the words in both the previously translated phrase pair and the current phrase pair.

Linear Adaptation with Topics Chen et al. [2013] find that training corpus in different domains vary significantly in their reordering characteristics for particular phrase pairs, thus they introduce the linear model for reordering model adaptation. Given N sub-corpora (or N domain corpus), they first train a reordering model on each domain corpus, and then define the global reordering model as a linear combination of the reordering models on each domain corpus:

$$p(o|f,e) = \sum_{i=1}^{N} \alpha_i p_i(o|f,e) \tag{8.27}$$

where $p_i(o|f,e)$ is the reordering model on sub-corpus i, and the weight α_i are learned by maximizing the probability of phrase-pair orientations in the in-domain development set:

$$\hat{\alpha} = \texttt{argmax}_{\alpha} \sum_{o,f,e} \tilde{p}(o,f,e) \log \sum_{i=1}^{N} \alpha_i p_i(o|f,e) \tag{8.28}$$

where $\tilde{p}(o, f, e)$, proportional to $c(o, f, e)$, is the empirical distribution of counts in the development set. Chen et al. [2013] propose to smooth the in-domain sample and weight instances by document frequency to further improve the mixture adaptation. They demonstrate this adaptation significantly improve the statistical machine translation system.

8.4.2 Maximum Entropy Classification with Topics

However, Chen et al. [2013] manually divide the training data into multiple domains, instead of using automatic techniques such as topic models. Wang et al. [2014] introduce topic models to the maximum entropy based phrase reordering models, proposed by Xiong et al. [2006].

Under the ITG constraints [Wu, 1997], three Bracketing Transduction Grammar (BTG) rules are used to constrain the translation and reordering:

$$A \rightarrow [A^1, A^2] \tag{8.29}$$

$$A \rightarrow < A^1, A^2 > \tag{8.30}$$

$$A \rightarrow f/e \tag{8.31}$$

where A is a block with a pair of source and target strings; A^1 and A^2 are two consecutive blocks; the two rules are reordering rules which merge two blocks into a larger block in a straight or inverted order. Rule 3 translates a source phrase f to a target phrase e and generate a block A. These three rules are continuously used until the whole source sentence is covered, and a hierarchical segmentation tree of the source sentence is generated at the same time.

Based on this hierarchical setting, Xiong et al. [2006] treat the reordering problem as a classification with two labels: straight and inverted between two consecutive blocks, and build up a maximum entropy classification model as the reordering model:

$$p(o|c(A^1, A^2)) = \frac{\exp(\sum_i \theta_i f_i(o, c(A^1, A^2)))}{\sum_{o'} \exp(\sum_i \theta_i f_i(o', c(A^1, A^2)))} \tag{8.32}$$

where θ_i are the feature weights, $c(A^1, A^2)$ indicates the attributes of

A^1 and A^2, and $f_i(o, c(A^1, A^2))$ are binary features defined as,

$$f_i(o, c(A^1, A^2)) = \begin{cases} 1, & \text{if } (o, c(A^1, A^2)) \text{ satisfies certain condition} \\ 0, & \text{else} \end{cases} \tag{8.33}$$

This framework only uses the features extracted from the blocks instead of the whole block (in contrast to lexicalized reordering), thus it is flexible to reorder any blocks [Xiong et al., 2006].

Adding Topic-based Features Following this framework, Wang et al. [2014] integrate two more types of topic-based features into the reordering model, in additional to the boundary word features used in Xiong et al. [2006]. First, they choose the topic with maximum probability in a document to be the *document topic feature* for that document. Besides, they also use the topics of the content words that locate at the left and rightmost positions on the source phrases as the *word topic features* to capture topic-sensitive reordering patterns.

During the decoding process, Xiong et al. [2006] infer the topic distributions of the test documents first and then apply this proposed topic-based reordering model as one sub-model to the log-linear model to obtain the best translation:

$$e_{\texttt{best}} = \texttt{argmax}_e \Big\{ \sum_{m=1}^{M} \lambda_m h_m(e, f) \Big\} \tag{8.34}$$

where $h_m(e, f)$ are the sub-models or features of the whole log-linear model, λ_m are their weights accordingly, which are tuned on the development set.

This framework is very flexible to encode any topic-based features, and any multilingual topic models we have discussed so far can be applied to extract better topics.

8.5 Beyond Domain Adaptation

In addition to translation models, language models and reordering models, there are also other modules of SMT, such as word alignment, where topic models have also been applied. Zhao and Xing [2006] build up a novel bilingual topical admixture (BiTAM) to improve the word alignment in SMT. BiTam assumes each document pair is an admixture of topics, and the topics for each sentence pair within that document pair are sampled from the same document-topic distribution. Each topic also has a topic-specific translation table. Therefore, the sentence-level word alignment and translations are coupled by the hidden topics. This BiTam model captures the latent topical structure and generalizes word alignments and translations via topics shared across sentence pairs, thus the quality of the alignments is improved.

Besides, coherence, which ties sentences of text into a meaningfully connected structure [Xiong and Zhang, 2013], is another important piece to SMT.Xiong and Zhang [2013] introduce a topic-based coherence model to improve the document translation quality. They learn the sentence topic for source documents, based on which they predict the target topic chain; they then incorporate the predicted target coherence chain into the document translation decoding process.

In general, multilingual topic models obtain topics with high quality, since different languages can complement each other to reduce topic ambiguity. Many different approaches, have been explored to apply such multilingual topic models to improve different pieces of statistical machine translation. With such topic knowledge, the variations of different languages can be better captuered to make the translations more natural and coherent.

9

Conclusion

While we have attempted to cover a variety of the applications of topic models to help individuals navigate large text datasets, no finite survey could enumerate all of the applications of topic models in text, which have been applied to part of speech tagging [Toutanova and Johnson, 2008], word sense induction [Brody and Lapata, 2009], and entity disambiguation [Kataria et al., 2011]. It goes without saying that we have also omitted many other applications outside text, such as biology Pritchard et al. [2000], understanding source code Maskeri et al. [2008], music analysis Hu and Saul [2009], and many more.

9.1 Coping with Information Overload

A challenge in topic modeling is how to make inference efficient enough to both scale to large datasets and to provide low-latency interactive experiences to help provide support to a user in the loop. Three broad directions support this efficient learning of large topic models. There are three broad strategies for processing documents more quickly.

The first is through decreasing the average number of times a computer needs to look at a document to learn a topic model; i.e., to improve *throughput.* Online algorithms Hoffman et al. [2010] only look at a document once, update the topics, and then move on to the next document. This is often much faster than batch approaches which require many passes over the same set of documents. Another option is to *distribute* computation across many machines [Zhai et al., 2012].

A complementary approach is to reduce the time a computer spends on any particular document: improving *efficiency.* This is possible by improving how long it takes to sample document assignments [Yao et al., 2009, Li et al., 2014] or compute variational parameters [Mimno et al., 2012].

The final approach to improve the efficiency of probabilistic algorithms for topic models is to rethink the inference process entirely. Novel approaches view topic model inference as a factorization of a co-occurence matrix [Arora et al., 2013] or as a spectral decomposition [Anandkumar et al., 2012b]. These approaches often are much faster than traditional approaches as they use word types—rather than documents—as the central unit of computation.

9.2 Deeper Representations

Part of the benefit of topic models is that the topic distribution of a document (θ) serves as a low-dimensional representation of what the document means. This numerical vector is useful for finding similar documents (Chapter 2), displaying documents to a user (Chapter 3), or connecting documents across languages (Chapter 8).

Increasingly, vector-based representations have been useful "all the way down". Vector-based representations of words and phrases can improve next word prediction [Bengio et al., 2003], sentiment analysis [Socher et al., 2012], and translation [Devlin et al., 2014]. And this is not just for text—representation learning has taken hold of speech, vision, and machine learning generally.

The impact of representation learning on topic modeling remains unclear as we go to press in 2017. We see several ways that representation learning and topic modeling could benefit each other in the future.

Evaluation Evaluation methods from topic models have made their way into representation learning, which suggests that some of the lessons learned in making topic models interpretable (Section 3) could also be applied in representation learning. This could help mollify some critics of representation learning who argue that the results are often uninterpretable or deceptive [Szegedy et al., 2013].

Synthesis Topic modeling is also blending with more expressive latent representation models [Ranganath et al., 2015]. Topic models could help representation learning solve some of its difficulty summarizing larger segments of text. Paragraphs and sentences are difficult to model as a single vector, and techniques more sophisticated than simple averaging don't seem worth the hassle [Iyyer et al., 2015].

Parallel Evolution Another possible path is less intertwined: topic models and deep-learning representation learning could peacefully coexist for some time. Topic models offer advantages of speed and interpretability, while representation learning can do better for prediction-based tasks. They would well become tools that many text miners have in their toolkit, with specific circumstances for using either.

9.3 Automatic Text Analysis for the People

However, in our view, the primary research challenge of topic models is not to make these models and their inference more complicated but rather to make them more accessible. As we've described, topic models can help scholars and ordinary people navigate large text collections.

However, using topic models still requires extensive data and computer skills. Our job as information scientists is not complete until these tools (or suitable alternatives) are available to everyone who needs them.

This requires making the tools more usable: the preprocessing often required of topic models is not straightforward: should we remove non-English documents, what should we consider a document, how should we handle metadata? Nor are the modeling choices needed to make sense of the data trivial: how many topics should we use, which of the many possible models should we use, what inference technique gives us the best tradeoff between speed and accuracy. Existing topic models do a poor job of communicating what options are available to a user and what consequences these choices have.

However, even if the process of creating a topic model becomes intuitive, the output must also be interpretable. Distributions over words are the language that these models use to create representations of document collections, but it is not how users think about topics: they would much rather have phrases [Mei et al., 2007], sentences [Smith et al., 2016], or pictures [Aletras et al., 2014b]. However, providing these representations is non-trivial and requires a deeper understanding of a corpus than today's topic models can manage.

Finally, topic models need a more systematic investigation of how they can assist users' workflow for typical information seeking, organization, and management tasks. While the applications covered in this survey show examples of how people can use topic models from applications from history to political science, how topic models can augment or replace existing workflows lacks the same attention given to—for example—search engines.

9.4 Coda

We hope that you have enjoyed our survey of topic models' applications. For further information, we'd encourage the reader to investigate the topic modeling bibliography,[1] join the topic modeling mailing list,[2], or the book's associated webpage.

[1] https://mimno.infosci.cornell.edu/topics.html

[2] https://lists.cs.princeton.edu/mailman/listinfo/topic-models

References

Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, 2008. ISSN 1532-4435.

Nikolaos Aletras, Timothy Baldwin, Jey Han Lau, and Mark Stevenson. Representing topics labels for exploring digital libraries. In *Proceedings of the IEEE/ACM Joint Conference on Digital Libraries*, pages 239–248, 2014a.

Nikolaos Aletras, Timothy Baldwin, Jey Han Lau, and Mark Stevenson. Representing topics labels for exploring digital libraries. In *Proceedings of the IEEE/ACM Joint Conference on Digital Libraries*, pages 239–248, 2014b.

Mark Algee-Hewitt, Ryan Heuser, and Franco Moretti. On paragraphs. scale, themes, and narrative form. *Stanford Literary Lab Pamphlets*, 1(10), October 2015.

Anima Anandkumar, Dean P. Foster, Daniel Hsu, Sham Kakade, and Yi-Kai Liu. A spectral algorithm for latent Dirichlet allocation. In *Proceedings of Advances in Neural Information Processing Systems*, 2012a.

Anima Anandkumar, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Yi kai Liu. A spectral algorithm for latent dirichlet allocation. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 917–925. Curran Associates, Inc., 2012b. URL `http://papers.nips.cc/paper/4637-a-spectral-algorithm-for-latent-dirichlet-allocation.pdf`.

David Andrzejewski and David Buttler. Latent topic feedback for information retrieval. In *Knowledge Discovery and Data Mining*, 2011.

David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In *Proceedings of the International Conference of Machine Learning*, 2009.

Sanjeev Arora, Rong Ge, Yoni Halpern, David M. Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the International Conference of Machine Learning*, 2013.

Anton Bakalov, Andrew McCallum, Hanna Wallach, and David Mimno. Topic models for taxonomies. In *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '12, 2012.

J. Bellegarda. A latent semantic analysis framework for large-span language modeling. In *European Conference on Speech Communication and Technology*, 1997.

Jerome R. Bellegarda. Statistical language model adaptation: review and perspectives. volume 42, pages 93–108, 2004.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March 2003. ISSN 1532-4435. URL `http://dl.acm.org/citation.cfm?id=944919.944966`.

A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999.

Indrajit Bhattacharya. Collective entity resolution in relational data. *PhD Dissertation, University of Maryland, College Park*, 2006.

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

David M. Blei and Jon D. McAuliffe. Supervised topic models. In *Proceedings of Advances in Neural Information Processing Systems*. 2007.

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

David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):7:1–7:30, February 2010.

Shannon Bowen. Pseudo-events pay dividends from cleopatra to chipotle. *PR Week*, 2016. URL `http://www.prweek.com/article/1403267/pseudo-events-pay-dividends-cleopatra-chipotle#GZy4D2fyxLKZairV.99`.

G.E.P. Box and N.R. Draper. *Empirical model-building and response surfaces*. Wiley series in probability and mathematical statistics: Applied probability and statistics. Wiley, 1987. ISBN 9780471810339. URL `https://books.google.com/books?id=QO2dDRufJEAC`.

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

Jordan Boyd-Graber and Philip Resnik. Holistic sentiment analysis across languages: Multilingual supervised latent Dirichlet allocation. In *Proceedings of Empirical Methods in Natural Language Processing*, 2010.

Jordan Boyd-Graber, David M. Blei, and Xiaojin Zhu. A topic model for word sense disambiguation. In *Proceedings of Empirical Methods in Natural Language Processing*, 2007.

Percy Williams Bridgman. *The logic of modern physics*. Macmillan, New York, 1927.

Samuel Brody and Mirella Lapata. Bayesian word sense induction. In *Proceedings of the European Chapter of the Association for Computational Linguistics*, Athens, Greece, 2009.

Andre David Broniatowski, Mark Dredze, J. Michael Paul, and Andrea Dugas. Using social media to perform local influenza surveillance in an inner-city hospital: A retrospective observational study. *JMIR Public Health Surveill*, 1(1):e5, May 2015. . URL `/2015/1/e5/`.

John Burrows. Delta: a measure of stylistic difference and a guide to likely authorship. *Lit Linguist Computing*, 17(3):267–287, 2002.

Allison Chaney and David Blei. Visualizing topic models. In *International AAAI Conference on Weblogs and Social Media*, 2012a. URL `https://www.aaai.org/ocs/index.php/ICWSM/ICWSM12/paper/view/4645/5021`.

Allison June-Barlow Chaney and David M. Blei. Visualizing topic models. In *Proceedings of the AAAI Conference on Weblogs and Social Media*, 2012b.

Jonathan Chang and David M. Blei. Relational topic models for document networks. In *Proceedings of Artificial Intelligence and Statistics*, 2009.

Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. Reading tea leaves: How humans interpret topic models. In *Proceedings of Advances in Neural Information Processing Systems*, 2009.

Boxing Chen, George Foster, and Roland Kuhn. Adaptation of reordering models for statistical machine translation. In *Proceedings of the Association for Computational Linguistics*, 2013.

David Chiang, Steve DeNeefe, and Michael Pust. Two easy improvements to lexical weighting. In *Proceedings of the Human Language Technology Conference*, 2011.

Jaegul Choo, Changhyun Lee, Chandan K. Reddy, and Haesun Park. UTOPIAN: User-driven topic modeling based on interactive nonnegative matrix factorization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):1992–2001, 2013.

Jason Chuang, Christopher D. Manning, and Jeffrey Heer. Termite: Visualization techniques for assessing textual topic models. In *Advanced Visual Interfaces*, 2012. URL `http://vis.stanford.edu/papers/termite`.

Jason Chuang, Margaret E. Roberts, Brandon M. Stewart, Rebecca Weiss, Dustin Tingley, Justin Grimmer, and Jeffrey Heer. Topiccheck: Interactive alignment for assessing topic model stability. In *NAACL-HLT*, 2015. URL `http://idl.cs.washington.edu/papers/topic-check`.

P.R. Clarkson and A. J. Robinson. Language model adaptation using mixtures and an exponentially decaying cache. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 799–802, 1997.

Noah Coccaro and Daniel Jurafsky. Towards better integration of semantic predictors in statistical language modeling. In *International Conference on Acoustics, Speech, and Signal Processing*, 1998.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006. ISSN 1532-4435.

W.B. Croft and J. Lafferty. Language modeling for information retrieval. In *Kluwer International Series on Information Retrieval*, 2003.

Scott Deerwester, Susan Dumais, Thomas Landauer, George Furnas, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. Fast and robust neural network joint models for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1370–1380, Baltimore, Maryland, June 2014. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P14-1129`.

Vladimir Eidelman, Jordan Boyd-Graber, and Philip Resnik. Topic models for dynamic translation model adaptation. In *Proceedings of the Association for Computational Linguistics*, 2012.

Jacob Eisenstein, Brendan O'Connor, Noah A. Smith, and Eric P. Xing. A latent variable model for geographic lexical variation. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 1277–1287, 2010.

Jacob Eisenstein, Duen Horng Chau, Aniket Kittur, and Eric Xing. TopicViz: interactive topic exploration in document collections. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems*, pages 2177–2182. ACM, 2012.

Jacob Eisenstein, Iris Sun, and Lauren F. Klein. Exploratory text analysis for large document archives. In *Digital Humanities*, 2014.

George Foster and Roland Kuhn. Mixture-model adaptation for smt. In *Proceedings of the Second Workshop on Statistical Machine Translation*, 2007.

Jianfeng Gao, Kristina Toutanova, and Wen tau Yih. Clickthrough-based latent semantic models for web search. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2011.

Jianfeng Gao, Shasha Xie, Xiaodong He, and Alnur Ali. Learning lexicon models from search logs for query expansion. In *Proceedings of Empirical Methods in Natural Language Processing*, 2012.

Matthew Gardner, Joshua Lutes, Jeff Lund, Josh Hansen, Dan Walker, Eric Ringger, and Kevin Seppi. The topic browser: An interactive tool for browsing topic models. In *Proceedings of Advances in Neural Information Processing Systems*, 2010a.

Matthew J Gardner, Joshua Lutes, Jeff Lund, Josh Hansen, Dan Walker, Eric Ringger, and Kevin Seppi. The topic browser: An interactive tool for browsing topic models. In *Proceedings of the NIPS Workshop on Challenges of Data Visualization*, 2010b.

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

Daniel Gildea and Thomas Hofmann. Topic-based language models using em. In *EEuropean Conference on Speech Communication and Technology*, 1999.

Andrew Goldstone and Ted Underwood. The quiet transformations of literary studies: What thirteen thousand scholars could tell us. *New Literary History*, 45(3), Summer 2014.

Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl 1):5228–5235, 2004.

Amit Gruber, Michael Rosen-Zvi, and Yair Weiss. Hidden topic Markov models. In *Artificial Intelligence and Statistics*, 2007.

Jacob Harris. Word clouds considered harmful. `http://www.niemanlab.org/2011/10/word-clouds-considered-harmful/`, 2011.

Eva Hasler, Barry Haddow, and Philipp Koehn. Sparse lexicalised features and topic adaptation for SMT. In *Proceedings of IWSLT*, 2012.

L. Hirschman and R. Gaizauskas. Natural language question answering: The view from here. *Nat. Lang. Eng.*, 7(4):275–300, December 2001. ISSN 1351-3249. . URL `http://dx.doi.org/10.1017/S1351324901002807`.

Matthew Hoffman, David M. Blei, and Francis Bach. Online learning for latent Dirichlet allocation. In *Proceedings of Advances in Neural Information Processing Systems*, 2010.

Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of Uncertainty in Artificial Intelligence*, 1999a.

Thomas Hofmann. Probabilistic latent semantic indexing. *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999b.

Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

Diane Hu and Lawrence K. Saul. A probabilistic model of unsupervised learning for musical-key profiles. In *International Society for Music Information Retrieval Conference*, 2009.

Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. Interactive topic modeling. *Mach. Learn.*, 95(3):423–469, June 2014a. ISSN 0885-6125. . URL `http://dx.doi.org/10.1007/s10994-013-5413-0`.

Yuening Hu, Ke Zhai, Vladimir Edelman, and Jordan Boyd-Graber. Polylingual tree-based topic models for translation domain adaptation. In *Association for Computational Linguistics*, 2014b.

R. Iyer and M. Ostendorf. Modeling long distance dependencies in language: topic mixtures versus dynamic cache models. 7:236–239, 1999.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. Deep unordered composition rivals syntactic methods for text classification. In *Association for Computational Linguistics*, 2015. URL `docs/2015_acl_dan.pdf`.

Fred Jelinek and Robert L. Mercer. Interpolated estimation of Markov source parameters from sparse data. In *Proceedings, Workshop on Pattern Recognition in Practice*, pages 381–397, 1980.

Matt L. Jockers. *Macroanalysis: Digital Methods and Literary History*. Topics in the Digital Humanities. University of Illinois Press, 2013. ISBN 9780252094767. URL `http://books.google.com/books?id=mPOdxQgpOSUC`.

Matthew Jockers and David Mimno. Significant themes in 19th century literature. *Poetics*, 41(6):750–769, December 2013.

Patrick Juola. Authorship attribution. *Foundations and Trends in information Retrieval*, 1(3):233–334, 2006.

Saurabh S. Kataria, Krishnan S. Kumar, Rajeev R. Rastogi, Prithviraj Sen, and Srinivasan H. Sengamedu. Entity disambiguation with hierarchical topic models. In *Knowledge Discovery and Data Mining*, 2011.

S. M. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. In *IEEE Transaction on Acoustics, Speech and Signal Processing*, 1987.

Reinhard Kneser and Jochen Peters. Semantic clustering for adaptive language modeling. In *International Conference on Acoustics, Speech, and Signal Processing*, 1997.

Reinhard Kneser, Jochen Peters, and Dietrich Klakow. Language model adaptation using dynamic marginals. In *European Conference on Speech Communication and Technology*, 1997.

Philipp Koehn. *Statistical Machine Translation*. Cambridge University Press, 2009. ISBN 9780521874151. URL `http://books.google.com/books?id=D21UAAAACAAJ`.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2003.

Thomas K Landauer and Michael L Littman. Fully automatic cross-language document retrieval using latent semantic indexing. In *Proceedings of the 6 th Annual Conference of the UW Centre for the New Oxford English Dictionary*, 1990.

John Langford, Lihong Li, and Alex Strehl. Vowpal Wabbit, 2007.

Mark A. Largent and Julia I. Lane. STAR METRICS and the Science of Science Policy. *Review of Policy Research*, 29(3):431–438, 2012. . URL `http://dx.doi.org/10.1111/j.1541-1338.2012.00567.x`.

Jey Han Lau, David Newman, Sarvnaz Karimi, and Timothy Baldwin. Best topic word selection for topic labelling. In *Coling 2010: Posters*, pages 605–613, Beijing, China, August 2010. URL `http://www.aclweb.org/anthology/C10-2069`.

Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. Automatic labelling of topic models. In *Proceedings of the Association for Computational Linguistics*, pages 1536–1545, 2011.

Victor Lavrenko and W. Bruce Croft. Relevance based language models. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

Aaron Q Li, Amr Ahmed, Sujith Ravi, and Alexander J Smola. Reducing the sampling complexity of topic models. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 891–900. ACM, 2014.

Chenghua Lin and Yulan He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2009. ISBN 978-1-60558-512-3. .

Yue Lu, Qiaozhu Mei, and ChengXiang Zhai. Investigating task performance of probabilistic topic models: an empirical study of plsa and lda. *Information Retrieval*, 14(2):178–203, 2011.

David J. C. Mackay and Linda C. Bauman Petoy. A hierarchical Dirichlet language model. *Natural Language Engineering*, 1:1–19, 1995.

Xian-Ling Mao, Zhao-Yan Ming, Zheng-Jun Zha, Tat-Seng Chua, Hongfei Yan, and Xiaoming Li. Automatic labeling hierarchical topics. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2383–2386, New York, NY, USA, 2012. ACM. ISBN 978-1-4503-1156-4. . URL `http://doi.acm.org/10.1145/2396761.2398646`.

Mitchell P. Marcus, Beatrice Santorini, and Mary A. Marcinkiewicz. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

Girish Maskeri, Santonu Sarkar, and Kenneth Heafield. Mining business topics in source code using latent dirichlet allocation. In *ISEC*, 2008. ISBN 978-1-59593-917-3. .

Spyros Matsoukas, Antti-Veikko I. Rosti, and Bing Zhang. Discriminative corpus weight estimation for machine translation. In *Proceedings of Empirical Methods in Natural Language Processing*, 2009.

Andrew Kachites McCallum. Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet, 2002.

Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. Automatic labeling of multinomial topic models. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 490–499, 2007.

Ian Matthew Miller. Rebellion, crime and violence in qing china, 17221911: A topic modeling approach. *Poetics*, 41(6):626–649, December 2013.

David Mimno. Computational historiography: Data mining in a century of classics journals. *J. Comput. Cult. Herit.*, 5(1):3:1–3:19, April 2012. ISSN 1556-4673. . URL `http://doi.acm.org/10.1145/[illegible]0165.2160168`.

David Mimno and Andrew McCallum. Topic Models Conditioned on Arbitrary Features with Dirichlet-multinomial Regression. In *Proceedings of the 2008 Conference on Uncertainty in Artificial Intelligence (UAI)*, 2008. URL `http://www.cs.umass.edu/~{}mccallum/papers/dmr-uai.pdf`.

David Mimno, Hanna Wallach, Jason Naradowsky, David Smith, and Andrew McCallum. Polylingual topic models. In *Proceedings of Empirical Methods in Natural Language Processing*, 2009.

David Mimno, Hanna Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of Empirical Methods in Natural Language Processing*, 2011.

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

Franco Moretti. The slaughterhouse of literature. *Modern Language Quarterly*, 61(1):207–227, 2000.

Franco Moretti. *Distant Reading*. Verso, 2013a. ISBN 9781781680841. URL `https://books.google.com/books?id=YKMCy9I3PG4C`.

Franco Moretti. Operationalizing, or the function of measurement in literary theory. *New Left Review*, 84, Nov/Dec 2013b.

Frederick Mosteller and David L. Wallace. *Inference and Disputed Authorship: The Federalist*. Addison-Wesley, Reading, Mass., 1964.

Christof Müller and Iryna Gurevych. A study on the semantic relatedness of query and document terms in information retrieval. In *Proceedings of Empirical Methods in Natural Language Processing*, pages 1338–1347, 2009.

R. Nallapati and W. Cohen. Link-PLSA-LDA: A new unsupervised model for topics and influence of blogs. In *International Conference on Weblogs and Social Media*, 2008.

Shravan Narayanamurthy. Yahoo! lda, 2011. URL `https://github.com/shravanmn/Yahoo_LDA/wiki`.

D. Newman, A. Asuncion, P. Smyth, and M. Welling. Distributed Inference for Latent Dirichlet Allocation. In *Proceedings of Advances in Neural Information Processing Systems*. 2008.

David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. Automatic evaluation of topic coherence. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

David J. Newman and Sharon Block. Probabilistic topic decomposition of an eighteenth-century american newspaper. *Journal of the American Society for Information Science and Technology*, 18(1):753–767, 2006.

H. Ney, U. Essen, and R. Kneser. On structuring probabilistic dependencies in stochastic language modelling. *Computer Speech and Language*, 8:1–38, 1994.

Thang Nguyen, Jordan Boyd-Graber, Jeff Lund, Kevin Seppi, and Eric Ringger. Is your anchor going up or down? Fast and accurate supervised topic models. In *North American Association for Computational Linguistics*, 2015a. URL `docs/2015_naacl_supervised_anchor.pdf`.

Viet-An Nguyen, Jordan Boyd-Graber, and Philip Resnik. Lexical and hierarchical topic regression. In *Proceedings of Advances in Neural Information Processing Systems*, 2013.

Viet-An Nguyen, Jordan Boyd-Graber, Philip Resnik, and Jonathan Chang. Learning a concept hierarchy from multi-labeled documents. In *Neural Information Processing Systems*, 2014.

Viet-An Nguyen, Jordan Boyd-Graber, Philip Resnik, and Kristina Miler. Tea party in the house: A hierarchical ideal point topic model and its application to republican legislators in the 112th congress. In *Association for Computational Linguistics*, 2015b.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the Web. Stanford Digital Library Working Paper SIDL-WP-1999-0120, Stanford University, 1999.

Bo Pang and Lillian Lee. *Opinion Mining and Sentiment Analysis*. Now Publishers Inc, 2008. ISBN 1601981503.

Laurence A. Park and Kotagiri Ramamohanarao. The sensitivity of latent dirichlet allocation for information retrieval. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases*, 2009.

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

Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281, New York, NY, USA, 1998. ACM Press. ISBN 1-58113-015-5. .

Jonathan K. Pritchard, Matthew Stephens, and Peter Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155:945–959, 2000.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of Empirical Methods in Natural Language Processing*, 2009.

Daniel Ramage, Susan T. Dumais, and Daniel J. Liebling. Characterizing microblogs with topic models. In *International Conference on Weblogs and Social Media*, 2010a.

Daniel Ramage, Christopher D. Manning, and Daniel A. Mcfarland. Which universities lead and lag? toward university rankings based on scholarly output. In *In Proc. of NIPS Workshop on Computational Social Science and the Wisdom of the Crowds*, 2010b.

Rajesh Ranganath, Linpeng Tang, Laurent Charlin, and David Blei. Deep exponential families. In *Proceedings of Artificial Intelligence and Statistics*, 2015.

Philip Resnik and Eric Hardisty. Gibbs sampling for the uninitiated. Technical report, University of Maryland, 2009. URL `http://www.umiacs.umd.edu/~resnik/pubs/gibbs.pdf`.

Lia M. Rhody. Topic modeling and figurative language. *Journal of Digital Humanities*, 2(1), 2012.

Margaret E. Roberts, Brandon M. Stewart, and Dustin Tingley. *stm: R Package for Structural Topic Models*, 2014. URL `http://www.structuraltopicmodel.com`. R package version 1.0.8.

J. Rocchio. Relevance feedback in information retrieval. In G. Salton, editor, *The Smart retrieval system - experiments in automatic document processing*, pages 313–323. Englewood Cliffs, NJ: Prentice-Hall, 1971.

Nick Ruiz and Marcello Federico. Topic adaptation for lecture translation through bilingual latent semantic models. In *WMT Workshop on Statistical Machine Translation*, 2011.

Gerard. Salton. *Automatic Information Organization and Retrieval.* McGraw Hill Text, 1968. ISBN 0070544859.

Kristie Seymore and Ronald Rosenfeld. Using story topics for language model adaptation. In *European Conference on Speech Communication and Technology*, 1997.

Kristie Seymore, Stanley F. Chen, and Ronald Rosenfeld. Nonlinear interpolation of topic models for language model adaptation. In *The 5th International Conference on Spoken Language Processing, Incorporating The 7th Australian International Speech Science and Technology Conference*, 1998.

Alison Smith, Jason Chuang, Yuening Hu, Jordan Boyd-Graber, and Leah Findlater. Concurrent visualization of relationships between words and topics in topic models. In *ACL Workshop on Workshop on Interactive Language Learning, Visualization, and Interfaces*, 2014.

Alison Smith, Sana Malik, and Ben Shneiderman. Visual analysis of topical evolution in unstructured text: Design and evaluation of topicflow. In *Applications of Social Media and Social Network Analysis*, pages 159–175. Springer, 2015.

Alison Smith, Tak Yeon Lee, Forough Poursabzi-Sangdeh, Leah Findlater, Jordan Boyd-Graber, and Niklas Elmqvist. Evaluating visual representations for topic understanding and their effects on manually generated labels. *Transactions of the Association for Computational Linguistics*, 2016.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. Semantic compositionality through recursive matrix-vector spaces. In *Proceedings of Empirical Methods in Natural Language Processing*, 2012.

F. Song and W.B. Croft. A general language model for information retrieval. In *International Conference on Information and Knowledge Management*, 1999.

Stan Development Team. Stan: A c++ library for probability and sampling, version 2.5.0, 2014. URL `http://mc-stan.org/`.

Jinsong Su, Hua Wu, Haifeng Wang, Yidong Chen, Xiaodong Shi, Huailin Dong, and Qun Liu. Translation model adaptation for statistical machine translation with monolingual topic information. In *Proceedings of the Association for Computational Linguistics*, 2012.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2013. URL `http://arxiv.org/abs/1312.6199`.

Rick Szostak. Classifying science. *Classifying Science: Phenomena, Data, Theory, Method, Practice*, pages 1–22, 2004.

Edmund M. Talley, David Newman, David Mimno, Bruce W. Herr, Hanna M. Wallach, Gully A. P. C. Burns, A. G. Miriam Leenders, and Andrew McCallum. Database of NIH grants using machine-learned categories and graphical clustering. *Nature Methods*, 8(6):443–444, May 2011. ISSN 1548-7091.

Yik-Cheung Tam, Ian Lane, and Tanja Schultz. Bilingual-lsa based lm adaptation for spoken language translation. In *Proceedings of the Association for Computational Linguistics*, 2007.

Timothy R. Tangherlini and Peter Leonard. Trawling in the sea of the great unread: Sub-corpus topic modeling and humanities research. *Poetics*, 41 (6):725–749, December 2013.

Kristina Toutanova and Mark Johnson. A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In *Proceedings of Advances in Neural Information Processing Systems*, pages 1521–1528. 2008.

Fernanda B. Viégas and Martin Wattenberg. Tag clouds and the case for vernacular visualization. *Interactions*, 15(4):49–52, 2008.

Hanna Wallach, David Mimno, and Andrew McCallum. Rethinking LDA: Why priors matter. In *Proceedings of Advances in Neural Information Processing Systems*, 2009.

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

Shiliang Wang, J. Michael Paul, and Mark Dredze. Social media as a sensor of air quality and public response in china. *J Med Internet Res*, 17(3):e22, Mar 2015. . URL `http://www.ncbi.nlm.nih.gov/pubmed/25831020`.

Xing Wang, Deyi Xiong, Min Zhang, Yu Hong, and Jianmin Yao. A topic-based reordering model for statistical machine translation. *Natural Language Processing and Chinese Computing*, 496:414–421, 2014.

Xing Wei. *Topic Models in Information Retrieval*. Ph.D. dissertation, University of Massachusetts Amherst, 2007.

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

Theresa Wilson and Janyce Wiebe. Annotating attributions and private states. In *CorpusAnno '05: Proceedings of the Workshop on Frontiers in Corpus Annotations II*, 2005.

Frank Wood and Yee Whye Teh. A hierarchical nonparametric Bayesian approach to statistical language model domain adaptation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 12, 2009.

Dekai Wu. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. 23:377–403, 1997.

Xinyan Xiao, Deyi Xiong, Min Zhang, Qun Liu, and Shouxun Lin. A topic similarity model for hierarchical phrase-based translation. In *Proceedings of the Association for Computational Linguistics*, 2012.

Deyi Xiong and Min Zhang. A topic-based coherence model for statistical machine translation. In *Association for the Advancement of Artificial Intelligence*, 2013.

Deyi Xiong, Qun Liu, and Shouxun Lin. Maximum entropy based phrase reordering model for statistical machine translation. In *Proceedings of the Association for Computational Linguistics*, 2006.

Tze-I Yang, Andrew J. Torget, and Rada Mihalcea. Topic modeling on historical newspapers. In *ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, 2011.

Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *Knowledge Discovery and Data Mining*, 2009. ISBN 978-1-60558-495-9.

Xing Yi and James Allan. A comparative study of utilizing topic models for information retrieval. In *ECIR*, volume 5478 of *Lecture Notes in Computer Science*, pages 29–41. Springer, 2009.

Heng Yu, Jinsong Su, Yajuan Lv, and Qun Liu. A topic-triggered language model for statistical machine translation. In *International Joint Conference on Natural Language Processing*, 2013.

Jia Zeng, W. K. Cheung, and Jiming Liu. Learning topic models by belief propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(5):1121–1134, 2013. ISSN 0162-8828. .

C. Zhai and J Lafferty. A study of smoothing methods for language models applied to information retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001a.

C. Zhai and J Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2001b.

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

Bing Zhao and Eric P. Xing. BiTAM: Bilingual topic admixture models for word alignment. In *Proceedings of the Association for Computational Linguistics*, 2006.

Jun Zhu, Amr Ahmed, and Eric P. Xing. MedLDA: maximum margin supervised topic models for regression and classification. In *Proceedings of the International Conference of Machine Learning*, 2009. ISBN 978-1-60558-516-1. .