Text as Actuator: Text-Driven Response Modeling and Prediction in Politics

Tae Yano

CMU-LTI-13-006

Language Technologies Institute
School of Computer Science
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213
www.lti.cs.cmu.edu

Thesis Committee:

Noah A. Smith
William W. Cohen
Jason I. Hong
Philip Resnik

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
In Language and Information Technologies*

Contents

Abstract

In this work, we develop a series of prediction tasks on ”actuating text”, defined as text that evokes – or is written to evoke – responses from its readership. Examples include blog posts with reader comments and product reviews with social tagging or ratings. Some traditional text collections, such as legislative proceedings, can also be seen as varieties of this type of text: legislative actions associated with a bill are, after all, the legislature’s collective reaction to the original text.

This thesis examines response prediction tasks in two distinct domains in contemporary U.S. politics: the political blogosphere and the United States Congress. In the blogosphere, we examine the relation between political topics and user comments they generate among the highly partisan readership community. In the U.S. Congress, we examine how bills survive the congressional committee system, a highly selective scrutinizing phase that happens before the general voting, and the relationship between a congress person’s microblog messages and the campaign contribution they receive from interest groups.

We propose several probabilistic models to predict attributes of the responses based on statistical analysis of the associated texts. We anticipate that such models will ultimately prove useful in user assistive applications like recommendation and filtering systems, or serve as a technique to gather critical intelligence for scholars, content providers, or whoever is interested in learning the response trends among the target populations. In addition to achieving high prediction accuracy, we aim to shed llight on the underlying response process, thereby contributing to social and political science research.

Acknowledgement

I would like to thank my thesis advisors, Noah A. Smith and William W. Cohen for their constant support and encouragement. I am in fact a lucky one to land on their guidance. I have always tried to live up to their intellectual standards. I hope I have sometimes succeeded.

I would like to thank the members of my dissertation committee, Philip Resnik and Jason I. Hong. I have deeply admired their work, intelligence, and their research vision. I am very thankful for their advice, kindness and patience.

The work I present here is supported by great individuals who I have a good fortune to be acquainted with. They are my friends, co-workers, and co-authors: Michael Heilman, Dipanjan Das, Shay Cohen, Kevin Gimpel, Andre Martins, Nathan Schneider, Brendan O'Connor, Dani Yogatama, Yanchuan Sim, Victor Chahuneau, Lingpeng Kong, David Bamman, Waleed Ammar, Sam Thomson, Bryan Routledge, Naomi Saphra, Behrang Mohit, Cari Sisson Bader, William Yang Wang, Dana Movshovitz-Attias, Bhavana Dalvi Mishra, Malcolm Greaves, Katie Rivard Mazaitis, Nan Li, Ramnath Balasubramanyan, Mahesh Joshi, Richard C. Wang, Frank Lin, Ni Lao, Andrew Arnold, Noboru Matsuda, John D. Wilkerson, Jacob Eisenstein, Chris Dyer, Daniel Preotiuc-Pietro, Michael Gamon, Patric Pantel, and Justin Cranshaw.

I am obliged to single out a few individuals here for their help directly relevant to this thesis. The key idea behind the“impact” score we used in the third chapter was formed through the discussion with Brendan O'Connor. John D. Wilkerson at University of Washington provided us with the variable data on the congressional committee. He is also the most patient co-author I have ever met. Nathan Schneider have provided me with the countless editorial helps for many of my research publications, including this dissertation. His kindness seems boundless at times.

I am fortunate to have met two great female computer scientists in my earlier career. They provided me with great roll models: Julia Hirschberg is a professor at Columbia University, where I obtained my master of science in Computer Science. She was my academic advisor. Becky Passonneau is a computational linguist at Columbia University. She supervised my first honest research work at Columbia.

I would like to thank the great instructors in Computer Science I have met in the past, Subash Shankar, Stewart Weiss and Jeffrey Ely.

Chapter 1

Introduction

We will develop a series of prediction tasks on *actuating* text in this work. In our context, actuating text is a text which evokes, or is written to evoke, ***responses*** from its readership. Pragmatically, we use the term to refer to a text collection with coupled observations on ***reactions*** from the real world.

Many types of online document collections fit this description. Examples include blog posts with readership comments, product reviews with collaborative tagging or ratings, and news stories amplified and spread by quoting or forwarding. Some long-existing corpora, such as congressional bill collections or floor debate manuscripts, can be seen as variations of actuating text, as voting results or amendments are, in a sense, a collective reaction from the legislative body to the bill or deliberation. Note that, as we defined them, actuating texts do not need to be user generated texts (UGTs) or of social media, although these are perhaps the most visible examples today. The increased visibility of social media is certainly a big factor to motivate response predictions such as ours.

The main goal of this dissertation is to deliver novel data-driven ***prediction*** models on responses based on statistical analyses of the associated texts. Corpus-based prediction models are useful in many types of real world applications. Moreover, the interactions between texts and response could reveal a variety of interesting social meanings. In this dissertation, we will consider a few distinctive kinds of document collections, each with novel prediction tasks related to ***politics*** in the United States.

1.1 Text and Response Prediction

Why should we care about predicting response from text? First, community-oriented documents such as those mentioned above are becoming more and more prevalent, and there are many practical problems concerning these documents. Additionally, many types of user-generated content, often text, are increasingly the subjects of research works in sentiment analysis or knowledge discovery (O'Connor et al., 2010b; Takeshi et al., 2010; Benson et al., 2011; Ritter et al., 2010).

Moreover, since many of those texts are byproducts of fast-growing modes of public interaction, they are often studied by the social science researchers interested in collective human behavior and its dynamics (Dodds and Danforth, 2008; Kittur et al., 2009; Yardi and Boyd, 2010).

Notice that a broad range of pragmatic questions in this domain can be cast as a form of "response prediction". Consider the case of a lazy blog reader who dislikes wasting time with boring news, and suppose he wishes to read only the most popular blog posts among the hundreds. There are potentially many ways to define the "popularity" of a writing, but one straightforward approach is perhaps to use the readership responses as a proxy for a popularity measure. The reader therefore wishes to find an article which gathered many responses from the readership, or, better yet, *will* gather many responses in the future. Systems which give reasonable predictions of the future response volume would certainly be desirable.

Consider further the situation when the reader wants advice on what *would be* interesting to *him*. This is a question often raised in personalized recommendation systems. At the core of any such system is the predictive system on personal response (whether or not they will find it interesting) to the text. Similar settings arise in many types of document collection where there is a large volume of texts (e.g. news feeds, conference papers, peer reviews on movie or products, tweets).

Not surprisingly, we began to see more works on text-driven response prediction in natural language processing research in recent years. (Joshi et al., 2010) presented text-driven movie revenue prediction tasks. Their model seeks to predict the moviegoers' box-office spending from the reviews written by movie critiques. The underlying assumption is that moviegoers' actions are somehow related to the reviews. (Gerrish and Blei, 2011) examined the prediction of congressional action from the bill texts. The same authors also addressed the citation patterns in scientific paper collections (Gerrish and Blei, 2010). Citations are in a sense a type of readers' response, indicative of interests or agreement toward the target publication. (Yogatama et al., 2011; Dietz et al., 2007) also address the same question. Some types of document-level sentiment prediction tasks seek to predict a binary response ("thumbs up") or a numerical response (such as star rating) from the readership based on the movie review or product description. The question can be cast as a prediction of user reaction caused by the document contents (Pang and Lee, 2008).

1.2 Proposed Prediction Tasks

In this work, we present case studies of text-driven prediction in the domain of American politics. The first part focuses on the political blogosphere, concerning how texts evoke reactions in partisan communities:

- Predicting who (within a blog community) is going to respond to a particular post.
- Predicting how popular a particular post will be among the blog readership.

The second part focuses on the United States Congress, concerning the American legislative system and its members, and how texts sheds light on its operation:

- Predicting bill survival through the Congressional committee system.
- Predicting interest groups' electoral contributions from public microblog messages by members of the U.S. Congress.

Settings and Assumptions

For convenience, we will always refer to the real world reactions (of all varieties) as the "response", or "response variable", in this dissertation. We will call the textual data which is associated with the response the "document" or "actuating document" when it is not clear from the context.

In building these predictive systems, we take a probabilistic approach. Therefore the heart of this dissertation is design and examination of stochastic models of (actuating) documents coupled with the responses they evoke. We view such predictive models as parameterized probability distributions, whose parameters are estimated using data. We train these models (or, learn these parameters) in a supervised learning setting. Therefore the models will learn the statistical patterns between documents and responses from the paired examples in the training data. We evaluate them by estimating their predictive accuracies on a held-out ("out-of-sample") test set. Here are some more general settings we will assume throughout the rest of this work:

- We take it for granted that the two components (documents and their responses) are given, well defined, and presumably interdependent.
- We assume that the detail of the linkage between the two components is not explicit. Even when there are seemingly apparent links, more useful and better generalizable structures may be latent. For example, given a text and a group of people who responded to the text, we do not necessarily know what elements of the text captured the attention of each person. Furthermore, it is possible that some of the respondents reacted to different elements of the text from others, and perhaps for different reasons.
- We presume that annotating all these detailed analyses is expensive, or else impossible.

1.3 Statement of Purpose

Formally, the goals of this dissertation are the following:

In this work, we develop a set of novel statistical models for predicting response actuated by text. We examine four types of response related to American politics in two domains: reader responses and post popularity in the political blogosphere; Congressional committee decisions and electoral campaign finance in the U.S. Congress. For each task, our goals are to construct models which (1) yield high prediction accuracy, and (2) provide a human-understandable data-driven explanation of the underlying response process.

Our chosen tasks deal with important subject matter in contemporary politics. Progress in this area is of high concern to social scientists and political scientists, and also offers novel contributions to statistical text analysis research. We anticipate that models like the ones we introduce will ultimately be useful in applications like recommendation and filtering systems, as well as in social science research that makes use of text as data; development of such applications is outside the scope of this thesis.

1.4 Contributions

In the beginning of this chapter, we motivated response predictions from the point of practical applicability. In this section, we will note our contributions in other contexts.

Statistical analysis of text for extrinsic prediction tasks ("text-driven prediction") is a subject that has been explored before, but it is only recently that the field has started to receive a steady stream of attention from the natural language processing research community. (See Section 1.1.) Text-based analysis of reader *reactions* are dealt with in such areas as sentiment analysis, opinion mining, and most recently, text-driven forecasting. Our response prediction models are novel contributions to these growing fields of natural language processing research.

The essence of text-driven forecasting tasks is the exploitation of textual evidence to predict real world events. In a closely related area, an increasing number of quantitative political scientists advocate "text-as-data" (Laver et al., 2003; Laver and Garry, 2000) approaches to various problems. The key idea in this approach is to treat text as categorical data in the statistical analysis. Similar algorithms are used in both text-driven forecasting and text-as-data approaches to political science, but their emphases are slightly different. Political scientists are more interested in the explanatory power of statistical models (for example, how meaningfully they capture and represent the signals in the text), while text-driven forecasting tends to care more about quantitative predictive performance. As our work is relevant to both disciplines, we maintain both of those goals. We hope our work is a meaningful contribution from both perspectives.

All the prediction tasks we chose here have clear utilities in some useful applications. They also relate to some interesting questions in our society. Our pragmatical contributions are the creative solutions we will offer for each of these tasks. Beyond these immediate merits, we view our dissertation as an attempt at a meaningful synergy between NLP and computational social science. Increasingly, traces of human activities are available online. Such data often takes the form of user generated texts. There seem to be tremendous opportunities for social scientists, but taking advantage of such user generated data is not always straightforward. Not small part in the problem is the technical difficulties in dealing with the large scale NLP. Therefore, there is a large incentive toward the collaboration between the social science and the corpus based NLP research. However, how to form a valid research framework in this context is not always clear.

We believe that text-driven response prediction, and text-driven prediction in general, is one way to unite social science questions and text analysis into a computational framework. In this work, although we use different techniques for different problems, our approaches to the problems follow the same pattern. We first formalize the problem as a response prediction task, simplify the inquiry process as probabilistic model building and inference. We then postulate the stochastic relationships between the texts and the prediction targets. Within this framework, we can explore a variety of hypotheses on the relationships between the text and the response (reaction from the population) through model structure design or feature engineering. To be sure, we do not claim that this is always the best way to approach all the problems in this domain, but we argue that this is one viable solution, which could lead to meaningful results. We hope to demonstrate this point through this work.

1.5 Road Map

We will describe each prediction task in more detail in the rest of the thesis. Each task is largely self-contained, and its structure is essentially parallel: We first describe the background of our domain, then the task and the corpora. All our corpora are closely related to some interesting subjects in current politics. We will discuss the significance of these texts, both in real life and in academic research, then motivate our particular approach and model design choice. We then present the specifics of basic models, some extensions, and experimental results. We conclude each chapter with the discussion on our findings.

In chapter 2 we present the prediction tasks for the blogosphere, and in chapter 3 we examine the models for the U.S. Congress. In the final chapter we present a summary of our contributions and plan for future work.

Chapter 2

The Blogosphere

In this chapter we describe our first two prediction tasks, both concerning response generation in political blogs. The goal of the first task is to reason about which blog posts would evoke responses from which readers. The second task is to examine the popularity (in the form of response volume) of a given post.

We consider our tasks quite practical since blogging, though a relatively new mode of publishing, plays a major role in contemporary political journalism (Wallsten, 2008; Lawrencea et al., 2010; de Zúñiga, 2009). Thousands of people turn to blogs for political information (Eveland and Dylko, 2007). Popular bloggers such as Daily Kos, Andrew Sullivan, or Matthew Yglesias attract a large number of followers, and their articles are read widely around the internet. A mechanism which can forecast how people will react to the posts could serve as a core analytic tool for recommendation, filtering, or browsing systems. Also, community around political blogging is quite an interesting new subject in political science. Political blog sites typically form ideologically homogeneous readership communities, with distinctive attitudes toward various issues (Lawrencea et al., 2010; Karpf, 2008). Data-driven computational modeling such as ours can illustrate issue preference in, and draw contrastive studies among, the blogging communities. They can be easily turned into an automatic means to achieve such profiling, which would be an interesting tool for the blog providers (as a trend analysis) or scholars who wish to study contemporary partisan communities.

In the following sections, we will first define our tasks with more precise scoping (Section 2.1), then present a short discussion on political blogosphere (Section 2.2). We describe our data set in Section 2.3, and our general approach in Section 2.4. We cover each prediction model, including experimental results, in two separate sections (Section 2.5 and Section 2.6) We conclude this chapter with the summary of our contributions and the plan for future work.

The work described here is previously published in (Yano et al., 2009), (Yano and Smith, 2010).

2.1 Task Definition

In this chapter we consider two prediction problems. We have introduced them first in Chapter 1. These are:

- Predicting who (within a blog community) is going to respond to a particular post.
- Predicting how popular a particular post will be among the blog readership.

In both cases, the operative scenario is straightforward; the system is to take a new blog post as an input, and then output the prediction about its would-be response. The systems differ in terms of what aspects of response their prediction is about. While many clues might be useful in predicting response (e.g., the posts author, the time the post appears, the length of the post, etc.), our focus is text in this work, so we define the input to be the textual contents of the blog posts. We ignore non-textual content such as sounds, graphics, or video clips, etc. We will explain more about how we standardize the raw text for the experiments later in the chapter (Section 2.5.3).

For the first task, the system is to output, for each user, the likelihood that she is going to comment on the post. We assume that the set of users (given the blog site) is defined a priori; we expect the system to score all of these users. Since this set of likelihood scores induces the ordering among the users, this prediction task can be casted as a user-ranking task; this is how we evaluate the system.

For the second task, we define the "popularity" of the blog post to be proportional to the volume of comments it receives. Therefore, the output from the system is one scalar value, the prediction of the *volume* of the comments evoked by the input. We primarily use an individual comment as the unit of counting, but additionally consider the count of words as the target output. Further details on the experimental procedures are in Section 2.5.3 (for the first task) and in Section 2.6.3 (for the second task).

We will design and implement the prediction systems, then evaluate them with the real world data. Since we like to contrast among the blog cultures, we will experiment with data from several different blog sites, and fit a separate model for each. In both tasks, we assume the strictly predictive setting; predictors are to yield the output based only on the content of the post's main entry. Any information on any parts of the users' comments are not available at the prediction time. In all our experiments we trained and evaluated our model with the blog corpus prepared by our team (Section 2.3).[1] We will describe this data later in this document. Presently, we will discuss our subject, the political blogosphere.

2.2 Background

Blogging is studied by computer scientists who research large scale networks or online communities (Leskovec et al., 2007a; Agarwal et al., 2008; Leskovec et al., 2007b; Gruhl et al., 2004).

[1] The resource is available for public use in `http://www.ark.cs.cmu.edu/blog-data/`

Among natural language processing researchers, blogs or other user generated texts are particularly important for sentiment analysis or opinion mining (Ounis et al., 2006; Bautin et al., 2008; Chesley et al., 2006; Ku et al., 2006; Godbole et al., 2007; Yano et al., 2010). Blogging is also an important subject in political science (Wallsten, 2008; Karpf, 2008; Mullen and Malouf, 2006; Malouf and Mullen, 2007).

2.2.1 The Political Blogosphere

Blogging has become more prominent as a form of political journalism in the last decade, though it differs from the traditional mainstream media (MSM) in many ways. One difference is that a blog is often more personal and **subjective**, since it is from its inception meant primarily for personal journaling. As noted, much research on subjectivity, sentiments, and opinions is being done on blog text. Meanwhile, objective reporting is unequivocally the core of journalism ethics and standards.[2] In blogging culture, stringent compliance to the journalistic ethic of objectivity does not yet seem to be the social norm.

Blogging seems to uniquely position itself as an ideal thought outlet for concerned citizens (Wallsten, 2008). For many, blogging serves as an online soapbox in grassroots politics. Moreover, blog sites are often used as means of activism, such as solicitations for donations, calls for petitions, or announcements for political rallies and demonstrations. In (Wallsten, 2008), the authors explored types of political blogging activities. Blog sites are often venues for discussion. On many sites, readers are encouraged to express their opinions in the form of comments, thus turning it into an occasion for interactive communication, further nurturing the sense of community among participants.

Aside from the aforementioned subjectivity, another trait in political blogging much differs from MSMs is its unabashed **partisanship** (Lawrencea et al., 2010). Unlike the MSM, many of the popular blogs such as Daily Kos,[3] Think Progress,[4] Hot Air,[5] or Red State,[6] are not only more opinionated, but also unyieldingly partisan. Meanwhile, most of traditional media outfits view an accusation for partiality and imbalance as a serious accusation.[7][8][9] Related, or perhaps a consequence of this partisan culture is an apparent **balkanization** of blog journalism. In their seminal study of the political blogosphere, (Adamic and Glance, 2005), and also (Lawrencea et al., 2010; Karpf, 2008), argued that the political blogosphere is an unrelentingly divided world. They found that blogging communities prefer to form ideologically homogeneous subgroups, rather than reaching out to the other side of political spectrum. Other studies on the blogosphere observe its echo chamber effects (Gilbert et al., 2009), which likely reinforce partisan viewpoints.

[2] `http://asne.org/content.asp?pl=24&sl=171&contentid=171`

[3] `http://dailykos.com/`

[4] `http://thinkprogress.org/`

[5] `http://hotair.com`

[6] `http://www.redstate.com/`

[7] `http://www.onthemedia.org/2011/mar/18/does-npr-have-a-liberal-bias/`

[8] `http://www.npr.org/blogs/ombudsman/2010/06/17/127895293/listeners-hear-same-israeli-palestinian-coverage-differently`

[9] We do not here make the claim that MSM is anyway perfectly non-partisan.In fact, media slant is a subject of many scholastic inquiry, such as in (Gentzkow and Shapiro, 2010).We instead claim that the social norm still consider MSM *ought to* be non-partisan.

As a consequence of this populism, partisanship, and balkanization, the political blogosphere is rather a unique microcosm of contemporary community politics. In this sense, the political blogosphere presents itself as an unprecedented research opportunity; what can we find in this huge quantity of spontaneous, near-real-time trace of political thought and behavior, which likely mirrors various political subcultures in real life?

2.2.2 Why Blog? Why Predict Comments?

Earlier we motivated the utility in text-driven prediction using blog recommendation as an example. Aside from such practical utility, we view predictive modeling of reactions as one way to investigate these political communities. Feedback from the engaged readers is an integral part of cultural identity. Moreover, since blog posts and user comments form a stimulus-response relationship, comments define the community by shaping the interactive patterns between the texts (blog posts) and reader response (comments). Later we will see that the statistical trends discovered by the model differ across the partisan cultures. Depending on the ideological orientation of a community, certain issues stimulate more response, while others are ignored by the readers.

Another scholastic motivation is to address the question of how user-generated texts (such as comments) can be made useful. Spontaneous user texts are often noisy and difficult to deal with by conventional NLP assumptions. Although the influx of social media data in recent years has started to incentivize more works on user texts, the research potential in this area has yet to be fully explored. Comment contents in particular are usually among the most ill-tempered data, and are often omitted even in the works concerning blogs (Yano and Smith, 2010). Nonetheless, often the most substantial amount of blog contents are indeed the reader comments. (Among the blog data we collected, this is certainly the case for most of the sites. See Table 2.1). Also, comments tend to reflect more personal voice, which makes them a desirable subject for such tasks as sentiment analysis or opinion mining. In their pioneering work, Mishne and Glance (Mishne and Glance, 2006) showed the value of comments in characterizing the social repercussions of a post, including popularity and controversy.

Part of the motivation for our work is to contribute to the development of this important trend in text analysis by making a clear case of comments' usefulness. We like to note that since our initial publication, we have seen an increase in the number of research on comment and comment like texts. Our works on blog comments are one of the earliest computational exploration on the subject, and have been cited by some of the notable works on comment texts in recent years (Park et al., 2011; Filippova and Hall, 2011; Mukherjee and Liu, 2012; Potthast et al., 2012; Ko et al., 2011; Fang et al., 2012), as well as the works in the political sentiment detection and opinion mining in the blogosphere. (Balasubramanyan et al., 2012, 2011; Das et al., 2009). The political news recommendation system based on comment analysis presented in (Park et al., 2011) is precisely the type of intelligent software applications which we envision the current work to be useful.

	MY	RWN	CB	RS	DK
Time span (from 11/11/07)	–8/2/08	–10/10/08	–8/25/08	–6/26/08	–4/9/08
# training posts	1607	1052	1080	2045	2146
# words (total)	110,788	194,948	183,635	321,699	221,820
(on average per post)	(68)	(185)	(170)	(157)	(103)
# comments	56,507	34,734	34,244	59,687	425,494
(on average per post)	(35)	(33)	(31)	(29)	(198)
(commenters, on average)	(24)	(13)	(24)	(14)	(93)
# words in comments (total)	2,287,843	1,073,726	1,411,363	1,675,098	8,359,456
(on average per post)	(1423)	(1020)	(1306)	(819)	(3895)
(on average per comment)	(41)	(31)	(41)	(27)	(20)
Post vocabulary size	6,659	9,707	7,579	12,282	10,179
Comment vocabulary size	33,350	22,024	24,702	25,473	58,591
Size of user pool	7,341	963	5,059	2,789	16,849
# test posts	183	113	121	231	240

Table 2.1: Details of the blog data used in this chapter. "MY" = Matthew Yglesias, "RWN" = Right Wing News, "CB" = The Carpetbagger Report, "RS" = Red State, "DK" = Dairy Kos.

2.3 Data: Political Blog Corpus

To support our data driven approach in political blogs, we have collected blog posts and comments from 40 blog sites focusing on American politics during the period from November 2007 to October 2008, contemporaneous with the United States Presidential elections. The discussions on these blogs focus on American politics, and many themes appear: the Democratic and Republican candidates, speculation about the results of various state contests, and various aspects of international and (more commonly) domestic politics. The sites were selected to have a variety of political leanings. From this pool we chose five blogs which accumulated a large number of posts during the period and use them to experiment with our prediction models: The Carpetbagger Report (CB),[10] Daily Kos (DK), Matthew Yglesias (MY),[11] Red State (RS), and Right Wing News (RWN).[12] CB and MY ceased as independent bloggers in August 2008.[13]

Because our focus in this work is on blog posts and their comments, we discard posts on which no one commented within six days. We also remove posts with too few words: specifically, we retain a post only if it has at least five words in the main entry, and at least five words in the comment section. All posts are represented as text only (images, hyperlinks, and other non-text contents are ignored). To standardize the texts, we remove from the text 670 commonly used stop words, non-alphabet symbols including punctuation marks, and strings consisting of only symbols and digits. We also discard infrequent words from our dataset: for each word in a post's main entry, we kept it only if it appears at least one more time in some main entry. We apply the same word pruning to the comment section as well. In addition, each users handle is replaced with a unique integer.

[10] http://www.thecarpetbaggerreport.com

[11] http://matthewyglesias.theatlantic.com

[12] http://www.rightwingnews.com

[13] The authors of those blogs now write for larger online media, CB for Washington Monthly, and MY for Think Progress, and The Atlantic.

See Table 2.1 for the detail of this data. The data is available from `http://www.ark.cs.cmu.edu/blog-data/`. Since its release in 2010, the data have been used in several other publications to date, such as (Balasubramanyan et al., 2011; Ahmed and Xing, 2010; Eisenstein, 2013; Balasubramanyan et al., 2012).

Qualitative Properties of Blogs

We believe that readers' reactions to blog posts are an integral part of blogging activity. Often comments are much more substantial and informative than the post. While circumspective articles limit themselves to allusions or oblique references, readers' comments may point to heart of the matter more boldly. Opinions are expressed more blatantly in comments. Comments may help a human (or automated) reader to understand the post more clearly when the main text is too terse, stylized, or technical.

Although the main entry and its comments are certainly related and at least partially address similar topics, they are markedly different in several ways. First of all, their vocabulary is noticeably different. Comments are more casual, conversational, and full of jargon. They are less carefully edited and therefore contain more misspellings and typographical errors. There is more diversity among comments than within the single-author post, both in style of writing and in what commenters like to talk about. Depending on the subjects covered in a blog post, different types of people are inspired to respond.

Blog *sites* are also quite different from each other. Their language, discussion topics, and collective political orientations vary greatly. Their volumes also vary; multi-author sites (such as DK, RS) may consistently produce over twenty posts per day, while single-author sites (such as MY, CB) may have a day with only one post. Single author sites also tend to have a much smaller vocabulary and range of interests. The sites are also culturally different in commenting styles; some sites are full of short interjections, while others have longer, more analytical comments. On some sites, users appear to be close-knit, while others have high turnover.

In the next section, we describe how we apply topic models to political blogs, and how these probabilistic models are used to make predictions.

2.4 Proposed Approach: Probabilistic Topic Model

In this chapter we explore the **generative approach**. This means that we will first design a stochastic model over the generative process of the data (the so called "generative story"), and then perform the prediction task as posterior inference over the query (or prediction target) variables.

The procedure seems a bit roundabout compared to the discriminative approach, which seeks to directly optimize an objective criterion. The generative approach, however, has a few advantages which are particularly desirable for our task. One is its expressiveness; it is relatively straightforward to encode hypotheses or insights into computational frameworks with the generative approach. Another is the generative approach's flexibility; we can often augment basic models with arbitrary random variables, while still facilitate fairly principled learning algorithms using standard techniques. We will see both of these advantages in action later in our model description section (Section 2.5.1).

The heart of the generative approach is the design of the generative story. Recall that in this task we prefer a model which not only performs well on the prediction task, but also provides insights as to *why* some blog posts inspire reactions. A natural generalization is to consider how the *topic* (or topics) of a post influence commenting behavior. We therefore use a **topic model** to describe the data generation process. We will design our own flavor of a topic model rather than employing the existing varieties. We start with an existing model, Latent Dirichlet Allocation (Blei et al., 2003) , and gradually augment this base model to cater to the unique aspects of blog texts.

Latent Dirichlet Allocation (LDA) is a generative probabilistic model of text similar in spirit to the unigram language model, but goes beyond it by positing a hidden topic distribution, drawn distinctly for each document, that defines a document-level mixture model. The topics are unknown in advance, and are defined only by their separate word distributions, which are discovered through probabilistic inference from data. Like many other techniques that infer topics as measures over the vocabulary, LDA often finds very intuitive topics. It also can be extended to model other variables as well as texts (Griffiths and Steyvers, 2004; Steyvers and Griffiths, 2007).[14]

In the next section we present a brief technical review of LDA, emphasizing the aspects most relevant to our current task. We build up our own model in the following section.

2.4.1 Technical Review of Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA), a type of latent topic model, is formally an admixture model over a set of discrete random variables. The model has been applied to variety of tasks in natural language processing, such as topic clustering, corpus exploration, or as a means of dimensionality reduction. For our purpose, we view the model as a Bayesian extension to the class-mixture language model, or the 0th order Markov model over the text. Connections between LDA and mixture models have been drawn before in (Blei et al., 2003), (Heinrich, 2008), and a few others. We present the discussion here to emphasize the modularity of the generative model construct, as we later extend the LDA for our particular purpose. The discussions in (Blei et al., 2003) and (Heinrich, 2008) include more thorough analysis.

Let's first consider a simpler mixture model over the text. Let $\boldsymbol{w}_d$ denote a document d represented as a bag of unigrams, and z_d as the document' thematic class, which has an associated

[14] LDA is in fact a formalism applicable to any type of categorical data. Its use is by no means limited to textual data, nor to natural language research. We explain the algorithm assuming text analysis as a main application domain for the sake of simplicity.

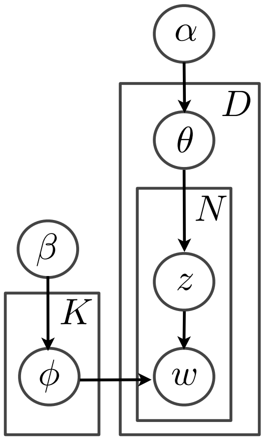

Figure 2.1: Plate notation for Latent Dirichlet allocation

(class conditional) unigram language model. The joint distribution of this model is the following:

$$p(\boldsymbol{w}_d, z_d) = p_\theta(z_d) \cdot \prod_{i=1}^{N_d} p(w_{d,i} \mid z_d)$$

Lets assume that the texts are represented as multinomial distribution(s) over the finite vocabulary, and reiterate the above function as the generative story:

1. Choose a class label z_d according to the label distribution θ.
2. For i from 1 to N_d (the length of the document):
   (a) Choose a word $w_{d,i}$ according to the class's word distribution Multinomial(z_d)

Assuming multinomial distribution, the parameters for this model can be estimated via maximum likelihood estimation when all the document-class labels are observed. When the labels are not observed, various flavor of expectation maximization (EM) algorithm can be used (Nigam et al., 1999). Note that this is the type of generative model which Naive Bayes classification algorithm is derived from. Naive Bayes has been studied extensively for both supervised and unsupervised document classification tasks.

Latent Dirichlet Allocation augments the simple mixture model with three additional generative hypothesis:

- Each word can be associated with different thematic classes. Thematic classes are the "topics".
- A thematic class is itself a random variable drawn from a document specific multinomial distribution.

- The document level multinomial distribution is also a random variable drawn from a corpus-specific Dirichlet distribution.

Those additional assumptions lead to different generative story:

1. For each topic k from 1 to K:
   (a) Choose a distribution ϕ_k over words according to a symmetric Dirichlet distribution parameterized by β.
2. For each document d from 1 to D:
   (a) Choose a distribution θ_d over topics according to a symmetric Dirichlet distribution parameterized by α.
   (b) For i from 1 to N_d (the length of the document):
      i. Choose a topic $z_{d,i}$ according to the topic distribution θ_d.
      ii. Choose a word $w_{d,i}$ according to the word distribution $\phi_{z_{d,i}}$.

Above we treat θ_d, the multinomial parameters for the distribution over the topics, as another set of random variables drawn from the Dirichlet distribution. This is often called a Bayesian approach. The corresponding joint probability (for one document) distribution is the following:

$$p(\boldsymbol{w}_d, \boldsymbol{z}_d, \theta_d) = p_\alpha(\theta_d) \cdot \prod_{i=1}^{N_d} p_\beta(w_{d,i}|\phi_{z_{d,i}}) \cdot p(z_{d,i}|\theta_d)$$

Often plate notation, a type of diagram, is used to express compound distributions such as LDA. We add this alternative representation in Figure 2.1. Note that these three representations, mathematical expression, generative description, and plate notation, all describe the same stochastic system. For the thorough discussion, see (Blei and Lafferty, 2009; Steyvers and Griffiths, 2007; Heinrich, 2008).

2.4.2 Notes on Inference and Parameter Estimation

Latent topic models like LDA can be used for a variety of tasks, including predictions such as classification (predicting θ_d or $\boldsymbol{z}_d$ given a new document $\boldsymbol{w}_d$), or document modeling (predicting an unseen part of $\boldsymbol{w}_d$ from the observed part of $\boldsymbol{w}_d$).[15] To solve such prediction problems, it is necessary to find the posterior distributions over the query variables. An often taken strategy is to estimate the model parameters ($\boldsymbol{\phi}$, and sometimes also α and β) through empirical Bayes methods (Gelman et al., 2004), then run inference over the query variables.

The central question in model parameter estimation for Bayesian models such as the above is

[15] The latter is sometimes called document completion task, and often used as an evaluation for LDA-like latent variable models for text.

posterior inference of the latent variables. In this model two sets of random variables, topic distributions $\boldsymbol{\Theta}$, and topic assignments $\boldsymbol{Z}$, are latent variables. They are usually assumed unobservable (therefore unannotated in the data) even during training time. One popular approach is aforementioned expectation maximization (EM) technique and similar iterative optimization algorithms. They typically require inference over the latent variables during the E-step. In original LDA paper the authors used Variational EM, where the mean-field approximation method is used for the E-step. Another variation of EM method using MCMC sampling is introduced in (Griffiths and Steyvers, 2004).

In our experiments (Section 2.5.3) we choose a sampling approach for model training, with Gibbs sampling (a type of MCMC sampling) for the E-step. The idea is first introduced in (Griffiths and Steyvers, 2004), but the authors devised the training algorithm only for the basic LDA. Although the models we introduce in this chapter are extensions of LDA, each has much different objective functions. Naturally, the quantities to compute during the optimization are much different. In the subsequent sections, we will provide the necessary details, such as the analytical form of the posterior distribution over the latent variables, to reconstruct our algorithm given knowledge of the basic EM algorithm for LDA, rather than spelling out the algorithms step-by-step. Training algorithms for LDA (and similar Bayesian models) have been explained in the numerous journal papers, tutorial, and text books in the past. For a detailed description of sampling algorithms, see (Heinrich, 2008) or (Blei and Lafferty, 2009).

2.5 Predicting Reader Response

In this section we discuss the first prediction task, predicting who (within a blog community) is going to respond to a particular post. We employ the generative approach; we first design the generative story, then derive the prediction procedure as inference over the query variables. We start with a standard latent topic model (LDA) as a basic building block. A topic model embodies the idea that the text generation is driven by a set of (unobserved) thematic concepts, and each document is defined by a subset of those concepts. This assumption is fairly reasonable with political blogs since discussions in politics are issue-oriented in nature. We do not apply LDA as a plug-in solution to our task, however. Rather, we *extend* the concept, making a new generative model to tailor to the particulars of our data and prediction goals.

Later in the experimental section we adopt a typical NLP “train-and-test” strategy that learns the model parameters on a training dataset (consisting of a collection of blog posts and their commenters and comments), and then considers an unseen test dataset from a later time period. We present the quantitative results on user prediction tasks, as well as the qualitative analysis of what discovered through the training.

Variable	Description
D	Total number of documents
θ_d	Distribution over the topics for document (blog post) d
α	Dirichlet hyper-parameters on θ_d
K	Total number of topics
ϕ_k	Distribution over words conditioned on topic k
β	Dirichlet hyper-parameters on ϕ_k
$z_{d,i}$	The (latent) random variable for topic at position i in d
$w_{d,i}$	Random variable for the word at position i in d
ϕ'_k	Distribution over words (in comment) conditioned on topic k
ψ_k	Distribution over user ids conditioned on topic k
β'	Dirichlet hyper-parameters on ϕ'_k
γ	Dirichlet hyper-parameters on ψ'_k
$z'_{d,j}$	The (latent) random variable for topic at position j in comment of d
$w'_{d,j}$	Random variable for the word at position j in comment of d
$u_{d,j}$	Random variable for the word at position j in comment of d

Table 2.2: Notations for the generative models. The ones above the center line are also used in the plain LDA model.

2.5.1 Model Specification

Earlier we discussed the qualitative difference between the post and comment sections (Section 2.2). The main post and its comments are certainly (at least thematically) related. However, we observed that they are markedly different in its style in a number of ways. We therefore assume here that for each post, the comment section shares the same set of topics with the post's main entry, but uses the languages much different from the main section to express these topics. We will make this change by bestowing an additional set of conditional distributions for comment side.

Here are some hypothesis we seek to encode into our model:

- Comments certainly talk about the topics similar to the post;
- Comments are related to the posts topic, but have distinct style;
- Comments often consist of a sequence of individual comments, each usually authored by different readers.

We first create a new generative story with these insights in the following section. As LDA was to the simpler mixture model, our model can be understood as the modular extension to the basic LDA model. We then turn our stochastic model for the user prediction tasks.

Generative story

As in LDA, our model on blogs postulates a set of latent topic variables (θ_d) for each document d, and each topic k has a corresponding multinomial distribution ϕ_k over the vocabulary. In

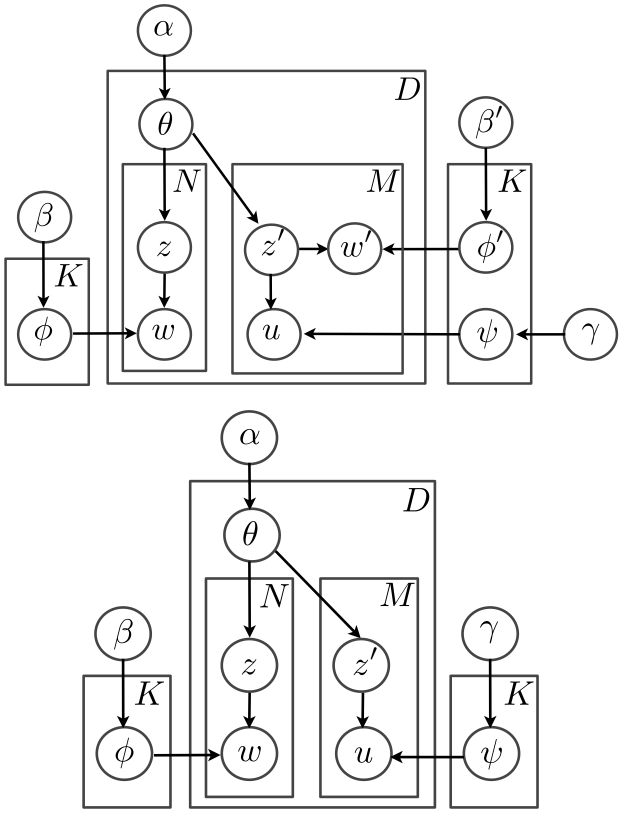

Figure 2.2: Top: CommentLDA. In training, w, u, and w' are observed. D is the number of blog posts, and N and M are the word counts in the post and the all of its comments, respectively. Here we "count by verbosity". Bottom: LinkLDA (Erosheva et al., 2004), with variables reassigned.

addition, the model generates the comment contents from a multinomial distribution ϕ'_k, and a bag of users who respond to the post (represented as their user handles), from a distribution γ_k, both of them conditioned on the topic. The arrangement is to capture the differences in language style between posts and comments. In the experiment section, we call this model **CommentLDA**. The complete generative story of this model is the following. For each blog post d from 1 to D:

1. Choose a distribution θ_d over topics according to a symmetric Dirichlet distribution parameterized by α.
2. For i from 1 to N_d (the length of the post):
   (a) Choose a topic $z_{d,i}$ according to the topic distribution θ_d.
   (b) Choose a word $w_{d,i}$ according to the post word distribution $\phi_{z_{d,i}}$.
3. For j from 1 to M_d (the length of the comments on the post, in words):

(a) Choose a topic $z'_{d,j}$ according to the topic distribution θ_d.

(b) Choose an author $u_{d,j}$ according to the commenter distribution $\psi_{z'_{d,j}}$.

(c) Choose a word $w'_{d,j}$ according to the comment word distribution $\phi'_{z'_{d,j}}$.

The corresponding plate notation is shown in Figure 2.2. Note that the model is identical to LDA until step 2. The joint distribution of the above generative story is below (for one document). Additional terms on the right collectively represent the third component of the generative story, which account for the generation of the comment contents:

$$p(\boldsymbol{w}_d, \boldsymbol{w'}_d, \boldsymbol{z}_d, \boldsymbol{z'}_d, \boldsymbol{u}_d, \theta_d) = p_\alpha(\theta_d) \cdot \prod_i^{N_d} p_\beta(w_{d,i} \mid \phi_{z_{d,i}}) \cdot p(z_{d,i} \mid \theta_d)$$
$$\cdot \prod_j^{M_d} p_{\beta'}(w'_{d,j} \mid \phi'_{z'_{d,j}}) \cdot p_\gamma(u_{d,j} \mid \psi_{z'_{d,j}}) \cdot p(z'_{d,j} \mid \theta_d)$$

In the plate diagram, the additional chamber on the left side represent this part.

One way to look at this model is that now the latent thematic concept, or topic k, is described by three different types of representation:

- A multinomial distribution ϕ_k over post words;
- A multinomial distribution ϕ'_k over comment words; and
- A multinomial distribution ψ_k over blog commenters who might react to posts on the topic.

Also, in this model, the topic distribution, θ_d, is all that determines the text content of the post, comments, and which users will respond to the post. In other words, post text, comment text, and commenter distributions are all interdependent through the (latent) topic distribution θ_d.

Prediction

Given the trained model and a new blog post, we derive the prediction on the commenting users through a series of posterior inferences. For a new post d, we first infer its topic distribution θ_d; since we do not observe any part of the comment, we estimate this posterior from the words in the post $\boldsymbol{w}_d$ alone; Once the document level topic distribution is estimated, we can infer the distribution over the users in the following way:

$$\begin{aligned} p(u \mid \boldsymbol{w}_d, \boldsymbol{\psi}, \gamma, \alpha) &= \sum_{k=1}^{K} p(u \mid k, \boldsymbol{\psi}; \gamma) \cdot p(k \mid \boldsymbol{w}_d; \alpha) \\ &= \sum_{k=1}^{K} \psi_{k,u} \cdot \hat{\theta}_{d,k} \end{aligned} \tag{2.1}$$

To obtain $\hat{\theta}_d$, we run one round of Gibbs sampling given the $\boldsymbol{w}_d$ (while fixing all the model parameters $\boldsymbol{\phi}$, $\boldsymbol{\phi'}$, $\boldsymbol{\psi}$, α, β,, γ) then renormalize the the sample counts:

$$\theta_{d,k} = \frac{\mathrm{C}(k; \boldsymbol{z}_d) + \alpha_k}{\sum_{k'=1}^{K} \mathrm{C}(k'; \boldsymbol{z}_d) + \alpha_{k'}}$$

Where $\mathrm{C}(k; \boldsymbol{z}_d)$ is the count of topic k within the sample set $\boldsymbol{z}_d$. Sampling of $\boldsymbol{z}_d$ is done in the same way as the sampling during the EM procedure, which we review in the next section.

2.5.2 Notes on Inference and Estimation

We train our model using standard Bayesian estimation. Specifically, we fix $\alpha = 0.1$, $\beta = 0.1$, and learn the values of word distributions $\boldsymbol{\phi}$ and $\boldsymbol{\phi'}$ and user distribution $\boldsymbol{\psi}$ by maximizing the likelihood of the training data: $p(\boldsymbol{W}, \boldsymbol{W'}, \boldsymbol{U} \mid \alpha, \beta, \gamma, \boldsymbol{\phi}, \boldsymbol{\phi'}, \boldsymbol{\psi})$. Marginalized above are the latent variables, $\boldsymbol{\Theta}$, $\boldsymbol{Z}$, and $\boldsymbol{Z'}$. Note that if these latent variables are all given, the model parameters can be computed in closed form. For example, the distribution over the words (in the post) conditioned on the topic k, ϕ_k is:

$$\phi_{t,k} = \frac{\mathrm{C}(t; \boldsymbol{z}_k) + \beta_t}{\sum_{t'=1}^{T} \mathrm{C}(t'; \boldsymbol{z}_k) + \beta_{t'}} \tag{2.2}$$

Where $\mathrm{C}(t; \boldsymbol{z}_k)$ is the count of the tokens in the document assigned to the term t and topic k. The above equation follows directly from the standard inference procedure in the Bayesian network. The other model parameters, $\boldsymbol{\phi'}$ and $\boldsymbol{\psi}$, can be computed similarly from the sample statistics. Since the values for these latent variables are unknown, we approximate them using Gibbs sampling.

To build a Gibbs sampler, the univariate conditionals (or full conditionals) $p(z_{d,i} = k \mid \boldsymbol{z}^{\neg d,i}, \boldsymbol{w}, \alpha, \beta)$ must be found. In particular, here we use collapsed Gibbs sampling (Casella and Robert, 2004), forming the conditional distribution over the latent topic assignment $z_{d,i}$ while marginalizing out the document level topic assignments $\boldsymbol{\theta}_d$:

$$p(z_{d,i} = k \mid \boldsymbol{z}^{\neg d,i}, w_{d,i} = t, \alpha, \beta) = \frac{\mathrm{C}(k; \boldsymbol{z}_d^{\neg d,i}) + \alpha_k}{\sum_{k'=1}^{K} \mathrm{C}(k'; \boldsymbol{z}_d^{\neg d,i}) + \alpha'_k} \times \frac{\mathrm{C}(k, t; \boldsymbol{z}_{\cdot}^{\neg d,i}) + \beta_t}{\sum_{t'=1}^{T} \mathrm{C}(k, t'; \boldsymbol{z}_{\cdot}^{\neg d,i}) + \beta_{t'}}$$

Where $\mathrm{C}(k; \boldsymbol{z}_d^{\neg d,i})$ is the count of the tokens in the document d assigned to the topic k, excluding the token at the ith position. Similarly, $\mathrm{C}(k, t; \boldsymbol{z}_{\cdot}^{\neg d,i})$ is the count of the tokens assigned to the topic k and term t excluding the token at the ith position. When sampling the latent topic assignment in the comment side, $z'_{d,j}$, the derived conditional distribution include the influence from the co-occurrence statistics in the comment words and the commenting users:

$$\begin{aligned} p(z'_{d,j} = k \mid \boldsymbol{z}^{\neg d,j}, w_{d,j} = t, u_{d,j} = v, \alpha, \beta', \gamma) &= \frac{\mathrm{C}(k; \boldsymbol{z}_d^{\neg d,j}) + \alpha_k}{\sum_{k'=1}^{K} \mathrm{C}(k'; \boldsymbol{z}_d^{\neg d,j}) + \alpha'_k} \\ &\times \frac{\mathrm{C}(k, t; \boldsymbol{z}_\cdot^{\neg d,j}) + \beta'_t}{\sum_{t'=1}^{T} \mathrm{C}(k, t'; \boldsymbol{z}_\cdot^{\neg d,j}) + \beta'_{t'}} \cdot \frac{\mathrm{C}(k, v; \boldsymbol{z}_\cdot^{\neg d,j}) + \gamma_v}{\sum_{v=1}^{V} \mathrm{C}(k, v'; \boldsymbol{z}_\cdot^{\neg d,j}) + \gamma_{v'}} \end{aligned}$$

Both univariate conditionals can be derived using standard techniques, exploiting the facts that both prior distributions are Dirichlet distributions, which is the conjugate prior for the multinomial.[16] Note also that the count of the latent assignments are the sufficient statistics to estimate the model parameters.

2.5.3 Model Variations

We experiment with several variations of the model.

On (not) weighting comment contents

What if we assume that the participants' identities explain away everything about the comment? In other words, what if the comment *content* is utterly random given the user? Or if blog commenters always say the same things to any post, no matter what the topics are? Then it would make more sense to omit the comment contents entirely from the model. This hypothesis suggest the following model:

$$\begin{aligned} p(\boldsymbol{w}_d, \boldsymbol{z}_d, \boldsymbol{z'}_d, \boldsymbol{u}_d, \theta_d) = p(\theta_d; \alpha) \cdot \prod_{i=1}^{N_d} p(w_{d,i} \mid \phi_{z_{d,i}}, \beta) \cdot p(z_{d,i} \mid \theta_d) \\ \cdot \prod_{j=1}^{M_d} p(u_{d,j}; \psi_{z'_{d,j}}, \gamma) \cdot p(z'_{d,j} | \theta_d) \end{aligned}$$

Analogous models are introduced in (Erosheva et al., 2004), although the variables are given different meanings in their model.[17] In our experiment section, we call this model **LinkLDA**. LinkLDA models which users are likely to respond to a post, but it does not model what they will write. The graphical model is depicted in Figure 2.2 (below). Similar models were applied to different tasks in natural language processing research, such as relation extraction or polarity classification, with competitive results (Ritter et al., 2010; Paul et al., 2010). We will see later that for some blogs we can achieve better prediction performance if comment contents are utterly discounted.

On how to count users

In the above generative story, we designed the model so that a user handle is generated at each word position. The choice is rather arbitrary, and a few alternatives are possible.

[16] See (Heinrich, 2008) for more detailed discussion on the issue.

[17] Instead of blog commenters, they modeled citations.

As described, CommentLDA associates each comment word token with an independent author. In both LinkLDA and CommentLDA, this "**counting by verbosity**" will force ψ to give higher probability to users who write longer comments with more words. We consider two alternative ways to count comments, applicable to both LinkLDA and CommentLDA. These both involve a change to step 3 in the generative process.

Counting by response (replaces step 3): For j from 1 to U_i (the number of users who respond to the post): (a) and (b) as before. (c) *(CommentLDA only)* For ℓ from 1 to $\ell_{i,j}$ (the number of words in u_j's comments), choose w'_ℓ according to the topic's comment word distribution $\phi'_{z'_j}$. This model collapses all comments by a user into a single bag of words on a single topic. The counting-by-response models are deficient, since they assume each user will only be chosen once per blog post, though they permit the same user to be chosen repeatedly.

Counting by comments (replaces step 3): For j from 1 to C_i (the number of comments on the post): (a) and (b) as before. (c) *(CommentLDA only)* For ℓ from 1 to $\ell_{i,j}$ (the number of words in comment j), choose w'_ℓ according to the topic's comment word distribution $\phi'_{z'_j}$. Intuitively, each comment has a topic, a user, and a bag of words.

The three variations—counting users by verbosity, response, or comments—correspond to different ways of thinking about topics in political blog discourse and user participations. Counting by verbosity will let garrulous users define the topics. Counting by response is more democratic, letting every user who responds to a blog post get an equal vote in determining what the post is about, no matter how much that user says. Counting by comments gives more say to users who engage in the conversation *repeatedly*.

2.5.4 Experimental Results

For each of the five political blogs in our corpus, we trained the three variations each of LinkLDA and CommentLDA. Model parameters $\boldsymbol{\phi}$, $\boldsymbol{\psi}$, and (in CommentLDA) $\boldsymbol{\phi'}$ were learned by maximizing likelihood, with Gibbs sampling for inference, as described in Section 2.4.2. The number of topics K was fixed at 15. We then estimated users' comment likelihood for each blog post d in the test set as follows: First, we removed the comment section (both the words and the authorship information) from the data. Then, we ran a Gibbs sampler with the partial data, fixing the model parameters to their learned values and the words in the post to their observed values. This gives a posterior topic mixture for each post (θ in the above equations).[18] Upon fixing the topic mixture, we then computed the posterior distribution over the users as in Eq. 2.1.

Note that these posteriors have different meanings for different variations:

- When counting by verbosity, the value is the probability that the next (or any) comment word will be generated by the user, given the blog post.

[18] For a few cases we checked the stability of the sampler and found results varied by less than 1% precision across ten runs.

- When counting by response, the value is the probability that the user will respond *at all*, given the blog post. (Intuitively, this approach best matches the task at hand.)
- When counting by comments, the value is the probability that the next (or any) comment will be generated by the user, given the blog post.

Recall also that both LinkLDA and CommentLDA embody some assumptions about the readership populations. The comparative results on the model performance would to some extent support or refute the various assumptions that we make for each site. As we see below, models perform differently for different blog sites.

Evaluation Setup

In our experiments, we apply our model's outputs to a *user ranking* task. The predictive output from our models is the posterior probability over the known users. Since this is a multinomial distribution, the likelihood of each user commenting on the given post is simply the value of the corresponding element in the probability vector. Since this includes all the users, this set of likelihood scores induces a natural ordering among them.

To measure performance, we compute "Precision at top n" between the predictive ranking from the model and the actual commenters in the test set, for various values of n. Precision at n is a method often used for relevance scoring evaluation in the field of Information Retrieval. It is applicable when comparing a set of gold standards (a set of relevant documents, or more generally, a set of "positive" class examples) to a proposed ranked list.[19] Since this induces the ranking quality score for each post, we report the macro-averaged precision across all the posts in the test set (Table 2.3). Although there are several other evaluation metrics applicable here (such as Mean Average Precision (MAP) or Mean Reciprocal Rank(MRR)), in our experiments, we focus on the top ranked users. MAP and similar rank-to-set evaluation metrics which consider all positions in the ranking are desirable if ordering is important function of the proposed systems. For an application like blog recommendation, correctly ordering the entire set of users is perhaps less important than identifying highly interested users with high probability.

In addition to the macro-average of the precisions at the various n, we report the macro-average of the precisions at the *break-even point*. This metric is sometime called *R-precision*. This is the precision score at the special value of n where recall equals precision. Since this scoring method implicitly assumes that the *size* of the correct set is known in advance, we label this as "Oracle". R-precision is shown to be highly correlated to the MAP score, though it represents only a single point (Manning et al., 2008) in the precision-recall spectrum. In all cases we use the same temporal training-test splits, which we described in Section 2.3; that section also describes the detail of our text processing and standardization.

Baselines

As a simple baseline method we implemented a post-independent prediction that ranks users by their comment frequency. Since blogs often have a "core constituency" of users who post frequently, this is a strong baseline. We also compared to a Naïve Bayes classifier. We built one

[19] See chapter 8 of (Manning et al., 2008) for more detail.

classifier for each known user with word counts in the post's main entry as features. Since Naïve Bayes classifiers give probabilistic scores for each class (in our case, each user), we simply order the users by their likelihood for each test blog post.

Results

We report in Tabel 2.3 the performance of our predictions at various cut-offs (n). The oracle column is the precision where it is equal to the recall, equivalent to the situation when the true number of commenters is known. (The performance of random guessing is well below 1% for all sites at the cut-off points shown.) "Freq." and "NB" refer to our baseline methods. "Link" refers to LinkLDA and "Com" to CommentLDA. The suffixes denote the counting methods: verbosity ("-v"), response ("-r"), and comments ("-c"). Recall that we considered only the comments by the users seen at least once in the training set, so perfect precision, as well as recall, is impossible when new users comment on a post; the *Max* row shows the maximum performance possible given the set of commenters recognizable from the training data.

Our results suggest that, if asked to guess 5 people who would comment on a new post given some site history, we will get 25–37% of them right, depending on the site, given the content of a new post. We achieved some improvement over both the baseline and Naïve Bayes for some cut-offs on three of the five sites, though the gains were very small for RS and CB.

LinkLDA usually works slightly better than CommentLDA, except for MY, where CommentLDA is stronger, and RS, where CommentLDA is extremely poor. Differences in commenting style are likely to blame: MY has relatively long comments in comparison to RS, as well as DK. MY is the only site where CommentLDA variations consistently outperformed LinkLDA variations, as well as Naïve Bayes classifiers. This suggests that sites with more terse comments may be too sparse to support a rich model like CommentLDA.

In general, counting by response works best, though counting by comments is a close rival in some cases. We observe that counting by response tends to help LinkLDA, which is ignorant of the word contents of the comment, more than it helps CommentLDA. Varying the counting method can bring as much as 10% performance gain.

Each of the models we have tested makes different assumptions about the behavior of commenters. Our results suggest that commenters on different sites behave differently, so that the same modeling assumptions cannot be made universally.

2.5.5 Descriptive Aspects of the Models

Aside from prediction tasks such as above, the model parameters by themselves can be informative. ϕ defines which words are likely to occur in the post body for a given topic. ϕ' tells which words are likely to appear in the collective response to a particular topic. Similarity or divergence of the two distributions can tell us about differences in language used by bloggers and their readers in the communities. γ expresses users' topic preferences. A pair or group of participants may be seen as "like-minded" if they have similar topic preferences (perhaps useful

MY

	Freq.	NB	Link-v	Link-r	Link-c	Com-v	Com-r	Com-c	max
n = 5	23.93	25.13	20.10	26.77	25.13	22.84	**27.54**	22.40	*94.75*
n = 10	18.68	19.28	14.04	18.63	18.85	17.15	**20.54**	18.50	*89.89*
n = 20	14.20	14.20	11.17	14.64	14.61	12.75	14.61	**14.83**	*73.63*
n = 30	11.65	11.63	9.23	12.47	11.91	10.69	12.45	**12.56**	*58.76*
Oracle	13.81	13.54	11.32	14.03	13.84	12.77	**14.35**	14.20	*92.60*

RS

	Freq.	NB	Link-v	Link-r	Link-c	Com-v	Com-r	Com-c	Max
n = 5	**25.45**	22.07	14.63	25.19	24.50	14.97	15.93	17.57	*80.77*
n = 10	16.75	16.01	11.9	**16.92**	16.45	10.51	11.42	12.46	*62.98*
n = 20	11.42	11.60	9.13	**12.14**	11.49	8.46	8.37	8.85	*40.95*
n = 30	9.62	9.76	7.76	**9.82**	9.32	7.37	6.89	7.34	*29.03*
Oracle	17.15	16.50	11.38	**17.98**	16.76	11.30	10.97	12.14	*91.86*

CB

	Freq.	NB	Link-v	Link-r	Link-c	Com-v	Com-r	Com-c	Max
n = 5	33.38	36.36	32.06	**37.02**	36.03	32.39	35.53	33.71	*99.66*
n = 10	28.84	31.15	26.11	31.65	**32.06**	26.36	29.33	29.25	*98.34*
n = 20	24.17	25.08	19.79	24.62	**25.28**	20.95	24.33	23.80	*88.88*
n = 30	20.99	**21.40**	17.43	20.85	21.10	18.26	20.22	19.86	*72.53*
Oracle	21.63	23.22	18.31	22.34	**23.44**	19.85	22.02	21.68	*95.58*

RWN

	Freq.	NB	Link-v	Link-r	Link-c	Com-v	Com-r	Com-c	Max
n = 5	32.56	25.63	28.14	32.92	32.56	29.02	**36.10**	32.03	*90.97*
n = 10	30.17	**34.86**	21.06	29.29	27.43	24.07	29.64	27.43	*76.46*
n = 20	22.61	**27.61**	17.34	22.61	21.15	19.07	23.8	19.82	*52.56*
n = 30	19.7	**22.03**	14.51	18.96	17.43	16.04	19.26	16.25	*37.05*
Oracle	**27.19**	18.28	19.81	26.32	25.09	22.71	25.97	23.88	*96.16*

DK

	Freq.	NB	Link-v	Link-r	Link-c	Com-v	Com-r	Com-c	Max
n = 5	24.66	**35.00**	20.58	33.83	28.66	22.16	33.08	26.08	*100.00*
n = 10	19.08	**27.33**	19.79	27.29	22.16	18.00	25.66	20.91	*100.00*
n = 20	15.33	**22.25**	15.83	21.39	18.33	16.54	20.66	17.47	*100.00*
n = 30	13.34	**19.45**	13.88	19.09	16.79	14.45	18.29	15.59	*99.09*
Oracle	9.64	**13.97**	10.35	13.44	12.60	10.92	12.74	11.82	*98.62*

Table 2.3: Commenter prediction precision. The numbers are macro-averaged across posts. Each column contains results for the cut off value noted on the top. See the text body for more explanation.

religion; in both:	people, just, american, church, believe, god, black, jesus, mormon, faith, jews, right, say, mormons, religious, point
in posts:	romney, huckabee, muslim, political, hagee, cabinet, mitt, consider, true, anti, problem, course, views, life, real, speech, moral, answer, jobs, difference, muslims, hardly, going, christianity
in comments:	religion, think, know, really, christian, obama, white, wright, way, said, good, world, science, time, dawkins, human, man, things, fact, years, mean, atheists, blacks, christians
primary; in both:	obama, clinton, mccain, race, win, iowa, delegates, going, people, state, nomination, primary, hillary, election, polls, party, states, voters, campaign, michigan, just
in posts:	huckabee, wins, romney, got, percent, lead, barack, point, majority, ohio, big, victory, strong, pretty, winning, support, primaries, south, rules
in comments:	vote, think, superdelegates, democratic, candidate, pledged, delegate, independents, votes, white, democrats, really, way, caucuses, edwards, florida, supporters, wisconsin, count
Iraq war; in both:	american, iran, just, iraq, people, support, point, country, nuclear, world, power, military, really, government, war, army, right, iraqi, think
in posts:	kind, united, forces, international, presence, political, states, foreign, countries, role, need, making, course, problem, shiite, john, understand, level, idea, security, main
in comments:	israel, sadr, bush, state, way, oil, years, time, going, good, weapons, saddam, know, maliki, want, say, policy, fact, said, shia, troops
energy; in both:	people, just, tax, carbon, think, high, transit, need, live, going, want, problem, way, market, money, income, cost, density
in posts:	idea, public, pretty, course, economic, plan, making, climate, spending, economy, reduce, change, increase, policy, things, stimulus, cuts, low, financial, housing, bad, real
in comments:	taxes, fuel, years, time, rail, oil, cars, car, energy, good, really, lot, point, better, prices, pay, city, know, government, price, work, technology
domestic policy; in both:	people, public, health, care, insurance, college, schools, education, higher, children, think, poor, really, just, kids, want, school, going, better
in posts:	different, things, point, fact, social, work, large, article, getting, inequality, matt, simply, percent, tend, hard, increase, huge, costs, course, policy, happen
in comments:	students, universal, high, good, way, income, money, government, class, problem, pay, americans, private, plan, american, country, immigrants, time, know, taxes, cost

Table 2.4: The most probable words for some CommentLDA topics (MY).

in collaborative filtering).

Following previous work on LDA and its extensions, we show words most strongly associated with a few topics, arguing that some coherent clusters have been discovered. Table 2.4 shows topics discovered in MY (using counting by comments). This is the blog site where our models most consistently outperformed the baseline, therefore we believe the model was a good fit for this dataset. Since the site is concentrated on American politics, many of the topics look alike. Table 2.4 shows the most probable words in the posts, comments, and both together for five hand-picked topics that were relatively transparent. The probabilistic scores of those words are computed with the scoring method suggested by (Blei and Lafferty, 2009).

The model clustered words into topics pertaining to religion and domestic policy (first and last topics in Table 2.4) reasonably. Some of the religion-related words make sense in light of current affairs. Mitt Romney was a candidate for the Republican nomination in 2008 presidential election and is a member of the Church of Jesus Christ of Latter-Day Saints. Another candidate, Mike Huckabee, is an ordained Southern Baptist minister. Moktada al-Sadr is an Iraqi theologian and political activist, and John Hagee is an influential televangelist.

Some words in the comment section are slightly off-topic from the issue of religion, such as *dawkins* or *wright*, but are relevant in the context of real-world events. Richard Dawkins, who is a well known evolutionary biologist, become associated to religious issues in politics since he is a vocal critic of intelligent design. We believe that the word "wright" is a reference to Rev. Jeremiah Wright of Trinity United Church of Christ in Chicago. He is not ordinarily a political figure, but his inflammatory rhetoric was negatively associated with then-candidate Barack Obama.

Notice those words rank highly only in the comment section, showing differences between discussion in the post and the comments. This is also noticeable, for example, in the "primary" topic (second in Table 2.4), where the Republican primary receives more discussion in the main post, and in the "Iraq war" and "energy" topics, where bloggers discuss strategy and commenters focus on the tangible (*oil*, *taxes*, *prices*, *weapons*).

2.6 Predicting Popularity

What makes a blog post noteworthy? There are many ways to define the worth or popularity of a blog post. One plausible assumption is that the extent to which readers are inspired to leave comments directly reflects the popularity of the post.[20] Then a reasonable measure for popularity will be the aggregated volume of users' comments.

In this section we examine the relationship between blog contents and popularity through building of the text-driven prediction models for comment volume. We seek to accurately identify which posts will attract a high-volume response, and to also gain insight about the community of blog readers and their interests. A popularity predictor would be useful in improving technologies for blog search, recommendation, summarization, and so on. A popularity predictor also has the potential to reveal the interests of a blog's readership community to its authors, readers, advertisers, and scientists studying the blogosphere. The latter type of usage prefers the models which easier to understand by human. We will examine our model from this perspective as well as the prediction performance later in the experiment section.

The basic approach here is similar to the previous prediction task. We employ a generative

[20] (Mishne and Glance, 2006) empirically tested the correspondence between this post popularity and the post comment volume.

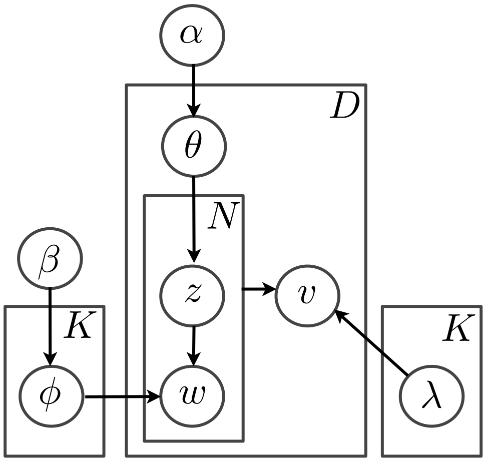

Figure 2.3: Plate notation for Topic Poisson model. The arrow connecting the topics to the comment volume (v) is drawn from the plate rather than the single node to reflect the model detail that the volume is conditioned on the *aggregate* statistics from $z_{0..n}$. See the generative story below.

approach; therefore, we will first design the stochastic generative model (or generative story), and then derive the prediction procedure as the posterior inference over the unknown variables. As in the last task, we take a standard latent topic model as a starting block, and then develop our own model by gradually augmenting it with those components unique to our task and data set. We also adapt the core concept that the post's content and its responses are connected through the shared set of topics. The chief difference at this time is the type of variable for response. While before the response was represented as a set of multinomials, in this task we postulate it as a single count variable.

The change is a natural extension from the basic model. It require us to devise suitable training and posterior inference procedures (Section 2.6.2). It however allows us a novel way to explore and visualized the learned model (Section 2.6.5). Presently we start with our generative story, then describe the posterior inference procedure in the next section.

2.6.1 Model Specification

In the previous task we designed models which simultaneously generate comment contents and user id. In this section we suppose that one's interest in a blog community is something simpler, merely an aggregated *volume* of the comments without their details.

As in the last task, we view the generation of blog articles as driven by underlying thematic topics. We employ the LDA admixture model to concisely formalize this concept as a stochastic generative process. We further hypothesize that the same thematic topics that characterize

Variable	Description
D	Total number of documents
θ_d	Distribution over the topics for document (blog post) d
α	Dirichlet hyper-parameters on θ_d
K	Total number of topics
$\boldsymbol{\phi}_k$	Distribution over words conditioned on topic k
β	Dirichlet hyper-parameters on $\boldsymbol{\phi}_k$
$z_{d,i}$	The (latent) random variable for topic at position i in d
$w_{d,i}$	Random variable for the word at position i in d
λ_k	Mean Poisson parameter associated to topic k
m_k	Mixture coefficient for k
v_d	Random variable for the comment volume for d

Table 2.5: Notation for the generative models

the post's content also influence the generation of response; in this case, we represent it as the count or the volume of the comments. Since we need a type of distribution which supports count values, we represent the comment volume as a mixture of Poisson distribution. Poisson distribution is a discrete probability distribution often used for modeling the occurrence of events in a fixed interval of time. The shape of this distribution is controlled by a single mean parameter (a rate parameter). We associate each mean parameter of the component Poisson distribution to a unique topic, and then set the mixture coefficient to be proportional to the topic distribution of the main body of the blog post. This way the document-level topic mixture, θ, also has an effect on the response generation.

Generative story

We formally define this model, **Topic-Poisson model**, as the following generative story. Unless otherwise noted, we reuse the same variable names from the previous task. (See Table 2.5.) We introduce λ_k to represent the mean Poisson parameter associated to topic k, and m_k for the mixture coefficient for k, and v_d is the random variable for the comment volume for d:

1. (Once for the text collection:) For k from 1 to K, choose a distribution $\boldsymbol{\phi}_k$ over words according to a symmetric Dirichlet distribution parameterized by β.
2. For each blog post d from 1 to D:
   (a) Choose a distribution θ_d over topics according to a symmetric Dirichlet distribution parameterized by α.
   (b) For i from 1 to N_d (the length of the dth post):
      i. Choose a topic $z_{d,i}$ from the distribution θ_d.
      ii. Choose a word $w_{d,i}$ from $\phi_{z_{d,i}}$
   (c) For k from 1 to K, let

$$m_{d,k} \leftarrow \frac{\mathrm{C}(k; \boldsymbol{z}_d) + \alpha_k}{\sum_{k'=1}^{K} \mathrm{C}(k'; \boldsymbol{z}_d) + \alpha_{k'}} \tag{2.3}$$

(d) Choose a comment volume v_d from the Poisson mixture distribution;

$$v_d \sim \sum_{k=1}^{K} m_{d,k} \cdot \mathrm{Pois}(\cdot; \lambda_k) \tag{2.4}$$

The corresponding plate diagram is shown in Figure 2.3. Note that the model is identical to LDA until step 2-c (represented as the left chamber in the diagram), where we define document-specific mixture coefficients and generate the volume count from the mixture mode. The joint distribution of the above generative story is (for one document):

$$p(\boldsymbol{w}_d, \boldsymbol{z}_d, \theta_d, v_d) = p(\theta_d; \alpha) \cdot \sum_{k=1}^{K} m_{d,k} \cdot \mathrm{Pois}(v_k; \lambda_k) \cdot \prod_{i=1}^{N_d} p(w_{d,i}; \beta_{z_{d,i}}) \cdot p(z_{d,i}|\theta_d)$$

Note that this model is essentially a type of "supervised" or "annotated" LDA (Blei and McAuliffe, 2008; Blei and Jordan, 2003; Mimno and McCallum, 2008; Zhu et al., 2009), where response variables are generated based on topics, and therefore influence how those topics are learned during training. Most similar to our work (in terms of model design) is perhaps sLDA (Blei and McAuliffe, 2008), where the response variable, an unconstrained real number, is generated alongside the text data. The number is modeled as a random distribution from a normal linear model, where the mean parameter is defined as the dot product between the empirical topic frequency and the regression coefficients (Blei and McAuliffe, 2008).

In our work the response variable is a count value, which we represent as a Poisson mixture distribution. We use Gibbs sampling for model inference, while (Blei and McAuliffe, 2008) used a variational approximation. We initially experimented with the original sLDA algorithm for our prediction task but were not able to obtain sufficiently competitive results. We therefore chose not to extend their algorithm for this task. We also refrained from using it as our comparative baseline at this time for the same reason. Instead, we use Naïve Bayes classifier and elastic net regularized linear regression, both of which outperformed sLDA substantially in preliminary experiments.

We note that in (Blei and McAuliffe, 2008) the authors presented the discussion on how their basic model can be viewed as a specific version of the more general model, by appealing to the fact that the normal linear model is in fact a type of generalized linear model (GLM) (McCullagh and Nelder, 1989) with specific link function. They argued that other exponential family of distributions are also generalizable with their framework, but did not present specifics (or inference algorithms) for response types other than normal linear model.

Prediction

Given a trained model and a new blog post, we can derive a prediction on the comment volume as a series of posterior inference.[21] For a new post d, we first infer its topic distribution θ_d; since we do not observe any part of the comment, we estimate this posterior from the words in the post $\boldsymbol{w}_d$ alone. We have explained this part in the experimental section of the first task (Section 2.5.3). Once the θ_d is estimated, the expected value for v_d, the comment volume, can

[21] This can be viewed as a query into a stochastic model as well.

be computed trivially from the definition in Equation 2.4. Note the mean (expectation) of the component Poisson distribution is just the rate parameter, λ.

The remaining question here is the parameter estimation, or training, of the model, which we discuss in the next section.

2.6.2 Notes on Inference and Estimations

The inference over latent variables and the parameter estimation is slightly more complicated in this model than our previous task, since they are no longer a simple extension from a plain LDA. We will point out the necessary details here. For a complete picture, see Section 2.4.2 and 2.5.2.

As before, we take standard Bayesian estimation approach to the inference; We seek a maximum *a posteriori* estimate of $\boldsymbol{\phi}$ and $\boldsymbol{\lambda}$, marginalizing out $\boldsymbol{\theta}$, each word's topic z, and fixing $\alpha = 0.1$ and $\beta = 0.1$. During training (learning of the parameters), the words and volumes are, of course, observed. The topic assignments, $\boldsymbol{z}_d$, are latent variable and never will be observed at any time.

For inference over the latent variables $\boldsymbol{z}_d$, we use collapsed Gibbs sampling (Heinrich, 2008; Griffiths and Steyvers, 2004). Each latent topic depends in part on the volume, so that the Gibbs sampler draws topic $z_{d,i}$ for word $w_{d,i}$ according to:

$$p(z_{d,i} = k \mid \boldsymbol{z}^{\neg d,i}, w_{d,i} = t, v_d, \alpha, \beta, \theta_d, \boldsymbol{\lambda}) = \frac{\mathrm{C}(k; \boldsymbol{z}^{\neg d,i}) + \alpha_k}{\sum_{k=1}^{K} \mathrm{C}(k; \boldsymbol{z}^{\neg d,i}) + \alpha_k}$$
$$\times \frac{\mathrm{C}(t, k; \boldsymbol{z}^{\neg d,i}) + \beta_t}{\sum_{t'} \mathrm{C}(t', k; \boldsymbol{z}^{\neg d,i}) + \beta_t} \times \left(\sum_{k=1}^{K} m_{d,k} \cdot p(v_d | \lambda_k) \right)$$

where $m_{d,k}$ comes from Eq. 2.3, with $z_i = z$. Note that the sampling distribution depends on the mixture coefficients, which are calculated directly from the document's topics $\boldsymbol{z}_d$ in the current sample according to Eq. 2.3.

We use a mixture of Poisson for the comment volume, so that for all $v \in \mathbb{N}$,

$$p(v \mid \lambda_k) = e^{-\lambda_k} \lambda_k^v \big/ v! \tag{2.5}$$

To estimate the λ_k, we embed the Gibbs sampler in a stochastic EM algorithm (Casella and Robert, 2004) that re-estimates the λ_k after resampling the $\boldsymbol{z}$ for each document in turn, according to the maximum likelihood formula:

$$\lambda_k \leftarrow \left(\sum_{d=1}^{D} \theta_{d,k} v_d \right) \Big/ \left(\sum_{d=1}^{D} \theta_{d,k} \right) \tag{2.6}$$

In our application (Section 2.6.3), we re-estimate the Poisson parameters at each EM iteration and use them as the posterior distribution at the next.

2.6.3 Experimental Results

We implemented our Topic-Poisson model for all five blog sites from our Political Blog Corpus (Section 2.3). We will touch upon all the results, but focus our discussion more on the two blog sites which performed well on previous topic model experiments (Section 2.5.3), Matthew Yglesias ("MY") and RedState ("RS"). The section on the descriptive aspects of the models (Section 2.6.5) only discusses these sites. For each site, we first trained models with the training portion to estimate the model parameters, and then inferred the expected comment volume value, $\mathbb{E}[v_d]$ for each blog post d in the test set as described in Section 2.6.1.

In all cases we use the same temporal training-test splits as the first task (i.e., the test posts strictly come later than the training posts). The text processing too is done in the same way as the first task; all posts are represented as text only (images, hyperlinks, and other non-text elements were ignored) and as a bag of unigram words. Words occurring two times or fewer in the training data and stop words were removed. No stemming was performed. Posts with fewer than five words were discarded.

Since we have all the comment contents in our data, we have several choices in how to aggregate them into a single count. Volume might be measured as the number of words in the comment section, the number of comments, the number of distinct users who leave comments, or a variety of other ways.[22] In this experiment we consider two types of measurements, one in word tokens in all the comments (denoted "#word"), and the other in the count of individual comments ("#comment").

As an exploration, we experiment with a few variations of our Topic-Poisson model. We report their performance on the prediction tasks alongside the main model.

Evaluation Setup

Our Topic-Poisson model outputs a predicted "volume" of comments given an input blog post. Although it is conceivable that predicting the absolute volume would be directly useful for some end task (perhaps estimating storage capacity), we think that, in many user assistive application developments, *relative* quantity is likely more important. For example, if one is interested in comment volume as a proxy for post popularity, what is really important is which posts are expected to elicit *more* comments than "other" posts. (Note that other qualitative attributes, such as user interest or reader engagement, also concern relative quantities among the posts.) Since we do not have a clearly defined "others" to turn our predictive volumes into a (relative) "popularity" prediction, for the sake of comparison, we postulate a hypothetical mediocre blog post in our evaluation.

To be more specific, we apply our model's outputs to a type of popularity prediction, in which

[22] In fact, any aggregate statistics which can be represented as a single positive scalar value is usable in this model with no modification. Task-specific count values, such as the statistics in forwarding, or the count of "negative" words or other types of sentiment scores would be an interesting future application.

the task is to tell whether the given post will receive a more comments than our hypothetical mediocre article. Since we don't actually know how many comments the perfectly mediocre article produces, we approximate that number as the mean comment volume among the training blog posts.

Since this task essentially categorizes each post in the test set into the higher-than-average class or not, this is a binary classification task. Therefore, we use precision and recall measurements as the performance metrics. We compare our model with a classification algorithm (bag of words Naïve Bayes classifier) and a regression algorithm (elastic net regularized linear regression). In the case of the regression baseline, we transform the numerical predictive outputs to binary values using the same transformation function as the proposed model. Note that, even though our proposed model (and also our linear regression baseline) is applicable to classification tasks through a transformation, it is more powerful, as it gives a distribution over values for v, permitting more fine-grained prediction and analysis (e.g., ranking the set of posts by popularity.)

The mean volume is approximately 1424 words (35 comments) for MY and 819 words (29 comments) for RS. The distribution is skewed, with roughly one third of the posts having below-average volume. The MY data shows a strange effect: the test set has a much greater rate of high-volume posts (66%) compared to the training data (35%), potentially making the prediction task much harder. Note that accuracy is another applicable evaluation metric asides from precision and recall for our task. We chose precision and recall because this uneven distribution of our data makes the accuracy measurement less desirable.

Baseline: Bag of Words Naïve Bayes Model

Naïve Bayes is a widely used model for classification that can be used for the binary prediction task. Let $\bar{v}$ be the mean value of the volume variable we seek to predict, calculated on the training data. Let V be the (unknown volume) for a blog post that is represented as a word sequence $\boldsymbol{w} = \langle w_1, \ldots, w_N \rangle$.

$$\begin{aligned} p(V > \bar{v}, \boldsymbol{w}) &= p(V > \bar{v}) \times \textstyle\prod_{i=1}^{N} p(w_i \mid V > \bar{v}) \\ p(V < \bar{v}, \boldsymbol{w}) &= p(V < \bar{v}) \times \textstyle\prod_{i=1}^{N} p(w_i \mid V < \bar{v}) \end{aligned}$$

The generative model assumes that, first, the class ("high volume" or "low volume") is chosen according to a binomial distribution, then the words are generated IID conditioned on the class. Maximum likelihood estimates for the parameters are obtained straightforwardly from training data. Unobserved words are ignored at test time.

The results from this model are labeled as "NB" in Table 2.6 and Table 2.7. On all sites, the Naïve Bayes model tends to err on the side of precision, except for the comment volume prediction on Daily Kos ("DK"). This is also the task where this model outperforms all others (in terms of F1 score), along with the comment volume prediction task on RS. Note that comment volume in general is harder to predict from words for all the models.

Beyond the performance of the predictor on this task, we may ask what the model tells us about the blog and its readers. The Naïve Bayes model does not provide much insight. Ranked by likelihood ratio, $p(w \mid V > \bar{v})/p(w \mid V < \bar{v})$, the strongest features for "high word volume" from

		# words					# comments				
		prec.		rec.		F_1	prec.		rec.		F_1
MY	Naïve Bayes	72.5		41.7		52.9	42.6		38.8		40.6
	Regression	81.5		44.1		57.2	60.8		55.2		57.8
	T-Poisson	70.1	(±1.8)	63.2	(±2.5)	66.4	41.3	(±2.1)	53.1	(±3.5)	46.4
	k=30	71.8	(±2.0)	60.1	(±3.4)	65.4	45.3	(±2.1)	54.2	(±5.3)	49.3
	k=40	71.0	(±1.9)	63.4	(±2.7)	66.9	44.0	(±2.1)	58.8	(±3.3)	50.3
	T-NBin.	69.7	(±2.3)	62.5	(±2.5)	65.9	38.4	(±2.2)	45.7	(±3.3)	41.7
	C-LDA	70.2	(±2.3)	68.8	(±2.5)	69.4	37.2	(±1.5)	50.4	(±3.3)	42.8
RS	Naïve Bayes	64.1		25.7		36.6	37.8		34.1		35.0
	Regression	52.0		26.8		35.5	20.5		19.5		20.0
	T-Poisson	52.4	(±2.8)	33.5	(±2.0)	40.8	25.4	(±2.6)	27.9	(±2.9)	26.7
	k=30	58.3	(±3.5)	35.8	(±2.7)	44.3	24.2	(±3.5)	32.0	(±4.9)	27.5
	k=40	55.0	(±2.3)	33.6	(±2.7)	41.7	27.3	(±2.9)	35.4	(±4.6)	30.8
	T-NBin.	53.9	(±4.5)	25.2	(±2.7)	34.3	22.5	(±3.2)	24.9	(±3.4)	23.6
	C-LDA	57.8	(±3.2)	25.7	(±1.7)	35.6	22.6	(±3.2)	27.6	(±4.7)	24.8

Table 2.6: Experiments: precision and recall for "high volume" posts. NB= Naïve Bayes classifier, Reg. = regression , T-Poisson = Topic-Poisson, T-NBin. = Topic-Negative Binomial, C-LDA = CommentLDA. Topic models are "ave. (±s.d.)" across 10 runs.

MY are *alleged*, *instability*, *current*, *crumbling*, *canaries*, *imaginations*, *craft*, *cars*, *imagine*, *funnier*.

Baseline: Linear Regression

Regression is another approach suitable for predicting numerical values. We tested linear regression with elastic net regularization ("glmnet") (Friedman et al., 2010).[23] This approach permits easy tuning of the regularization constant. We trained 1,000 regression models (at different regularization settings) with the same word features as above. We report here the binary prediction performance of the best model, selected using a 10% held-out set. The regression output from the model is mapped to the binary prediction at the evaluation time as we did for our Topic-Poisson models. The results from this model is labeled as "Regression' in Table 2.6 and Table 2.7. The performance from this model is reasonably competitive on the MY site, however on RS and also on DK, the model performs worse than Naïve Bayes. This is the best performing model on MY and The Carpetbagger Report ("CB").

Results

We show our experimental results in Table 2.6 and Table 2.7. In the table, "T-Poisson", both $k = 30$ and $k = 40$, are our prediction model described in the Section 2.6.1. "k" referred to the topic size, fixed at 20, 30 and 40. "T-NBin" and "C-LDA" are the variations of this model. See the next section for more detail. Our two contending models are the naïve Bayes model and the regression model ("Naïve Bayes" and "Regression" in the table) which we explained in the

[23] `http://cran.r-project.org/web/packages/glmnet/`

		# words					# comments				
		prec.		rec.		F_1	prec.		rec.		F_1
CB	Naïve Bayes	99.8		18.5		31.3	66.7		41.3		51.0
	Regression	88.6		28.7		43.4	56.2		79.4		65.7
	T-Poisson	93.2	(±1.7)	42.4	(±3.8)	58.3	58.3	(±2.5)	63.3	(±3.1)	60.7
	k=30	96.5	(±2.3)	38.5	(±3.7)	55.1	53.0	(±3.4)	47.0	(±3.8)	49.8
	k=40	94.5	(±1.8)	36.7	(±3.2)	52.8	60.5	(±3.2)	57.9	(±3.8)	59.2
	T-NBin.	94.2	(±2.3)	42.2	(±2.8)	58.3	58.7	(±1.8)	62.4	(±2.7)	60.5
	C-LDA	92.3	(±1.6)	46.7	(±2.7)	62.0	55.2	(±2.0)	64.1	(±2.8)	59.3
DK	Naïve Bayes	97.9		48.7		65.0	69.4		76.6		72.8
	Regression	95.2		41.4		57.7	67.5		70.1		68.8
	T-Poisson	90.9	(±1.0)	59.8	(±1.8)	72.1	56.5	(±1.6)	83.4	(±2.3)	67.3
	k=30	92.9	(±1.9)	56.8	(±2.2)	70.5	59.6	(±1.6)	80.8	(±2.0)	68.6
	k=40	92.5	(±1.5)	55.9	(±0.9)	69.7	60.5	(±2.1)	78.3	(±1.9)	68.2
	T-NBin.	91.0	(±1.0)	60.1	(±1.4)	72.3	55.3	(±0.7)	81.7	(±2.0)	65.9
	C-LDA	93.4	(±1.2)	55.0	(±2.0)	69.2	59.5	(±2.1)	81.8	(±2.8)	68.9
RWN	Naïve Bayes	46.7		13.5		20.9	25.0		11.5		15.8
	Regression	63.4		50.1		55.9	30.8		46.2		36.9
	T-Poisson	49.2	(±1.1)	86.0	(±3.0)	62.6	25.3	(±1.4)	84.6	(±5.1)	38.9
	k=30	50.9	(±1.3)	77.3	(±2.8)	61.3	27.3	(±1.4)	81.9	(±6.5)	41.0
	k=40	51.0	(±1.2)	83.3	(±2.6)	63.3	27.3	(±1.1)	84.2	(±4.6)	41.2
	T-NBin.	49.5	(±1.5)	88.1	(±3.0)	63.4	24.9	(±1.1)	85.0	(±3.8)	38.5
	C-LDA	53.9	(±1.8)	72.9	(±4.5)	62.0	29.7	(±1.7)	76.2	(±5.7)	42.8

Table 2.7: Additional experiments: precision and recall for "high volume" posts. NB= Naïve Bayes classifier, Reg. = regression , T-Poisson = Topic-Poisson, T-NBin. = Topic-Negative Binomial, C-LDA = CommentLDA. Topic models are "ave. (±s.d.)" across 10 runs.

previous section.

Overall, our models are comparable to the two contending models. On all blog sites, T-poisson outperforms both baselines in the word volume prediction task. In the comment volume prediction task, for four of the blogs it outperforms only one of the two baselines; for the fifth, Right Wing News ("RWN"), it outperforms both. The standard deviation of the prediction performance on this site however is much larger than others, perhaps because of its smaller data size. On the MY data, our volume prediction model ("T-Poisson") improves recall substantially over Naïve Bayes, on both measures, with a slight loss in precision. Its precision lags behind the regression model, gaining in recall on word volume prediction but not on comment volume prediction. The effect is similar on RS data when predicting *word* volume, but the loss in precision is much greater, and the model is ineffective for *comment* volume. Note that comment volume on RS is harder to predict from words. The regression model is much less effective on the RS data set, falling behind Naïve-Bayes on both tasks.

2.6.4 Model Variations

Above Table 2.6 mentions several variations of the Topic-Poisson model. We will note some more comments on these results, and further explain the detail of these variations.

More topics

We tested the Topic-Poisson with various size of topics. These are shown as "$K = 30$" and "$K = 40$" in Table 2.6. More topics had a negligible effect on the word-volume task but improved the precision and recall on the comment volume task substantially. On inspection, the topics discovered by these models were more difficult to understand.

Negative Binomial distribution

Naturally, Poisson distribution is not the only distribution which supports the count value. One type of distribution often used in place of Poisson distribution is Negative Binomial. Negative Binomial is a discrete probability distribution, usually use to model the number of successes in a sequence of Bernoulli trials before a pre-defined number of failures. The shape of distribution is controlled by two parameter, therefore can be more flexible to fit the model.

We tested the model with a mixture of negative binomials instead of Poisson:

$$p(v; \rho_k, r_k) = \binom{v + r_k - 1}{r_k - 1} \rho_k^{r_k} (1 - \rho_k^v) \tag{2.7}$$

Where the two parameters of component negative binomials are represented as ρ_k and r_k. To train the model with this response variable, we need to optimize ρ_k and r_k during the M step of the training algorithm. Unlike the mixture of Poisson distribution, there is no closed form maximum likelihood estimate for Negative Binomial. Therefore we used moment matching method (Casella and Berger, 2001) to estimate ρ_k and r_k. The experimental results of this change (Table 2.6) were negative. While the negative binomial offers more expressive power and degrees of freedom than the Poisson, it tends toward the Poisson as $\rho \to 0$; estimated ρ_k values were, indeed, close to 0.

User identities and comment content

Borrowing the models we used for the previous idea, we further extended LDA to model the user identities and comment words in addition to the post words (along with the comment volume). The model thus prefers the topics which explain the comment volume, as well as those additional observations. We tested the CommentLDA model with "counting by comments" (see Section 2.6.1), and achieved substantial gains in word volume prediction on MY and CB with similar recall to other models. This approach is in general harmful to comment volume prediction. [24]

[24] Note that we did not use user id or comment words during the inference on the test examples, although they are used during the training of the model.

2.6.5 Descriptive Aspects of the Models

An attractive property of latent topic models is that they discover human interpretable topics from unstructured raw text data. An ordinarily latent topic model induces thematics topics, which are represented as a set of distribution over words. With our Topic-Poisson model, we can additionally characterize each topic by λ_k, the mean for its Poisson distribution over the comment volume. This parameter can be understood as the default popularity of the associated topics, which can be much different across different partisan communities and their ideological issue positions. Below we use the trained model for profiling, or summarizing blog sites, by *topic preference ranking*.

We contrasted the two blog sites, each from the opposite spectrum of American politics, in terms of issue popularities in each community. Table 2.8 shows the topics discovered in Matthew Yglesias (MY) by our model (using the word count as a proxy for the popularity volume) and Table 2.9 the topics from Red States (RS). Topics are ranked by λ_k; words are selected by the scoring algorithm introduced in (Blei and Lafferty, 2009), which account for the given words' overall frequency in the corpus as well as their likelihood in the topic.[25]

The most comment-worthy topics on the liberal blog MY appear to deal with the gender/race issue and the Democratic presidential primary race. On the conservative RS blog, the Republican presidential primary race, and discussion about the party itself dominate. On both blogs, discussion of internal races for party nominations is clearly the most likely to incite readers of these partisan blogs to comment. Some clear issues arise as topics. On MY, the Middle East and energy are the seventh and eighth topics; healthcare is slightly below the overall average. RS rates religion very high (fourth topic), with the economy just above average and Iraq/Afghanistan well below. Note that the least comment-worthy topics on MY have to do with sports and reading material, which interest Matthew Yglesias but perhaps not his readers.

The table also shows the binary accuracy on posts associated with each topic. We assign a post to each topic k that has $\theta_{d,k} \geq 0.25$ (a post can go to zero, one, or more topics), and measure binary prediction accuracy within the topic. These accuracies are based mostly on very small numbers of posts, so our analysis is tentative. On MY, the most comment-worthy topics are also the ones that our model is most accurate at classifying. Part of the variation across topics may be due to temporal changes in topics and reader interest between training and (later) testing periods.

2.7 Related Works

Below we review some works relevant to the tasks discussed in this chapter. Some of the works have already been mentioned in the previous sections. We repeat them here in the context when it serves for the sake of clarity. See section 1.1 and 2.2 for the additional discussions on the related works.

[25] It is basically an approximate measure for the word's "peculiarity" to the topic.

In topic modeling

Latent topic modeling has become a widely used unsupervised text analysis tool. The basic aim of those models is to discover recurring patterns of "topics" within a text collection. LDA was introduced by (Blei et al., 2003) and has been especially popular because it can be understood as a generative model and because it discovers understandable topics in many scenarios (Steyvers and Griffiths, 2007). Its declarative specification makes it easy to extend for new kinds of text collections. The technique has been applied to Web document collections, notably for community discovery in social networks (Zhang et al., 2007a), opinion mining in user reviews (Titov and McDonald, 2008), and sentiment discovery in free-text annotations (Branavan et al., 2008). (Dredze et al., 2008) applied LDA to a collection of email for summary keyword extraction. The authors evaluated the model with proxy tasks such as recipient prediction. More closely related to the data considered in this work, (Lin et al., 2008) applied a variation of LDA to ideological discourse.

A notable trend in the recent research is to augment the models to describe non-textual evidence alongside the document collection.[26] Several such studies are especially relevant to our work. (Blei and Jordan, 2003) were one of the earliest results in this trend. The concept was developed into more general framework by (Blei and McAuliffe, 2008). (Steyvers et al., 2004) and (Rosen-Zvi et al., 2004) first extended LDA to explicitly model the influence of *authorship*, applying the model to a collection of academic papers from CiteSeer. The model combined the ideas from the mixture model proposed by (McCallum, 1999) and LDA. In this model, an abstract notion "author" is associated with a distribution over topics. Another approach to the same document collection based on LDA was used for citation network analysis. (Erosheva et al., 2004), following (Cohn and Hofmann, 2001), defined a generative process not only for each word in the text, but also its citation to other documents in the collection, thereby capturing the notion of *relations* between the document into one generative process. (Nallapati and Cohen, 2008) introduced the Link-PLSA-LDA model, in which the contents of the citing document and the "influences" on the document (its citations to existing literature), as well as the contents of the cited documents, are modeled together. They further applied the Link-PLSA-LDA model to a blog corpus to analyze its cross citation structure via hyperlinks.

In this work, we aim to model the data *within* blog conversations, focusing on comments left by a blog community in response to a blogger's post.

One common use of those models is prediction. Although unsupervised settings are more common among the topic model researchers, some of them incorporate the observation on response variables during its training. One of the earliest to example of the trend is aforementioned (Blei and Jordan, 2003). Mimno et al. ((Mimno and McCallum, 2008)) and Zhu et al. ((Zhu et al., 2009)), also proposed LDA model with the labeled (or annotated) response variables in the generative story. Our volume prediction models (Section 2.6.1) are similar to those model in overall approach (supervised training with latent topic variables), but differ in terms of what each variables represent, and how the response variables parameterized. For example, in supervised LDA (Blei and McAuliffe, 2008) generalized linear model is incorporated in the generative story to represent the response variables, while in our model the response variables are sampled from a mixture of poisson distribution. We discussed our works's connection to SLDA in detail in the

[26] Some of the models mentioned in the previous paragraph fit into this description as well

model specification section (Section 2.6.1)

On politics, blogs, and user generated texts

Network analysis, including citation analysis, has been applied to document collections on the Web (Cohn and Hofmann, 2001). (Adamic and Glance, 2005) applied network analysis to the political blogosphere. The study modeled the large, complex structure of the political blogosphere as a network of hyperlinks among the blog sites, demonstrated the viability of link structure for information discovery, though their analysis of text content was less extensive. In contrast, the blog contents seems to be of interest to social scientists studying blogs as an artifact of the political process. Although attempts to quantitatively analyze the contents of political texts have been made, results from classical, supervised text classification experiments are mixed (Mullen and Malouf, 2006; Malouf and Mullen, 2007).

A few studies have focused on information in comments. The best known is perhaps (Mishne and Glance, 2006), which showed the value of comments in characterizing the social repercussions of a post, including popularity and controversy. Their large-scale user study correlated popularity and comment activity. In recent years, researchers start to pay more attention to comment texts for the purpose of opinion and sentiment analysis, or document recommendation or filtering (Park et al., 2011; Filippova and Hall, 2011; Mukherjee and Liu, 2012; Potthast et al., 2012; Ko et al., 2011; Fang et al., 2012). In closely related subject domain (Balasubramanyan et al., 2012, 2011; Das et al., 2009) study with political opinion and sentiment in blogs and comments from generative perspective, with application to extrinsic prediction tasks. Model design and prediction targets in these works, naturally, are much different from our models here.

2.8 Summary and Contribution

In this chapter we presented statistical models which predict readership reaction from a post on a political blog site.

The quantitative evaluations showed that those models are competitive in realistic settings. We also found that there are notable differences in performance across the blog sites. No one variation of the model works uniformly well across all blogs. Difference in blogging style, as well as the relative size of data, are most likely the cause of those differences.

Along with the quantitative evaluation, we reported here some of findings in the learned models. Those discoveries show that there are patterns between the post and each community's response. We built up our models from plain LDA, augment them so that the latent topics link the two halves of the blogging (post and comments). Because of this construction, our models can illustrate what topics would generate what reactions. In the case of the user prediction task, the reaction is summarized by the set of users who would be interested in the discussion of the post. We also show the differences in blogger and commenter language (or the different focus in the given issues) with CommentLDA.

In the case of the popularity prediction task, the reactions are represented as the volume of comments; thereby we are able to make a probabilistic statement on what topics interest the community. We showed that the model can tell in a succinct fashion not only which blog post would be popular but also what words relate to popularity in the community. Such a site profiling technique is potentially useful for researchers who need to analyze blog communities or for content providers who wish to keep track of the popularity trends among their readers.

To summarize, our contributions from this chapter are the delivery of the following:

1. Core analytical technologies which can be applied to such intelligent application as document prioritization of personal recommendation in and related to online media.
2. A novel case of inquiry into the political blog forums and the political subcultures reflected on their written communications.
3. Applications of latent topic modeling to a type of user generated text which has been much underused in traditional natural language research.

λ_k	**Topic Words**	**# Posts**	**Accuracy**
1873	women black white men people liberal civil working woman rights	7	(100)
1730	obama clinton campaign hillary barack president presidential really senator democratic	13	(77)
1643	think people policy really way just good political kind going	74	(72)
1561	conservative party political democrats democratic republican republicans immigration gop right	12	(50)
1521	people city school college photo creative states license good time	19	(58)
1484	romney huckabee giuliani mitt mike rudy muslim church really republican	3	(33)
1478	iran world nuclear israel united states foreign war international iranian	16	(69)
1452	carbon oil trade emissions change climate energy human global world	6	(33)
1425	obama clinton win campaign mccain hillary primary voters vote race	22	(64)
1352	health economic plan care tax spending economy money people insurance	22	(55)
1263	iraq war military government american iraq troops forces security years	24	(58)
1246	administration bush congress torture law intelligence legal president cia government	5	(20)
1215	mccain john bush president campaign policy know george press man	20	(60)
1025	team game season defense good trade play player better best	8	(38)
1007	book times news read article post blog know media good	23	(43)
	Overall:	183	(58)

Table 2.8: MY Topic-Poisson model: Poisson parameter estimate and top words for each topic. See text for explanation. Shown here is the complete set of fifteen topics induced by our model.

λ_k	**Topic Words**	**# Posts**	**Accuracy**
1546	romney huckabee mccain thompson rudy mitt fred campaign iowa mike	4	(75)
1378	mccain party conservative conservatives republican candidate john issues gop republicans	9	(44)
1030	paul dan ron energy think oil thomas pick mclaughlin change	6	(33)
977	man men america american life god great religion believe jesus	9	(44)
954	court law justice supreme amendment general attorney state school states	8	(50)
857	obama hillary clinton win democratic vote primary party race nomination	18	(50)
846	economy market fed markets money rate mortgage prices inflation financial	15	(60)
789	just way good people know right think time want say	146	(61)
692	tax health government care taxes state federal insurance spending money	14	(50)
618	obama barack hillary campaign clinton senator mccain wright john church	44	(63)
594	just moe really going senator know lane right update man	35	(62)
592	iraq war troops qaeda iraqi intelligence afghanistan surge security general	7	(42)
583	president trade united policy bush states foreign world israel iran	9	(77)
578	said hillary asked mccain host russert obama schieffer wallace barry	11	(45)
545	democrats republican congress house republicans senate democrat rep gop year	19	(36)
	Overall:	231	(59)

Table 2.9: RS Topic-Poisson model: Poisson parameter estimate and top words for each topic. See text for explanation. Shown here is the complete set of fifteen topics induced by our model.

Chapter 3

The Congress

In this section we delve into one of the most important institutions in American politics, the United States Congress. The Congress plays the central role in the nation's legislative process, and its political dynamics have long been the subject of scholastic inquiries. Quantitative inquiries are staples in this area; however, statistical text analysis has never been a mainstay due to the technical difficulties in large-scale natural language processing (NLP). The trend recently has changed significantly thanks to the opportunities in (and the demands from) the rise of online media and electronic archiving. Many thought-provoking questions in this area fit naturally with prediction tasks. We will take on two such questions in this work. The first concerns the survival (and death) of bills through the congressional committee system. The second examines the relationship between election contributions and lawmakers' public speech.

We first refine our tasks with more precise scoping (Section 3.1) in the following section, and then we present a short discussion on our subject (Section 3.2). We will describe each prediction model, including experimental results, in two separate sections (Section 3.3 and Section 3.4). Since each task involves a different approach and corpus, we will describe them individually in respective subsections. We conclude the chapter with a summary of our contributions and the plan for future work.

The part of the work described here is previously published in (Yano et al., 2012, 2013).

3.1 Task Definition

In this chapter we examine two prediction tasks; We have introduced them first back in chapter 1. These are:

- Predicting whether a bill is going to survive through the congressional committee system.
- Predicting campaign donations by industry and interest groups to the member of congress.

The textual evidence we focus on are the congressional bills (for the first task) and microblog messages (for the second task).

For the first task, we view congressional bills as the actuating documents, or the agents which evoke the committee decisions. We assume the following prediction scenario: The (predictive) system takes the contents and some meta-data on a bill as inputs. Then, the system outputs the binary prediction on whether the bill is endorsed by the referred committee(s) by the end of the congressional period.[1] As with the last task, we use text as the main evidence for prediction. Unlike the last task, we use some non-textual meta data along with the texts. (We will discuss the detail of our meta data in the next section.)

For the second task, we view the microblog messages by the members of Congress (MC) as the actuating documents. We postulate that the MC's publicly expressed opinions and preferences correlate with the amount of campaign contributions they receive from industries and interest groups. (In subsequent sections we refer the contribution breakdown by the source groups as a "Campaign Profile".) Political scientists have a variety of theories regarding the relationship between the politicians and campaign contributions. Some theorize that the (current and perceived) financial incentives from the industries influence MCs' public stances. Others view the MCs' publicly expressed ideologies as evoking the affinities (or sympathies) from interest groups in the form of campaign contributions.[2] Be they incentives or affinities, both views suggest that politicians' messages to their constituents are driven by their underlying agenda (or issue) preferences, which strongly relate to the money they receive.[3] The prediction setting is the following: given the input, the system will output each industry/interest group's relative share of contribution in the congressional member's entire campaign budget. In other words, the system outputs a probabilistic vector over the known contributing sectors. (Since the contribution size of the total campaign differs considerably among congressional members, we normalize the contribution amount for each member to ensure a more meaningful comparison across individual MCs.)

We will design and implement the prediction systems, then evaluate them with the real world data. In both tasks, we assume the strictly predictive setting. Therefore predictors are to yield the output based only on the content of the actuating texts. Any information on any parts of the reactions are not available at prediction time. In all our experiments we trained and evaluated our models with the corpus prepared by our team.[4] We will describe them in Section 3.3.1 and Section 3.4.1. Presently, we will discuss our subject, the United States Congress.

3.2 Background: The United States Congress

In the U.S., federal laws are passed by the U.S. Congress, which consists of the House of Representatives (commonly called the House) and the Senate. To become law, a bill (i.e., a proposed law) must pass both chambers and then be signed by the U.S. President. If the President refuses

[1] One congressional period equal two years. For example the current congress, the 113th, is between January 3, 2013 to January 3, 2015.

[2] See Section 3.2 for more.

[3] In this work we do not concern ourselves with the direction of causation. Our models assume that the underlying agenda preferences for each MC is a priori given. See Section 3.4.3 for details.

[4] The resource is available at `http://www.ark.cs.cmu.edu/bill-data/` and `http://www.ark.cs.cmu.edu/twt-data/`

to sign a bill (called a veto), it may still become law if both chambers of Congress override the veto through a two-thirds majority. Whenever rewrites (revisions, amendments) are necessary, the new version of the bill repeats the rounds of voting until all parties agree upon a final version (Johnson, 2003).

The Congress consists of democratically elected officials, or the Members of Congress (MC), who act on behalf of their constituencies. The members of the Senate ("senators") represent a state as a whole; each state elects two senators. The House consists of 435 members ("representatives"), each of whom represents a single district within a state. The number of districts a state has depends on the state's population size and therefore varies from state to state. California, a more populous state, has 53 districts, while Montana, a less populous state, has only one representative. Representatives serve for two-year terms and senators for six-year terms. Every two years, all of the representatives' and one third of the senators' seats are up for re-election.

Needless to say, the Congress is a key institution in the nation's lawmaking process, and the MCs, once elected, assume considerable influence therein. The two subjects we deal in this work – congressional committees and campaign finance – concern two of the most fundamental questions: how laws are shaped in the system, and how lawmakers win their seats.

3.2.1 The Committee – Where our Laws Are (Really) Made

When legislatures cast their votes ("Yea" or "Nay") for a bill on the chamber floor and votes are recorded, it is known as a "roll call" vote. A roll call vote means that votes are a part of official records and therefore are subject to the public's scrutiny. Both political scholars and concerned citizens pay close attention to the roll call records.

In comparison to recorded floor votes, the process by which bills come into existence is much less transparent. In fact, bills which get discussed on the floor make up only a small fraction of all the bills that are ever introduced to the system. What happens to those bills that do not culminate in a roll call vote? The majority of them die in **congressional committees**.

At the beginning, a bill is formally proposed by a member of Congress, known as its sponsor. Once proposed, it is routed to one or more (usually just one) of about twenty subject-specializing congressional committees in each chamber. The referred committee, in turn, deliberates on the merits of the bill, and then reports back to Congress if the committee decides the bill deserves further consideration. The legislative rules also give the committees the power to rewrite bills as they see necessary. Of 4,000 to 8,000 bills introduced to the House of Representatives in each Congress, on average 85 percent of them effectively die in the referred committee.[5] Surviving the congressional committee is perhaps the largest hurdle a bill must clear in its life. Once a bill is approved by the committee, it has nearly a 90-percent chance of eventually becoming a law. Since the committee system can prevent certain issues from ever entering into public discussion, it is is often described as the **agenda setting** system (Adler and Wilkerson, 2005; Krutz, 2005).

In spite of the committees' significant authority, the inner workings of committees are rather inscrutable. Unlike the floor votes, there are no clearly stipulated voting rules in the committees.

[5] Note that, in reality, no bills are ever officially killed. They are simply left expired at the end of the Congressional year. These bills could also be resurrected in a later Congress under another guise.

Moreover, the various actions inside the committees are difficult to track, thus subject to much less public scrutiny. Proceedings are not kept consistently, making them difficult to compare across different committees.

Naturally, there is much discussion among political scientists on committee politics (Adler and Wilkerson, 2005; Krutz, 2005; Hall, 1998; Evans, 1991). What are the working principles of the committees? What are the systematic constraints and what are the individual variables? Do the issues in the bill drive the committees decision, or is that decision solely a function of personal relations among the actors? And, most of all, how do those various factors relate to the final outcome by the committee as a whole?

3.2.2 Campaign Finance – How our Lawmakers Are Made

Campaign financing is one of the most important factors in winning an election. During the 2012 U.S. general election, Democrat and Republican candidates collectively spent nearly 1.1 billion dollars on congressional campaigns (771,967,566 dollars for the House and 312,679,221 dollars for the Senate).[6] This is more than a 45-percent increase in campaign spending since 2000, just three election cycles ago, when congressional candidates spent just over 600 million dollars.[7] Political campaigns are becoming rapidly more expensive, and there is no question that the vast amount of money spent by campaigns influences the American political landscape.

The rising price tags on elections, and the influence of financially endowed interest groups in politics, have always been of concern. Over the years, there have been several legislative attempts to control the campaign finances, such as the Federal Election Campaign Act of 1974 and the Bipartisan Campaign Reform Act (BCRA, also known as the McCain-Feingold Act) of 2002. The goal of these reforms is to control the influence of the wealthy by imposing contribution limits and financial disclosure. The majority of the direct contributions to the campaign committee come from individual contributions and Political Action Committees (PAC).[8][9] Current campaign finance law requires candidates for federal-level positions to identify all PACs and individuals who give more than a certain amount in one election cycle. The Federal Election Commission (FEC), an independent federal agency, keeps all of this information publicly available.

These top-down efforts, however, are not sufficient to eliminate the influence of money in political. Although the regulations make it more difficult for large-scale donors to concentrate their contributions, these rules can be circumvented in a number of ways. Because money has been such a decisive factor in recent years, grass-root organizations such as Consumer Watchdog[10], MapLight[11], and the Center for Responsive Politics[12] continuously disambiguate the obfuscated contributions in an effort to keep citizens apprised of the financial sources influencing different

[6] http://www.opensecrets.org/overview/

[7] These numbers are not adjusted for inflation. According to Consumer Price Index program from The Bureau of Labor Statistics http://www.bls.gov/cpi/, $1,000,000 U.S. dollars in 2000 is equivalent to $1,356,004.65 in 2012.

[8] http://www.consumerwatchdog.org/

[9] For a comprehensive list of the way to donate to political campaigns, see http://www.fec.gov/answers_general.shtml#How_much_can_I_contribute

[10] http://www.consumerwatchdog.org/

[11] http://maplight.org/

[12] http://www.opensecrets.org/

politicians.

Keeping abreast with the public concern, campaign finance is an active area of research in political science. There are extensive theoretical and empirical works on the subject. Some view the relation between interest groups and political contribution as quid quo pro, with the contribution essentially being the reward for the legislative service and advocacy (Denzau and Munger, 1986). Others see contribution as ideological affinity, wherein the contribution is viewed as the contributor's support for the ideology without there being necessarily an expectation of favors in return (Austen-Smith, 1987; Poole et al., 1987; McCarty and Poole, 1998). The fundamental questions here are: how exactly does the money influence the politics, and how do interest groups decide to which candidates to give?

3.2.3 Why text? Why "Text as Data"?

We anticipate that our prediction tasks are particularly useful for those who wish to monitor legislative politics. The first model identifies the bills most likely to be successful at a very early stage, while the second model connects the language in the public speech to the underlying financial influence from, or affinity to, the special interest groups. Beyond prediction capacity, the models can be used to gain further insights into the subjects through inspection of the models.

Another scholastic motivation in this study is to address a broader question on *how text can be made useful* in examining complex political processes. Rather surprisingly, in contemporary political science, textual evidences are rarely used in their quantitative analysis. Considering how politics, and lawmaking in particular, regularly produce vast amount of written artifacts, this particular omission calls for some reflection. Often mentioned issues are the difficulty (and inadequacy) in the large scale text processing and the cost of human annotation and curation. In comparison, many representative works in this field draw heavily from non-textual data such as personal or institutional attributes. The best known such resources perhaps are the roll call floor votes, which have been studied extensively (Poole and Rosenthal, 1991, 1985; Cox and Poole, 2002; Jackman, 1991; Clinton et al., 2004).

We believe that there is a great deal of untapped power in data-driven text analysis in politics. As we alluded to in the first chapter, many scholars in the field have expressed this sentiment in recent years. Literature points out that textual data is potentially useful in analyzing subjects with poorly curated records (Grimmer and Stewart, 2013). The congressional committee system, where there are no roll call records, is one example of such a situation. (Judicial systems or state level legislatures are perhaps other examples.) Another important factor is the advent of social networking and electronic archiving. Larger and larger quantities of textual information are available every day, further adding incentives toward the development of an efficient and economical text-driven research paradigm. One of our aims is to contribute to this important trend in political studies by making a clear case of text's effectiveness. We review a few representative works from this area of political science in Section 3.5.

Cong.	Maj.	Introduced			Survival Rate (%)		
		Total	Rep.	Dem.	Total	Rep.	Dem.
103	Dem.	5,311	1,856	3,455	11.7	3.4	16.2
104	Rep.	4,345	2,426	1,919	13.7	19.7	6.1
105	Rep.	4,875	2,796	2,079	13.2	19.0	5.4
106	Rep.	5,682	3,299	2,383	15.1	20.9	7.0
107	Rep.	5,768	3,104	2,664	12.1	17.5	5.8
108	Rep.	5,432	2,915	2,517	14.0	21.0	5.9
109	Rep.	6,437	3,652	2,785	11.8	16.9	5.1
110	Dem.	7,341	2,668	4,673	14.5	8.5	18.0
111	Dem.	6,571	1,949	4,622	12.6	8.1	14.5
Total		51,762	24,665	2,7097	13.2	15.9	10.7

Table 3.1: Count of introduced bills per Congress, along with survival rate, and breakdown by the bill sponsor's party affiliation. Note that the probability of survival increases by a factor of 2–5 when the sponsor is in the majority party. Horizontal lines delineate presidential administrations (Clinton, Bush, and Obama).

3.3 Predicting Bill Survival

In this section we present our first prediction task in Congress: prediction of bills' survival through the congressional committee system. As we noted, the committee system is one of the active areas of research in political science. Although the lack of curated data resources has been an obstacle in data-driven inquiries, some of the key studies were done using various public records. The difficulties in data collection in these studies vary, but the burden of curation often falls on the shoulders of individual researchers. Naturally, the most readily available evidence is the textual contents of the bills, the written artifacts invariably present in all cases. In this work we will show how this resource can be made useful for the data-driven analysis.

To be sure, we are not aiming to prove that text has the superior explanatory power in this complex process. On the contrary, we believe that textual evidence is truly useful when it works with the informed insights from domain specialists. Our interest therefore is to examine where text complements the easy-to-obtain metadata and compensates the more difficult to come by metadata. This particular emphasis in our inquiry reflects on our choice of prediction models for this task (Section 3.3.2), as well as the setup of our empirical studies, starting with the compilation of our corpus (Section 3.3.1); we will later use this corpus in our experiments and exploratory analysis sections.

3.3.1 Data: Congressional Bill Corpus

Since our purpose is to build a model which leverages *both* the expert coding of legislative insights and the textual contents of bills, our first task is to align the existing bills' metadata to the bills' textual contents. Although several congressional bill corpora exist, there is no suitable corpus at the time of this work which aligns the bills' text with the bills' metadata. For this reason we created a new corpus to support our data driven research: Our bill data consist of the text of all bills introduced in the U.S. House of Representatives from the 103rd to the 111th Congresses (1/3/1993 to 1/3/2011). We consider only the version of the bill as originally introduced.[13] In our corpus each bill is matched up with the body of texts, title, committee assignments, and a binary value indicating whether the bill is referred back to the full Congress (i.e. recommended) from the committee or not. All our text data were downloaded directly from the Library of Congress's THOMAS website.[14] We also extracted several meta contextual data, such as sponsors name or party affiliation, from each bills summary page at THOMAS.

Along with the bill data, we gathered information on the sponsors and committees. The House Clerk office is in charge of keeping committee membership information, however, since the office does not maintain publicly accessible online resource, we obtained our committee information from Charles Stewart's resource at MIT. [15] Additional sponsor and bill information (e.g sponsor party affiliation or bill topic) was obtained from E. Scott Adler and John Wilkersons Congressional Bills Project at the University of Washington.[16]

There were a total of 51,762 bills introduced in the House during this period. A total of 6,828 of them have passed the committee and progressed further in the legislative process. See Table 3.1 for the breakdown by congress and sponsor party. The average rate for a bill passing committee is 13.2%. The best passing rate was the 106th congress during the Clinton administration in 1998 to 2000, with 15.1% of bills passing through committee.

The data is available from http://www.ark.cs.cmu.edu/bill-data/. We would like to note that recent government initiatives in transparency[17] could result in more useful resources in the public domain. We anticipate that this trend will affect the evolution of our congressional bill data in coming years.

Exploratory Analysis of the Bill Corpus

In Table 3.1, we break down the bills by the sponsor's party affiliation to illustrate one of the legislative cues in committee politics. Since the committee membership quota exactly reflects the party division in the chamber (and the committee chairs are always appointed from the majority party), bills which are sponsored by a majority party member must have higher chances of survival. Our corpus shows that this commonsense supposition is to some extent true. The sponsors party affiliation therefore should be a good basis for predicting bill survival. Accordingly, the probability of survival increases by a factor of 2 to 5 when the sponsor is in the majority party. At the same time, the same statistics suggest that this feature is far from decisive. One factor is that the majority party MCs tend to propose more bills than minority party MCs. In fact, there have been more failed bills proposed by the majority party than by the minority party in absolute number (Table 3.2). These statistics indicate that committee decisions were not dictated by strictly black-and-white party control. What if the main sponsors are themselves in the referred

[13] A bill's contents, and sometimes even its title, can change significantly.

[14] http://thomas.loc.gov/home/thomas.php

[15] http://web.mit.edu/17.251/www/data_page.html

[16] http://congressionalbills.org/

[17] http://www.whitehouse.gov/the_press_office/TransparencyandOpenGovernment

	Sponsor in majority	Sponsor in minority
Survive	5,543 (10.7)	1,285 (2.5)
Die	25,435 (49.1)	19,499 (37.7)

Table 3.2: Total number of survived bills, by party affiliations. Numbers in the parenthesis are the percentage in all the data.

committees? Many successful bills are in fact sponsored by committee members, but this fact does not clearly separate the survivors from the casualties (Table 3.3).

	Sponsor in the committee	Sponsor not in the committee
Survive	4,247 (8.2)	2,581 (5.0)
Die	16,916 (32.7)	28,018 (54.1)

Table 3.3: Bill survival rate by sponsors committee affiliation.

With political science collaborator John Wilkerson, we identified several other bill attributes which capture some commonsense knowledge in this domain. Although none of these attributes clearly separate the data, most of them do appear more frequently with the successful bills than with the unsuccessful bills. Later in the chapter, we will discuss what non-textual bill attributes are incorporated into our survival prediction model. (See further discussion in Section 3.3.3.) Presently we will discuss our machine learning approach, and how these attributes are incorporated into a prediction model.

3.3.2 Proposed Approach: Discriminative Log-Linear Model

In previous chapter, we employed a generative approach to our predictive modeling. For the task discussed here, we use much different technique, a feature-based **discriminative model**. The specific type of discriminative model we employ is a log-linear (or sometimes called a MaxEnt model (Berger et al., 1996)), where the target classification function is derived from the conditional log-linear distribution over the response variables (Murphy, 2012; Smith, 2011; Friedman et al., 2009). The reason for the switch is pragmatic: generative models, though expressive and intuitive in many respects, are not particularly flexible in exploiting heterogeneous, possibly correlated, evidences. Feature based models, on the other hand, allow us to explore a variety of attributes with very little change in training and inference procedure. Given the emphasis of our research question, and the type of resources we have at our disposal, we believe our inquiry to be best conducted as experiments on feature engineering.

Our blog volume prediction model (Topic-Poisson model) from Chapter 2 takes a textual input (blog post) and outputs a prediction as to the comment volume it will receive in the future. The predictive volume is assumed to be a count (integer) value, and is modeled as a sample from a mixture of Poisson distributions. Although we developed the Topic-Poisson model for the purpose of blog popularity prediction, it can very well be applied to other types of text-driven prediction in which the target response is approximated as a single integer. If we chose to postulate the response value in our bill prediction task to be some integer value (perhaps the *count* of positive votes) rather than a binary indicator, the Topic-Poisson model can be used straightforwardly.[18] Alternatively, we can extend the Topic-Poisson model to handle a binary indicator response. (Application of sLDA is another possible idea, although we abandoned this idea early on because of sLDA's poor predictive performance on the unbalanced data set considered in the last chapter.)

Although we do not exclude these directions from the future possibilities, we chose the discriminative approach here since we believe it is the more apt approach for the current task. Similar to our bill survival task, (Gerrish and Blei, 2011, 2012) developed a topic model which models the generation of roll call votes (represented as a vector of indicator variables, each corresponding to one vote from one legislator) alongside the bill text, then applied the model to the roll call outcome prediction task. This model is not quite applicable to our task since the model assumes that for all bills the individual members' votes are known at training time.[19] Committee deliberations take place separately from floor voting, and the individuals' votes are usually not disclosed. More importantly, (Gerrish and Blei, 2011) reported that their best prediction results are comparable to a text regression model, a log-liner model which uses only text-based features. This result alone place the discriminative log-linear model as a strong alternative to the generative topic model. Moreover, we observed that log-linear model has critical engineering advantages particularly desirable for our task.

Our discussion with political domain specialists convinced us that utilizing bill metadata information alongside the text is important to achieve competitive prediction performance. Feature-based approaches (such as log-linear models), unlike generative topic models, offer a straightforward procedure to include arbitrary features into the predictive model, allowing various features to be combined in one probability distribution through feature functions. This considerably lessens the engineering overhead in experimenting with various, heterogeneous, bill attributes.[20] Our experimental results (Section 3.3.5) largely support this observation.

In the following sections, we will first review the conditional log-linear model and then introduce our features. As we will see, each set of features is motivated by different insights on how a committee makes its decision.

Technical Review of Log-linear Model

Log-linear model is a family of distributions over discrete random variables, where the probability density function is proportional to the exponential transformation of feature functions. In this task our interest is the *conditional* log-linear model:

[18] We of course need to make necessary transformations to make sure the target value falls into the realistic range.

[19] This assumption is only true for a few legislative actions such as final floor voting ("Roll Calls"). In our work, we focus not on the floor voting but bill survival in the committee system.

[20] In the generative approach, each bill attribute would be treated as a single distribution, and therefore different choice of attributes may require substantial changes to training and inference algorithms.

$$p_{\mathbf{w}}(y \mid x) = \frac{\exp \mathbf{w}^\top \mathbf{f}(x, y)}{\sum_{y' \in Y} \exp \mathbf{w}^\top \mathbf{f}(x, y')} \tag{3.1}$$

Where y is the response variable, and x is the evidence, or independent variable. $\mathbf{w}$ are "weight" parameters associated with each feature in the feature vector $\mathbf{f}(x, y)$.

Let x be a random variable associated with a bill, and let $\mathbf{f}$ be a feature vector function that encodes observable features of the bill. Let y be a binary random variable corresponding to bill survival ($y = 1$) or death ($y = 0$). Let's assume the distribution over y is a conditional log-linear model given the features of x. Since y is a binary variable, this can be modeled as a logistic regression, where:

$$p_{\mathbf{w}}(y = 1 \mid x) = \frac{\exp \mathbf{w}^\top \mathbf{f}(x)}{1 + \exp \mathbf{w}^\top \mathbf{f}(x)}$$

This leads to the predictive rule:

$$\hat{y}(x) = \begin{cases} 1 & \text{if } \mathbf{w}^\top \mathbf{f}(x) > b \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

where b is a bias parameter which fires for all examples.

We train the model by maximizing log-likelihood plus a sparsity-inducing log-prior that encourages many weights to go to zero:

$$\max_{\mathbf{w}} \sum_{i=1}^{N} \log p_{\mathbf{w}}(y_i \mid x_i) - \lambda \|\mathbf{w}\|_1 \tag{3.3}$$

where i indexes training examples (specifically, each training instance is a bill referred to a single committee). The second term is an L_1 norm, equivalent to a Laplacian prior on the weights (Friedman et al., 2009, 2010). Training, or fitting of the model parameters, requires searching for the set of parameter values which yiels the maximum value of this objective function with respect to the training data. Several optimization techniques are applicable here. For the experiments reported in this paper we use a variation of quasi-Newton optimization, a second-order gradient method. In specific, we use limited-memory BFGS (L-BFGS) algorithm as described in (Liu and Nocedal, 1989). The actual implementation of the algorithm is from `http://www.chokkan.org/software/liblbfgs/`, which includes the extension to orthant-wise limited-memory quasi-Newton (OWL-QN) (Andrew and Gao, 2007) which accommodates the L_1 regularized objective function.

L-BFGS is a general optimization algorithm which works on convex, differentiable objective functions. The application of the method to the specific problem requires specifying the analytical form of the objective function (Equation 3.3) and its gradient. The value of λ, which controls sparsity, is chosen on a held-out subset of the training data.

Linear models like this one are attractive because they are intelligible. The magnitude of a weight indicates a feature's importance in the prediction. We note that the L_1 regularizer is not ideal for identifying predictive features. When two features are strongly correlated, it tends to choose one of them to include in the model and eliminate the other, despite the fact that they are both predictive. It is therefore important to remember that a weight of zero does not imply that the corresponding feature is unimportant. We chose to cope with this potential elimination of good features so that our models would be compact and easily interpretable. In Section 3.3.3 we devise "impact" score of feature to further help the interpretability of our models.

3.3.3 Baseline: Legislative Metadata Features

Within the analytical framework of the log-linear model, various types of evidences can be incorporated as conditioning features. Since our chief concern is how the text would make a difference, we will first craft the prediction model without the textual information as a baseline.

Baseline features: Legislative metadata

In American politics, the survival and death of bills are often explained in terms of expertise, entrepreneurship, and procedural control, which are manifest in committee membership, sponsor attributes, and majority party affiliation. We therefore begin with a strong baseline that includes features encoding many expected effects on bill success. With close collaboration with political scientists we devised the following eleven classes of features (yielding 3,731 instantiated features). No features below concerns the texts in the title, nor in the body of the bill. All the basic features are binary functions:

1. For each party $p \in \{$Republican, Democrat, Independent$\}$, is the bill's sponsor affiliated with p?

2. Is the bill's sponsor in the same party as the committee chair? Equivalently, is the bill's sponsor in the majority party of the House?

3. Is the bill's sponsor a member of the committee?

4. Is the bill's sponsor a *majority* member of the committee? (This feature conjoins 2 and 3.)

5. Is the bill's sponsor the chairman of the committee?

6. For each House member $j \in \{$Ackerman, Adams, Adarhold ...$\}$, did j sponsor the bill?

7. For each House member j, is the bill sponsored by j and referred to a committee he chairs? (This feature conjoins 5 and 6.)

8. For each House member j, is the bill sponsored by j and is j in the same party as the committee chair? (This feature conjoins 2 and 6.)

9. For each state s, is the bill's sponsor from s?

10. For each month m, is the bill introduced during m?

11. For each congressional year $v \in \{1, 2\}$, is the bill introduced during the vth year of the (two-year) Congress?

Note that these listed here are not all the features we have considered. Several suggestions from political science literature (such as a bill's subcommittee referral status) are omitted due to their difficulty in data curation. Also, quite a few other features were eliminated during the preliminary examination. One surprisingly detrimental feature, omitted here, was the identity of the referred committee. This was surprising since bill success rates vary greatly across committees (e.g., the Appropriations committee recommends about half of the referred bills, while the Ways and Means committee recommends only 7 percent). We suspect that this feature simply has poor generalization ability across different Congresses. (In §3.3.4 we will consider preferences of *individuals* on committees, based on text, which appears to benefit predictive performance.)

Performance

Model		Error (%)		
		109th	110th	111th
most frequent class		11.8	14.5	12.6
§3.3.3	baseline (no text)	11.1	13.9	11.8

Table 3.4: Results on two bill survival prediction, using the expert-informed meta data features. (Including no text-driven feautures.) Comparing to our simple lower-bound, the baseline model reduce error by 0.7, 0.6, and 0.8 %.

	Prec.	Recall	F1
baseline (111th)	63.3	14.4	23.4

Table 3.5: Precision and recall on the predictions on 111th Congress.

Using the above metadata features (3,731 total instantiated features), we train our conditional log-linear model with the training portion of our data set, and then evaluate the model's prediction accuracy against the held-out test set. We use three different training-test splits. In one experiment, we use the 103rd to 110th Congresses (45,191 bills) as the training set and used the 111th Congress (6,571 instances) as the test set. We repeat the same experiment for the 110th and 109th Congresses, each time using the preceding congress as the training set (so that in testing the 110th Congress, bills from the 103rd to 109th Congresses were used for model training, and for the 109th Congress, bills from the 103rd to 108th. Table 3.4 shows the prediction error of our baseline model. Since our data set is highly skewed toward the positive (bill survival) class, we also show here the precision and recall measurements (Table 3.5).

Historically, an introduced bill has a 12 to 15 percent chance of survival. This means that a lazy (but rational) prediction scheme which invariably predicts a negative outcome performs at about 85 to 88 percent of accuracy, or 15 percent error. Note that, even though 12 percent error is a seemingly small number, the precision and recall scores of the model indicate that the model is missing many positive examples. Considering the 111th Congress, this most-frequent class predictor achieves an error rate of 12.6 percent. In comparison, our basic metadata model

Bill Survival	
sponsor is in the majority party (2)	0.525
sponsor is in the majority party and on the committee (4)	0.233
sponsor is a Democrat (1)	0.135
sponsor is on the committee (3)	0.108
bill introduced in year 1 (11)	0.098
sponsor is the referred committee's chair (5)	0.073
sponsor is a Republican (1)	0.069
Bill Death	
bill's sponsor is from NY (9)	-0.036
sponsor is Ron Paul (Rep., TX) (6)	-0.023
bill introduced in December (10)	-0.018
sponsor is Bob Filner (Dem., CA) (6)	-0.013

Table 3.6: Baseline model: high-impact features associated with each outcome and their impact scores.

achieved an 11.8 percent error rate, a small but statistically significant improvement (McNemars test, $p < 0.0001$). The error reductions in other Congresses are also significant.

"Impact" of features

When inspecting linear models, considering feature weights can be misleading, since (even with regularization) large weights does not necessarily correspond to large effects in the training or test data. In the blog volume prediction experiments (Chapter 2), we have attempted to interpret the feature weights of our baseline linear models (Section 2.6.3), and noticed that the weights from these models are much less understandable than the topics from the topic models. Upon further inspection, we realized that the difficulty is in part due to the fact that high weight features do not necessarily appear frequently in the held-out test set. To circumvent this problem, we employ the *impact* score, originally proposed by Brendan O'Connor, for the inspection of model feature weights. In our experience this metric is intuitive, easy to compute, and qualitatively better than the simple ranking by absolute feature weights. The score is computed as follows. For each feature on the final decision for class y, defined for feature j as:

$$\frac{w_j}{N}\sum_{i=1}^{N} f_j(x_i) \tag{3.4}$$

where i indexes test examples (of which there are N).

Impact is the average effect of a feature on the model's score for class y. Note that it is not affected by the true label for an example. Impact is additive, which allows us to measure and compare the influence of sets of features *within a model* on model predictions. Impact is not, however, directly comparable *across* models.

The highest impact features from the 111th Congress are shown in Table 3.6. Unsurprisingly, the model's predictions are strongly influenced (toward survival) when a bill is sponsored by someone who is on the committee and/or in the majority party. Feature 2, the sponsor being on the committee, accounted for nearly 27% of all (absolute) impact, followed by the member-specific features (6–8, 19%), the sponsor being in the majority and on the committee (4, 12%), and the party of the sponsor (1, 10%).

We should note that main purpose of impact score is exploration, not to draw any strong causation argument from our experimental results. Our purpose is simply to understand our empirical results. We also note that impact as a tool for interpreting models has some drawbacks. If a large portion of bills in the test set happen to have a particular feature, that feature may have a high impact score for the dominant class (death). This probably explains the presence of "sponsor is a Democrat" in Table 3.6; Democrats led the 111th Congress, and introduced more bills, most of which died.

3.3.4 Text-Driven Feature Engineering

We turn next to the bills' textual contents to augment the predictive power of our model. We propose four sets of text-driven features. From a computational perspective, each approach corresponds to the same algorithm implemented with a different set of feature functions. From a political science perspective, different features corresponds to alternative explanations on what drives committees' decisions. We will see that, in some cases, text is used to approximately realize such insights when human annotation is costly or impossible.

Text Feature 1: Functional Bill Categories

An important insight from political science is that bills' substance (contents) can be categorized in general ways that are related to their likelihood of success (Adler and Wilkerson, 2005). For example, bills which commemorate individuals or name buildings are considered easy to pass bills since there tend to have no political objections. Some observe that time-sensitive (or "urgent") cyclical bills such as budget or appropriation tend to advance smoothly because they often are well facilitated.

In (Adler and Wilkerson, 2005), the authors distinguish congressional bills into several functions that capture bills that are on the extremes in terms of the importance and/or urgency of the issue addressed. As a part of their empirical validation the authors individually inspected the contents of the House bills between 101st to 105th, and labeled them using the following categories:

- bills addressing **trivial** issues, such as those naming a federal building or facility or coining commemorative medals;
- bills that make **technical** changes to existing laws, usually at the request of the executive agency responsible for its implementation;
- bills addressing **recurring** issues, such as annual appropriations or more sporadic reauthorizations of expiring federal programs or laws; and

- bills addressing **important**, urgent issues, such as bills introduced in response to the 9/11 terrorist attacks or a sharp spike in oil prices.

We expect that these categories, if known a priori, can help explain which bills survive committees; therefore they could improve our model's predictive performance. The problem is that such annotation is quite expensive to repeat, since it requires substantial expert knowledge and close reading of the bills' contents. We propose to overcome this obstacle by labeling bills with *soft categories* from text classifiers tailored to distinguish the above categories.

Soft Labeling Bills with Text Classifiers

Out of the set previously annotated by the experts, we put aside the portion that overlaps with our bill collection (103rd–105th). Of these 14,528 bills, 1,580 were labeled as trivial, 119 as technical, 972 as recurring, and 1,508 as important. To categorize the bills in the other Congresses of our dataset, we trained binary logistic regression models to label bills with each of the three most frequent bill types (trivial, recurring, and important) based on unigram features of the body of bill text. (There is some overlap among categories in the annotated data, so we opted for three binary classifiers rather than multi-class.)

In a ten-fold cross-validated experiment, this model averaged 83% accuracy across the prediction tasks. We used the manually annotated labels for the bills in the 103rd–105th Congresses; for other bills, we calculated each model's probability that the bill belonged to the target category. These values are used to define binary indicators for each classifier's probability regions: $[0, 0.3)$; $[0.3, 0.4)$; $[0.4, 0.5)$; $[0.5, 1.0]$. For each of the three labels, we included two classifiers trained with different hyperparameter settings, giving a total of 24 additional features. In the experiment section (Section 3.3.5 we will refer these features as **Bill Functions**.

Text Feature 2: Bill Topics

Note that in above we are not classifying the bill by its issues or theme, which usually what "text classification" means without any modifiers. One question is whether the plain ordinary topic categories, as features for the bill survival prediction, would compensate (i.e. explain away) special functional categorization that we worked so hard to approximate.

The question is worth asking here because the thematic categorization can be done in purely unsupervised fashion using word co-occurrence statistics. LDA (and topic modeling in general), which we employed for Blogosphere prediction tasks (Chapter 2), is commonly used for this purpose. For the second set of text features, we propose to use the thematic topics induced from the bill texts.

Using LDA algorithm (Section 2.4.1) and training portion of the data, we fit a topic model with topic size (K) = 50. We then infer the topic distribution for each bill in the validation set using the standard Gibbs sampling based inference procedure. Let θ_d be the topic distribution for bill $d \in D_{test}$ We compute the posterior distribution over θ_d as described in Equation 2.5.1. Given this distribution, we label the bills with binary function indicating whether each topic's probability mass is 1) in $[0.75, 1.0]$ region, and 2) in $[0.50, 0.75]$ region. (We ignore the topics with $< .5$ probability mass.) This results in total of 100 additional features at the end. In the experiment section Section 3.3.5 we will refer these features as **Bill Topics**.

Text Feature 3: Approximating Committee Votes

We next consider a different view of text: as a means of profiling the preferences and agendas of legislators. We start with a hypothesis that committees operate similarly to the legislature as a whole, Therefore, the more the individual find the bill agreeable with her preference or agenda, the better the chance of survival. Of course, the real story is not that clear-cut; deliberation and compromise may take place before such a vote, but we focus on the basic for the simplicity sake.

Recall that the MCs vote on the floor, unlike the committee actions, are recorded votes. This data (roll call records) are frequently used in political science to estimate *spatial* models of legislators and legislation (Poole and Rosenthal, 1991, 1985; Cox and Poole, 2002; Jackman, 1991; Clinton et al., 2004). These models help visualize politics in terms of intuitive, low-dimensional spaces which often correspond closely to our intuitions about "left" and "right" in American politics. Recently, (Gerrish and Blei, 2011) showed how such models could naturally be augmented with models of text, and also could applied to predictions.

Note that all above models are based on the *observed* votes. Meanwhile, the votes in the committee are *hidden*. We propose to approximate the individual vote. Our approach is to construct a *proxy vote*, or an estimate of the votes by committee members, from the textual substance of the bill. We consider three variants, each based on the same estimate of individual committee members' votes, but which differ in the assumption on whose votes matter (and whose opinion would be ignored):

- Only the committee chairman's vote matters.
- Only majority-party committee members vote.
- All committee members vote.

We will compare these three versions of the proxy vote feature experimentally, but abstractly they can all be defined the same way.

Estimating Proxy Votes from Bill Texts

Let $\mathcal{C}$ denote the set of committee members who can vote on a bill x. We define the proxy vote to be:

$$\frac{1}{|\mathcal{C}|}\sum_{j\in\mathcal{C}}\mathbb{E}[V_{j,x}] \tag{3.5}$$

We treat the vote by representative j on bill x as a binary random variable $V_{j,x}$ corresponding to a vote for (1) or against (0) the bill. (If x is referred to more than one committee, we average the above feature across committees.) Naturally, we do not observe $V_{j,x}$; instead we estimate its expected value, which will be between 0 and 1. Note that, by linearity of expectation, the sum in equation 3.5 is the expected value of the number of committee members who "voted" for the bill; dividing by $|\mathcal{C}|$ gives a value that, if our estimates are correct, should be close to 1 when the bill is likely to be favored by the committee and 0 when it is likely to be disfavored.

The problem, of course, is how to figure out $\mathbb{E}[V_{j,x}]$. To estimate $\mathbb{E}[V_{j,x}]$, we use a simple probabilistic model of $V_{j,x}$ given the bill x and the past voting record of representative j.[21] Let $\mathcal{R}_j$ be a set of bills that representative j has publicly voted on, on the floor of the House, in the past.[22] For $x' \in \mathcal{R}_j$, let $V_{j,x}$ be 1 if j voted for the bill and 0 if j voted against it. Further, define a similarity measure between bills; here we use cosine similarity of two bills' tfidf vectors.[23] We denote by $sim(x, x')$ the similarity of bills x and x'.

The probabilistic model is as follows. First, the representative selects a bill he has voted on previously; he is likely to choose a bill that is similar to x. More formally, given representative j and bill x, randomly choose a bill X' from $\mathcal{R}_j$ according to:

$$p(X' = x' \mid j, x) = \frac{\exp(\mathrm{Sim}(x, x'))}{\sum_{x'' \in \mathcal{R}_j} \exp(\mathrm{Sim}(x, x''))}$$

An attractive property of this distribution is that it has no parameters to estimate; it is defined entirely by the text of bills in $\mathcal{R}_j$. Second, the representative votes on x identically to how he voted on X'. Formally, let $V_{j,x} = V_{j,x'}$, which is observed. The above model gives a closed form for the expectation of $V_{j,x}$:

$$\mathbb{E}[V_{j,x}] = \sum_{x' \in \mathcal{R}_j} p(X' = x' \mid j, x) \cdot V_{j,x'} \tag{3.6}$$

In addition to the proxy vote score in Equation 3.5, we calculate a similar expected vote based on "nay" votes, and consider a second score that is the ratio of the "yea" proxy vote to the "nay" proxy vote. Both of these scores are continuous values; we quantize them into bins, giving 141 features in total. In the experiment section Section 3.3.5 we will refer these features as **Proxy Vote**.[24]

Text Feature 4: Direct Use of Content as Bag of Words

For the fourth set of text features, we create features directly using unigram and bigram terms. Here the feature function is simply (binary) indicator function representing the presence or absence of each term in the bill. We determine the vocabulary of these terms in the following way; After processing the bill text with the light text normalization,[25] we list all the unigrams

[21] We note that the observable roll call votes on the floor of the U.S. House consist of a very different sample of bills than those we consider in this study; indeed, votes on the floor correspond to bills that *survived* committee. We leave attempts to characterize and control for this bias to future work.

[22] To simplify matters, we use all bills from the training period that j has voted on. (x itself is naturally never in $\mathcal{R}$.) For future predictions (on the test set), these are all in the past, but in the training set they may include bills that come later than a given training example.

[23] We first eliminated punctutations and numbers from the texts, then removed unigrams which occured in more than 75% or less than 0.05% of the training documents. Tfidf scores were calculated based on the result.

[24] We discretized the continuous values by 0.01 increment for proxy vote score, and 0.1 increment for proxy vote rate scores. We further combined outlier bins (one for extremely large values, one for extremely small values).

[25] Punctuation marks are removed from the text, and numbers are collapsed into single indicator. We did not apply stemming or lemmatization.

Mode	Error (%)		
	109th	110th	111th
Most frequent class	11.8	14.5	12.6
Baseline (no text)	11.1	13.9	11.8
Bill categories	10.9	13.6	11.7
Bill Topics	10.2	12.9	11.0
Proxy vote, all three	9.9	12.7	10.9
Unigram+bigram	8.9	10.6	9.8
Full model	8.9	10.9	9.6

Table 3.7: Results on bill survival prediction. Evaluated on three different Congress. Each model's improvement over the baseline is significant.

appeared in the the body of the bill and the unigram and bigram in the title of the bills. We then filtered out the terms appearing in fewer than 0.5 percent and more than 30 percent of training documents. This results in many feature terms from the bill body, and many feature terms from the bill title (24,515 features total). In the experiment section (Section 3.3.5) we will refer these features as **Unigrams** and **Bigrams**, or **Raw Text** when referring both types together.

Excepting this feature, all our text features were derived features. The derivation procedure is based on different ideas on how committees make decisions. Although the raw ingredients are the same (bill texts) in all cases, deriving the feature this way we impose some assumption on how committee function and how bill substances relate to such dynamics.[26] This is a nice setup in one sense since at the end we will learn something about the underlying hypothesis as well as the prediction performance. The down side is that, as in any feature selection scheme, the process can be lossy; we impose our assumption by instantiating some ideas and leaving out others. In the last round of feature engineering, we make our assumption very, very broad. The only hypothesis here is that committees (collectively) make decisions by considering the "contents "of bills, or the content of the bill somehow bias the outcome.

Simplifying the textual contents to a bag of words necessarily lose some linguistic information. We point out that this approach, unigram and bigram features, is a staple in sentiment and opinion mining (such as (Pang and Lee, 2004)) and text-driven prediction (such as (Kogan et al., 2009)). Naturally, this last set of features are much more difficult for interpretation. We use sparsity inducing L_1 regularization to reduce the feature size. In later section (Section 3.3.6) we examine the learned feature weights carefully with help from domain specialists.

3.3.5 Experimental Results

In Section 3.3.3, we implemented our conditional log-linear model with the meta-data features using 103rd to 110th Congresses (45,191 bills). On 111th Congress, this model achieved 11.8

[26] Functional bill category and Proxy votes can be understood from a machine learning perspective as task-specific dimensionality reduction methods on the words.

Model	Err (%)	F+	F–	T+	Prec.	Recall	F1
Most frequent class	12.6	0	828	0	–	–	–
Baseline (no text)	11.8	69	709	119	63.3	14.4	23.4
Bill categories	11.7	52	716	112	68.3	13.5	22.6
Bill topics	11.0	137	586	242	63.9	29.2	40.0
Proxy vote, chair only	10.8	111	596	232	67.6	28.0	39.6
Proxy vote, majority	11.3	134	606	222	62.4	26.8	37.5
Proxy vote, whole	10.9	123	596	232	65.4	28.0	39.2
Proxy vote, all three	10.9	110	606	222	66.9	26.8	38.3
Unigram+bigram	9.8	106	541	287	73.0	34.7	47.0
Unigram+bigram only	10.2	136	535	293	68.3	35.4	46.6
Full model (all above)	9.6	120	514	314	72.4	37.9	49.8

Table 3.8: Key experimental results; models were trained on the 103rd–110th Congresses and tested on the 111th. Baseline features are included in each model listed below the baseline. "T+" is the count of true positive examples. "F+" and "F-" is the count of false positive and false negative. "Prec." is precision. Each model's improvement on error reduction over the baseline is significant (McNemar's test, $p < 0.0001$ except bill categories, for which $p < 0.065$).

percent error rate (Table 3.4). In the subsequent experiments, we individually add text features to the baseline meta-data model and examine how they change the prediction error rate. We first each of the four types of text feature separately, then later use all the features at once to see the overall performance gain. All the experimental results referred in the discussion from this section is in Table 3.7, Table 3.8 and Table 3.9.

We use error (the inverse of accuracy) as our main performance metric since it considers the performance on both positive and negative classes. We opt for this metric for the current task since, unlike our previous binary prediction tasks (comment volume prediction from Chapter 2), negative class (bill death) is often as important as the positive class (bill survival). Error is also one of the most straightforward performance measurement on prediction tasks, therefore has a much intuitive appeal to social scientists who may not be familiar with computer science methodologies. As in the baseline experiments, we repeated the experiment using the 109th and 110th Congresses as test datasets (training only on bills prior to the test set) to avoid drawing conclusions based on a single, possibly idiosyncratic Congress (Table 3.7). The comparison shows that the error patterns are similar.

Table **??** gives more detailed analysis of the learned model from the experiments with 111th Congress. Since our data set is highly skewed toward the positive (bill survival) class, we include in this table precision and recall measurements (taking the bill survival as the "true" class), as well as true positive, false positive, and false negative counts. We also report some statistics on the size of learned models in Table 3.9.

Bill Function The results of the model with the bill functional category feature are shown in the second group in the Table 3.8. Including these features reduces the prediction error slightly but significantly relative to the baseline (just over 1% relative error reduction). We note that

Model	# Feats.	Size	Effective
Baseline (no text)	3,731	1,284	460
Bill categories	3,755	274	152
Bill topics	3,831	923	493
Proxy vote, chair only	3,780	1,111	425
Proxy vote, majority	3,777	526	254
Proxy vote, whole	3,777	1,131	433
Proxy vote, all three	3,872	305	178
Unigram+bigram	28,246	199	194
Unigram+bigram only	24,515	2,207	2,156
Full model (all above)	28,511	1,096	1,069

Table 3.9: Feature sizes of the models. "# Feats." is the total number of features available to the model; "Size" is the number of features with non-zero weights in the final selected sparse model; "Effective" is the number of features with non-zero impact (eq. 3.4) on test data.

preliminary investigations conjoining the bill category features with baseline features did not show any gains, although prior work by (Adler and Wilkerson, 2012) suggests that bill category interacts with the sponsor's identity.[27]

Considering the model's weights, the log-odds are most strongly influenced toward bill success by bills that seem "important" according to the classifiers. 55% of this model's features had non-zero impact on test-set predictions; compare this to only 36% of the baseline model's features.[28] Further, the 18 category features accounted for 66% of the total (absolute) impact of all features. Taken altogether, these observations suggest that bill category features are a more compact substitute for many of the baseline features,[29] but that they do not offer much additional predictive information beyond the baseline (error is only slightly reduced). It is also possible that our categories do not perfectly capture the perceptions of committees making decisions about bills. Refinement of the categories within the predictive framework we have laid out here is left to future research. Interestingly, the model achieved relatively high precision compared to other text based models.

Bill Topics The results of the model with the bill topic features are shown in the third group in Table 3.8. Though it is an unsupervised clustering technique, adding these features gives more predictive power than the bill functional category features. The model's precision and recall scores indicate that this error reduction is mostly due to the increase in the recall; an increase of more than 12 points, or more than twice as many correctly predicted bill survivals. In fact, the precision of the model is notably less than the bill functional category model. This is also reflected in the increased number of false positives (137 false positive examples, compare to 52 examples with the bill functional category model). Manual inspection of induced topics (not

[27] We leave a more careful exploration of this interaction in our framework to future work.

[28] Note that Ł$_1$ regularized models make global decisions about which features to include, so the new features influence which baseline features get non-zero weights. Comparing the absolute number of features in the final selected models is not meaningful, since it depends on the hyperparameter λ, which is tuned separately for each model.

[29] This substitutability is unsurprising in some scenarios; e.g., successful reauthorization bills are often sponsored by committee leadership.

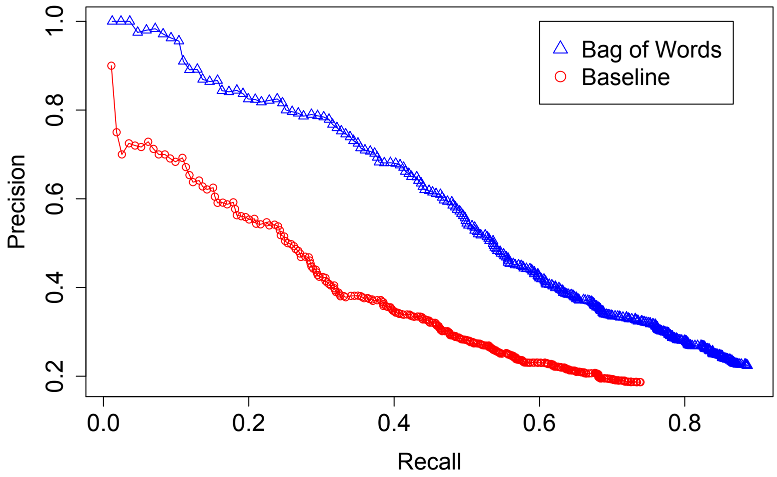

Figure 3.1: Precision-recall curve (survival is the target class) comparing the bag of words model to the baseline.

shown) does not reveal much correspondence to the bill functional category. There also seem to be a large number of redundant topics, most likely due to the topic size ($k = 50$).

Proxy Vote We show the results of adding the proxy vote feature in the fourth group in the Table 3.8. The four variations are 1) $\mathcal{C}$ defined to include the chair only, 2) Majority party members, 3) The full committee, and 4) All three sets of proxy vote features. All four models showed improvement over the baseline. Using the chairman-only committee (followed closely by whole committee and all three) turned out to be the best performing among them, with a 8% relative error reduction.

Nearly 58% of the features in the combined model had non-zero impact at test time, and 38% of total absolute impact was due to these features. Comparing the performance of these four models suggests that, as is widely believed in political science, the preferences of the committee chair are a major factor in which bills survive. Our results suggest further that committee decisions are better predicted by considering the preferences of the whole committee, not just those in the majority party.

Raw Text Perhaps unsurprisingly, this approach perform better than the other derived text features. Combined with baseline features, unigram features from text body, unigram and bigram features led to nearly 18% relative error reduction compared to the baseline and 9% relative to the best model above (Table 3.8). The model is very small (under 200 features), and 98% of the features in the model impacted test-time predictions. To further test this model's robustness, we measure the precision of the model at all possible recalls. (See Figure 3.1). The models' consistent improvement over the baseline on this metrics shows that its gain is not sensitive to the bias *b*. A key finding is that the bag of words model outperforms the bill categories and proxy vote models. This suggests that there is more information in the text contents than either the functional categories, thematic topics, or similarity to past bills.

Full Model Finally, we considered a model using all three kinds of text features. Shown in the last group in Table 3.8, this reduces error only 2% relative to the bag of words model. This leads us to believe that direct use of text captures most of what functional bill category and proxy vote features capture about bill success.

We also conducted the experiment with the model which include the raw text (unigram and bigram) feature, without the meta data features. The performance of this model is notably worse than model which uses both features. This results suggest that the text data is *complementalary* to the metadata features, that the text can capture something that the conventional meta data (at least the ones feasible without extensive data curation) can not.

3.3.6 Descriptive Aspects of the Models

Table 3.10 shows the terms with greatest impact. When predicting bills to survive, the model seems to focus on explanations for minor legislation. For example, *interior* and *resources* may indicate non-controversial local land transfer bills. In titles, *designate* and *located* have to do with naming federal buildings (e.g., post offices).

As for bills that die, the model appears to have captured two related facts about proposed legislation. One is that legislators often sponsor bills to express support or concern about an issue with little expectation that the bill will become a law. If such “position-taking” accounts for many of the bills proposed, then we would expect features with high impact toward failure predictions to relate to such issues. This would explain the terms *energy*, *security*, and *human* (if used in the context of human rights or human cloning). The second fact is that some bills die because committees ultimately bundle their contents into bigger bills. There are many such bills relating to tax policy (leading to the terms contained in the trigram *Internal Revenue Service*, the American tax collection agency) and *Social* Security policy (a collection of social welfare and social insurance programs), for example. The term *speaker* likely refers to the first ten bill numbers, which are “reserved for the speaker,” which actually implies that no bill was introduced. Our process for marking bills that survive (based on committee recommendation data) leaves these unmarked, hence they “died” in our gold-standard data. The experiments revealed this uninteresting anomaly.

3.4 Predicting Campaign Contributions

As noted earlier in this chapter, the consequence of the vast money required for the electoral campaigns, and the influence from the financially well endorsed sectors in politics is an important question among political scientists. To what extent can we measure or characterize the

Bill Survival			
Contents		Title	
resources	0.112	title as	0.052
ms	0.056	other purposes	0.041
authorization	0.053	for other	0.028
information	0.049	amended by	0.017
authorize	0.030	of the	0.017
march	0.029	for the	0.014
amounts	0.027	public	0.012
its	0.026	extend	0.011
administration	0.026	designate the	0.010
texas	0.024	as amended	0.009
interior	0.023	located	0.009
judiciary	0.021	relief	0.009

Bill Death			
Contents		Title	
percent	-0.074	internal	-0.058
revenue	-0.061	the internal	0.024
speaker	-0.050	revenue	-0.022
security	-0.037	prohibit	-0.020
energy	-0.037	internal revenue	-0.019
make	-0.030	the social	-0.018
require	-0.029	amend title	-0.016
human	-0.029	to provide	-0.015
concerned	-0.029	establish	-0.015
department	-0.027	SYMBOL to	-0.014
receive	-0.025	duty on	-0.013
armed	-0.024	revenue code	-0.013

Table 3.10: Full model: text terms with highest impact (eq. 3.4). Impact scores are not comparable across models, so for comparison, the impacts for the features from Table 3.6 here are, respectively: 0.534, 0.181, 10^{-4}, 0.196, 0.123, 0.063, 0.053; -0.011, 0, 0.003, 0.

nature of campaign contributors' influence on elected officials? In this section, we consider an empirical approach to this question by exploring the connection between the campaign contributions a member of Congress receives and his or her "public statements".

The basic question we address here is what influences a politician's public statements. One plausible explanation is that financial incentives from campaign contributors affect what congress persons say. We explore this idea through building of text-driven prediction models for campaign contribution profile, utilizing the large corpus of public *microblog message* statements by members of the U.S. Congress.

Microblogs, especially Twitter, have become an integral part of political campaigns, public outreach by politicians, and political discourse among citizens. Automatic analysis of microblog text has the potential to transform our understanding of public opinion (O'Connor et al., 2010a), communication between elected officials and their constituents (Golbeck et al., 2010), and information flow in society more generally (Lerman and Ghosh, 2010). As a tool of public outreach, twitter messaging communicates the voice of political candidates to the constituent; what needs to be fixed, what issues should be important, and what ought to be the nation's most urgent agenda. These are what the candidates chose to publicly defend, refute and appeal define their overt political stances. We conjecture here that they strongly relate to the candidates' financial incentives.

We begin with the assumption that the extent to which campaign contributions relate to politicians should be measurable in the *predictability* of those contributions, given the text. We therefore employ probabilistic modeling to infer associations between campaign contributions, as made available by the Federal Election Committee, and the text of tweets from members of Congress. Further, we show that judicious use of latent variables can help reveal linguistic cues associated with contributions from specific industries and interest groups. We like to draw the readers' attention to the fact that these campaign contribution data, although freely available to public, are difficult to use in data-driven analysis in their raw format. In this study we rely heavily on the manual annotation and disambiguation done by a watchdog organization. This manual annotation is quite labor intensive, and not easy to repeat for every important institutions (such as state or maniple level election).[30] Although the models we present here are trained with the federal level election data, the learned model from our work can potentially be useful in helping with these elections lacking curated annotations.

In the next sections, we formalize our idea into a text-driven forecasting task, then examine our model through empirical evaluations. Our basic question – the nature of politics and money – is a century old question; though the computational approach we take here is novel and to our knowledge there has been no previous such attempt. We revisit the topic modeling framework (which we cultivated in the previous chapter) in Section 3.4.2. The details of the generative model, and the way we derive the predictive system, differ much from the previous task. We explain the important details in Section 3.4.3. Presently, we describe our data set, composed anew for the present work.

[30] Another complication in the state level election is that each state has different donation and disclosure regulations.

3.4.1 Data: Congressional Tweet Corpus

Our dataset consists of two parts: Microblog messages (tweets) from the accounts officially associated with members of Congress (MCs) and 2012 electoral campaign contribution records.

Tweets from Capitol Hill

	# MCs	# tweets	# words
Republicans	249	166,520	1,574,636
Democrats	189	98,661	949,862
Independents	2	818	7,312
Total	440	265,999	2,531,810

Table 3.11: Statistics of our tweet dataset.

During the period from May 15–October 31, 2012, we collected through Twitter's search API publicly available tweets posted by Twitter handles officially associated with MCs. These handles were collected from Tweet Congress.[31] We manually filtered from this set MCs who were not seeking reelection in 2012. Although we do not know who authored any of these tweets, we assume that they are, for the most part, rationally and carefully crafted by the MC's staff. In (Golbeck et al., 2010) the authors manually coded a large corpus of MC tweets and found the majority of messages to be public relations and promotion, not personal. Our (less systematic) analysis of the data leads to a conclusion consistent with their finding.

Each tweet was lightly preprocessed. Hashtags and at-mentions were retained; URLs, non-alphabetic strings, and 134 common stop words were not. Downcasing was applied, and regular expressions were used to normalize some segmentation and lengthening variation. Finally, words occurring less than 10 times in the corpus were removed, resulting in a vocabulary of 19,233 word types, and an average tweet length of 9 word tokens. See Table 3.11 for more details.

Electoral Campaign Contributions

For each of the MCs in our tweet dataset, we collected 2012 general election campaign contribution data from the publicly available database maintained by the Center for Responsible Politics (CRP). [32] These data were originally released by the Federal Election Commission; CRP performs manual disambiguation and aggregation. Contributions are aggregated by industries and other interest groups defined by CRP. (Hereafter we use the term "industry" to refer to both types of groups.) 91 categories appear in the 2012 data; see Figure 3.2 for the total contributions of the top 20 industries. The variation across industries is very large; lawyers and law firms account for 8% of the total, and the top ten account for 46%.

In the subsequent discussion, c will always index MCs. We let $\boldsymbol{w}_c$ be the complete collection of word tokens tweeted by c; $\boldsymbol{w}_{c,t}$ is the tth message by c, and $w_{c,t,n}$ is the nth word in that message. For each MC, we convert absolute amounts to fractions of the total amount received

[31] http://www.tweetcongress.org

[32] http://www.opensecrets.org

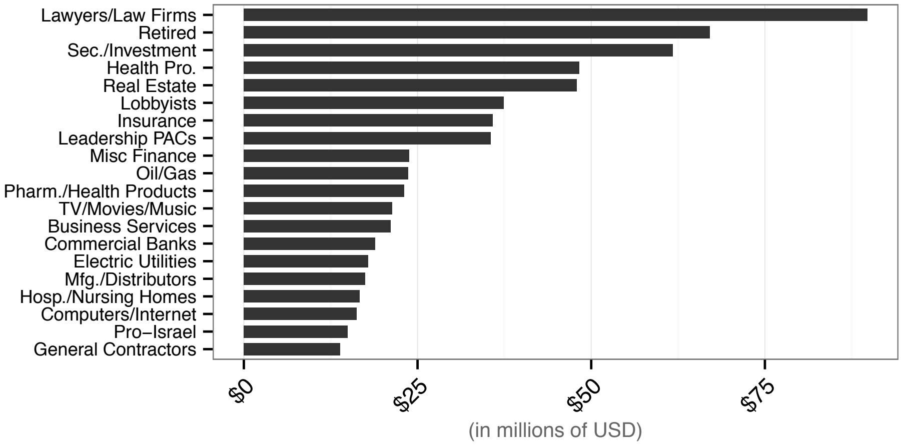

Figure 3.2: Top 20 contributor industries for the 2012 Congressional Elections (out of 91 in our data); statistics are as of the end of September 2012. Total spending is $1B, mean per industry is $11M, median $6.3M, s.d. $15M.

in contributions. This transformation is meant to help control for variation in spending levels across campaigns, which is large (mean $2.28M, s.d. $2.78M). Fixing the $I = 91$ industries, we let $\boldsymbol{\pi}_c \in \mathbb{R}^I$ denote the **contribution profile** for MC c, where $\sum_{i=1}^{I} \pi_{c,i} = 1$. We denote the complete collection of the message and the contribution profile in the corpus as $\boldsymbol{W}$ and $\boldsymbol{\Pi}$, respectively.

3.4.2 Proposed Approach: Generative Model Revisit

For the current task we revisit the generative approach which we cultivated in chapter 2. We have discussed several advantages to this approach in the previous sections (Section 2.4).

The generative approach requires us to state our assumptions about how the data are generated, and in particular how the random variables relate to each other. The two main random variables involved here are the tweet texts $\boldsymbol{W}$ and campaign profiles $\boldsymbol{\Pi}$. We therefore first describe a probabilistic model over these variables, or the joint probability distribution $p(\boldsymbol{W}, \boldsymbol{\Pi})$ to precisely state our assumptions on how the two relate to each other.

As in the tasks from chapter 2, we employ a latent topic model as the starting point, but this time we extend the basic model in a slightly different way from the last time. Previously we augmented the model by adding sets of multinomial random variables to represent richer sets of observation in the generative process. All the additional distributions are conditioned on the document specific topic distributions. This approach, sometimes called the "downstream approach" (Mimno and McCallum, 2008), is a popular tactic in extending hierarchical Bayesian models (such as LDA). A perhaps less popular alternative, often dubbed as "upstream approach", seeks

Author	
Sen. Claire McCaskill	Soooooo close. My spending freeze amndmnt got 59 votes today. Very bipartisan.We'll keep trying
Rep. Bill Cassidy	interiora permitting logjam blocks more job creation #louisiana #latcot
Rep. Cathy McMorris	Be sure to subscribe to your House RepublicanYouTube Channel – find them all here:
Sen. Orrin Hatch	proud of republicans standing together to not allow legislation through the senate until we vote against tax hikes #utpol
Rep. Grace Napolitano	Ron Artest and Mia St. John received mental health training last week, appreciate support for youth #mentalhealth
Rep. Mike Pompeo	remember to cast your vote for mike pompeo in the winfield courier online poll for the district race
Sen. Tom Harkin	Crunch Time on Health Reform http
Rep. Fred Upton	My op-ed with Rep. Joe Pitts: Republicans Offer a Prescription for a Healthier America
Rep. John Fleming	Fleming Urges Colleagues to Cut Spending and Extend Tax Cuts:
Rep. Mike McIntyre	We pray for the families of the brave Special Ops Forces who gave the ultimate sacrifice for our freedom.
Sen. Jerry Moran	Sounds frustrating. Please call my office in Topeka on Monday to see if we can help.
Rep. Steven Palazzo	Low taxes and small businesses will help the economy recover, not bloated stimulus packages. RT if you agree!

Table 3.12: Sample messages from our tweet corpus.

to tie the random variables through the sharing of Dirichlet parameters or document level topic distributions.

Instead of adding the campaign contribution as another set of downstream distributions, we associate each topic to one industry, thereby using the contribution profile $\boldsymbol{\pi}_c$ to directly represent the document-specific topic distribution. Given a new tweet message (or set of tweet messages by one author) we can derive the prediction through posterior inference over the topics mixture. Moreover, since each topic is associated with its own word distribution, the model training should discover a unique unigram language model for each industry. This is in a sense "guided" learning of topics; a few variations have been suggested in recent years to explore various types of guidance in parameter learning. See Section 2.7. Note that the tweet messages by the same MC share the same topic distribution across the corpus.

In our experiment we consider two different variations of this same idea. In each case, the approach is to reason inductively; we estimate the model parameters from a pool of training examples, and then estimate predictive performance on a held-out test set.

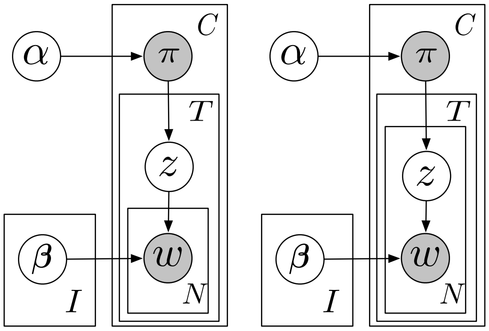

Figure 3.3: Graphical model representations of our two models, SIPT (left) and MIPT (right). The plates correspond to replicates; there are I industries and C MCs. T tweets per MC, and N words per tweet (T and N vary by MC and tweet; subscripts are suppressed for clarity). The difference is that SIPT assigns a single industry to each tweet, while MIPT assigns an industry to each *word*.

3.4.3 Model Specification

Below we consider two different variations. We present the generative stories first, followed by the prediction procedure.

Single Industry Per Tweet

In our first model, "single industry per tweet" (**SIPT**), we assume that each tweet is influenced by only one industry. In the first model, the generative story, for each MC c, is:

1. Draw a contribution profile $\boldsymbol{\pi}_c \sim \text{Dirichlet}(\boldsymbol{\alpha})$.
2. For each tweet by c, indexed by t:
   (a) Draw an industry $z_{c,t} \sim \text{Multinomial}(\boldsymbol{\pi}_c)$.
   (b) For each word in the tweet, indexed by n,
      i. draw $w_{c,t,n} \sim \text{Multinomial}(\boldsymbol{\beta}_{z_{c,t}})$.

Figure 3.3 (left) depicts SIPT as a graphical model. The joint distribution (for one MC c) is the following:

$$p(\boldsymbol{w}, \boldsymbol{\pi} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) = p(\boldsymbol{\pi} \mid \boldsymbol{\alpha}) \prod_t \pi_{z_t} \prod_n \beta_{z_t, w_{t,n}}$$

Note that we do not postulate a Dirichlet distribution over the parameters for the word distributions. Since we do not use sampling, it is not strictly necessarily to make this part of the generative story explicit (as it is the case for MCMC sampling). [33]

[33] It is necessary in order to derive collapsed Gibbs sampling algorithm.

Multiple Industries Per Tweet

Our second model, "multiple industries per tweet" (MIPT) assumes that each tweet is influenced by a mixture of industries, with each word being selected from a different industry. The generative story is, for each MC c, is:

1. Draw a contribution profile $\boldsymbol{\pi}_c \sim \text{Dirichlet}(\boldsymbol{\alpha})$.
2. For each tweet by c, indexed by t:
   (a) For each word in the tweet, indexed by n:
      i. Draw an industry $z_{c,t,n} \sim \text{Multinomial}(\boldsymbol{\pi}_c)$.
      ii. Draw $w_{c,t,n} \sim \text{Multinomial}(\boldsymbol{\beta}_{z_{c,t,n}})$.

Figure 3.3 (right) shows the graphical representation of MIPT. The joint distribution (for one MC c) is the following:

$$p(\boldsymbol{w}, \boldsymbol{\pi} \mid \boldsymbol{\alpha}, \boldsymbol{\beta}) = p(\boldsymbol{\pi} \mid \boldsymbol{\alpha}) \prod_t \prod_n \pi_{z_{t,n}} \beta_{z_{t,n}, w_{t,n}}$$

It is worth noting how these two models relate to some familiar probabilistic models for text. SIPT is similar to the mixture model which Naïve Bayes classifier is derived from. naïve Bayes is often used for text categorization in NLP, and we have briefly discussed this model in Section 2.4. As mentioned, Naïve Bayes training usually treats the label of a text as observed variable. In our model assumption, we instead assume that the only author-level (MC-level) proportions $\boldsymbol{\pi}_c$ is observable. MIPT is similar to LDA, a model used to infer latent topics in text collections which we reviewed in chapter 2 alongside Naïve Bayes. The topics in LDA are analogous to our industries. The difference is again in the assumption on how the variables are observed. LDA learns from documents whose associations to topics are completely unknown, so that each $\boldsymbol{\pi}_c$ ($\boldsymbol{\theta}$ in standard LDA notation) is latent. Here, the proportions are observed. Naturally, in both cases, the prediction and learning algorithms required are somewhat different from the classic models.

Prediction

Given a new MC c (who is not included in the training data), we wish to predict $\boldsymbol{\pi}_c$ from the set of messages $\boldsymbol{w}_c$. During prediction, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are fixed. For both models, exactly solving for $\boldsymbol{\pi}_c$ given the parameters and $\boldsymbol{w}_c$, and summing out all possibilities for $\boldsymbol{z}_c$, is intractable. For SIPT, we apply a single round of message passing, calculating each $z_{c,t}$ based on $\boldsymbol{\beta}$, then $\boldsymbol{\pi}_c$. (We found that additional rounds were not helpful during the preliminary examination.) For MIPT, which involves a more complex latent variable space, we apply mean field variational inference, an approximate technique widely used in Bayesian modeling (Wainwright and Jordan, 2008). The algorithm alternates between estimating posteriors over $\boldsymbol{z}_c$ and over $\boldsymbol{\pi}_c$.

3.4.4 Notes on Inference and Parameter Estimation

During learning, for a collection of MCs c, $\boldsymbol{\pi}_c$ is observed along with words $\boldsymbol{w}_c$, and the goal is to maximize likelihood with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. Because $\boldsymbol{\pi}_c$ is observed, we can estimate $\boldsymbol{\alpha}$

and $\boldsymbol{\beta}$ separately. This follows from the "d-separation" property observable in Figure 3.3: there is no active path between $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. For $\boldsymbol{\alpha}$, we seek the maximum likelihood estimate of the Dirichlet parameters given $\{\boldsymbol{\pi}_c\}_c$. There is no closed-form solution for the MLE, so we apply a well-known fixed-point iterative method (Minka, 2000).

To learn $\boldsymbol{\phi}$, we use a variational EM algorithm. This algorithm alternates between approximate inference over the $\boldsymbol{z}$ variables (given the current posterior over $\boldsymbol{\beta}$) and maximizing over $\boldsymbol{\beta}_i$ for each industry (given the posteriors over $\boldsymbol{z}$). For SIPT, we learn $\boldsymbol{\beta}$ using a single round of message passing, calculating each $z_{c,t}$ based on $\boldsymbol{\pi}_c$, then maximizing $\boldsymbol{\beta}$. For MIPT, our algorithm is quite similar to the learning algorithm for LDA given by (Blei et al., 2003), but without having to estimate posteriors over tweet-level proportions (since they are observed). As in standard LDA, there is a closed-form solution for maximization over $\boldsymbol{\beta}$.

3.4.5 Experimental Results

Our experiments are based on multi-fold cross-validation. In each trial, we held out five (distinct) MCs with tweet volume between the 25th and 75th percentiles. The remainder of the MCs' data were used to estimate parameters, and then predictions were made for the held-out MCs as described in the previous section. We repeat the process for 44 times, each time selecting the new set of held-out member, obtaining the predictive contribution profile for 221 members. Each prediction is a ranked list of industries based on our predicted value for $\boldsymbol{\pi}_c$. At the end of this process we obtain 221 ranked lists of industries, one for each test MC.

Performance Metrics

Recall that, for the evaluation of our first rank prediction task (blog response prediction from Chapter 2), we compared a predictive user ranking to the set of ground-truth users. In this task we also examine ranking performance, but this time both the prediction and the ground truth are ranked lists. We therefore take a slightly different approach in evaluation. We use two types of metrics: a rank-to-rank metric and a set-to-rank metric. Since the model output is a *set* of ranked lists (one for each test MC), for both metrics the final score is the macro average over all the MCs in the set.

The first, a rank-to-rank metric, is Kendall's τ rank correlation (Kendall, 1938), a fairly standard correlation measurement used in many fields. The metric measures the similarity of two orderings over the identical set of items. We applied this metric between the predictive ranking (over all 91 industries) and the ground-truth (also over 91 industries).

Our rank-to-rank metric consider the two rankings in full. While the comparison is fair, it puts some unwanted emphasis on the part of the ranking which we do not necessarily care about. An MC typically receives contributions from (at most) a few dozen industries, and none at all from others. Therefore the ground truth rankings include many industries which made negligible or no contributions to the candidate. The relative rankings of such industries are therefore irrelevant to the task. We conduct a set-to-rank comparison in addition to the rank-to-rank to reflect this concern.

Model	τ	MAP@5	MAP@10	MAP@15
fixed pred.	.49	.52	.60	.65
log. reg.	.49	.41	.49	.55
SIPT	**.55**	**.58**	**.66**	**.70**
MIPT	.44	.42	.51	.57

Table 3.13: Experimental results. τ is Kendall's τ statistic, and MAP@k is mean average precision at relevance cutoff k.

For the second evaluation, we use a modified version of standard average precision (Manning et al., 2008), a widely used set-to-rank comparison method. The metric is applicable when there is a set of (unranked) ground truth set and a ranked prediction. To apply this metric, we first induce the "true" set by truncating the ground truth industry list to the top n donors. We then compute the predictive ranking's average precision against this set. We report the average of those averages over all MCs in the test set.[34] We use several cut off values n. In our table this metric is labeled as "MAP", followed by the the cutoff value.

Baseline

We include a fixed prediction that ignores tweet content. This ranks the contribution source according to the global mean (among training set members). Due to the large difference in spending among the contributing groups (see Figure 3.2), this predictor is expected to be a strong baseline.

We also include a multinomial logistic regression-inspired discriminative model as a simpler machine learning technique. This model is trained using $\boldsymbol{\pi}_c$ to define a fractional assignment of MC c across industries. The regularized maximum likelihood estimator is:

$$\arg\max_{\boldsymbol{\theta}} \sum_c \sum_i \pi_{c,i} \log p_{\boldsymbol{\theta}}(i|\boldsymbol{w}_c) + \lambda \|\boldsymbol{\theta}\|_2^2$$

where the multinomial logit distribution $p_{\boldsymbol{\theta}}$ is based on the same unigram features considered by our generative models. λ is tuned using the 10% of the training data as the development set.

Results

Results, averaged across folds, are shown in Table 3.13. For all metrics, the model performances in bold font are statistically significant at $p < 0.001$ with Wilcoxon signed rank test against the baseline. Fixed majority prediction, as expected, gives fairly strong performance. (Incidentally, a totally random ranking will achieve .10 for MAP5.) Only SIPT improves over the baseline significantly on all metrics. This increased predictability indicates a connection between contribution profiles and public messages. Of course, a causal relationship cannot be inferred (in either direction).

The dramatic difference in predictive performance across models suggests the importance of

[34] To be more precise, the average precision is: $\sum_{i=1}^{N} Prec(R_i)/|N|$, where R_i is the i highest-ranked items. The metric is fairly standard in natural language processing and information retrieval research.

careful model design. The discriminative model posits a similar word-industry association to our model but ignores the message level, assuming all messages are equally explained proportional to $\boldsymbol{\pi}_c$. MIPT posits a very high dimensional latent structure that may not be learnable from the amount of training data available here. SIPT strikes a better balance.

3.4.6 Descriptive Aspects of the Models

The experiment provides compelling evidence of a connection between campaign contributions and messages. Meanwhile, we found the MIPT model gives *qualitatively* better word-industry associations with greater face validity, despite its inadequacy as a predictor. This is not uncommon in unsupervised topic modeling; similar observations have been made before (Boyd-Graber et al., 2009).

Table 3.14 shows words MIPT associates with some industries. Many terms appear that are topically related to issues of interest to these industries. We also see states where these industries do business (NC/tobacco, AK/fishing), and the names of MCs who head committees relevant to the industry's interests (Harkin/agriculture, Inouye/casinos). Deviations are also interesting; Sen. Hagan's name associates with women's issues (EMILY's List is one of her top donors), but not tobacco, despite her NC constituency. Difference between the two energy sectors are also intriguing; while many words appeared in both industries, some issues such as unemployment ("jobs") or healthcare ("stophealthhike") appear to be of special importance to the constituency in the mining community.

Subjectively, we found the industry-word associations discovered by SIPT and the discriminative model to be far less interpretable. So while MIPT does not perform well as a predictive model, it more successfully infers human-interpretable associations. We also ran original LDA on just the text (without campaign contribution data); the topics were difficult to distinguish from each other.

We emphasize that these associations were revealed using only campaign contribution data coupled with tweets by MCs. We, humans, have a good idea why Alaskan interest is tied to Fishing industry, or why the tobacco lobby interest manifests as the North Carolina related words (and speculate on why it is conspicuously lacking specific references to the words relate to smoking of tobacco). Our models are completely ignorant of such fact and the result is purely due to the statistical trends of word usage and its relation to financial incentives.

3.5 Related Work

Below we review some works relevant to the tasks discussed in this chapter. Some of the works have already been mentioned in the previous sections. We repeat them here in the context when it serves for the sake of clarity. See Section 1.1, 1.4 and 3.2 for the additional discussions on related work.

Industry	Associated Terms
Computers & Internet	#sopa, internet, sopa, rights, tech, ag, property, judiciary, holder, oppose, federal, open
Defense (Electronics)	security, border, homeland, cyber, subcommittee, hearing, defense, air, nuclear, briefing, mike, turner, military
Defense (Aerospace)	defense, afghanistan, armed, appropriations, services, inouye, subcommittee, committee, budget, secretary, military, fort
Agr. Services/Products	harkin, iowa, farm, announces, farmers, qs, ag, nebraska, drought, webcast, nelson, agriculture
Agr. Tobacco	nc, #ncsen, burr, #nc7, hours, carolina, office, schedule, #ncpol, north, county, staff
Fisheries & Wildlife	#alaska, alaska, #ak, murkowski, alaskans, anchorage, photo, ak, weekend, air, fairbanks, office, native
Energy (Misc)	energy, gas, natural, oil, clean, resources, forward, #utpol, #energy, looking, wind, forest
Energy (Mining)	#4jobs, energy, epa, bills, gas, @housecommerce, #energy, passed, regulations, gop, #jobs, #stopthetaxhike
Commercial Banks	hearing, committee, financial, subcommittee, oversight, services, reform, @financialcmte, consumer, cmte, chairman, secretary
Securities & Investment	bipartisan, senate, bill, pass, extension, cut, compromise, house, tax, passed, #4jobs, jobs, gop
Credit Unions	tax, ur, mortgage, recession, generation, honest, blog, people, rate, terrorist, don, self
Health Professionals	health, medicare, care, obamacare, #obamacare, reform, republicans, repeal, seniors, healthcare, americans, democrats
Casinos & Gambling	inouye, senator, lujn, hawaii, nevada, heck, joe, nv, berkley, meeting, attending, #hawaii
Pro-Israel	iran, women, rights, nuclear, israel, ben, violence, gop, senate, security, #vawa, cardin
Women's Issues	hagan, nc, women, stabenow, mo, #hawaii, contracting, vets, #mo, #women, game, #nc

Table 3.14: MIPT's word-industry associations, for some manually selected industries.

On Politics and Text as Data

Congressional committee system and its power over the legislative process is one of the active areas of research in political science. Aside from (Adler and Wilkerson, 2005, 2012), which we discussed in detail in 3.3.3, scholars have suggested various other factors which could affect bills' survival through the system. These include legislators' personal skills, regional or intellectual cliques among the lawmakers, and various types of bill functional categorizations (Burstein et al., 2005; Price, 1972; Cohen and Malloy, 2010; Zhang et al., 2007b). Several of them employ empirical, data-driven approach, although in much smaller scope. Cost of data curation is a substantial overhead in conducting empirical studies in this area. An exception is the federal roll call voting records, which are maintained by Library of Congress going back to the 101st Congress. [35] As mentioned, many seminal works of empirical analysis on legislative politics are on the roll call records (Poole and Rosenthal, 1985; Cox and Poole, 2002; Poole and Rosenthal, 1991; Bafumi et al., 2005; Jackman, 1991; Clinton et al., 2004). Congressional Bills Project at the University of Washington, from which we gathered some of our metadata, is among a few initiatives toward large-scale data analysis *beyond* roll call voting study. As we discussed in 3.2, large scale *text* analysis is a fairly new concept, though has been gathering attention in recent years.

The use of text "as data" in political science has recently become a active area of research (Grimmer and Stewart, 2012, 2013; O'Connor et al., 2011). Many works in this area leverage tools and techniques from NLP (Laver et al., 2003; Laver and Garry, 2000; Klebanov et al., 2008; Benoit et al., 2009). The key idea here is to treat text as another categorical data in the statistical analysis. Algorithms used for text-driven forecasting are often applicable in this area. Latent topic models such as LDA are quite popular choice since they can be trained in unsupervised fashion, and often present humanly interpretable, succinct description of statistical trends in the texts. (Grimmer, Forthcoming, 2010) use Hierarchical Bayesian topic model with Von Mises-Fisher distribution over text to analyze congressional speech and press release. (Quinn et al., 2006) use hierarchical models similar to LDA to study Supreme Court proceedings. Many more recent works use various forms of latent topic model for corpus exploration, text categorization, and visualization purpose (Martin and Quinn, 2002; Quinn, 2004; Monroe et al., 2008; Quinn et al., 2010). Close to our bill survival model, (Gerrish and Blei, 2011, 2012) combined topic models with spatial roll call models to *predict* votes in the legislature from text alone. Their best results, however, came from a text regression model quite similar to our direct text model. Our work is different in that the models are discriminative, and utilize bill metadata information alongside bill texts. Also, our focus is not floor voting prediction but bill survival in the committee system, which happens much earlier than the floor voting.

On text-driven-prediction and tweets

In computer science, considerable recent work has modeled text alongside data about social behavior. This includes text-driven predictions (Kogan et al., 2009; Lerman et al., 2008), various kinds of sentiment and opinion analysis (Thomas et al., 2006; Monroe et al., 2008; O'Connor et al., 2010a; Das et al., 2009), and exploratory models (Steyvers and Griffiths, 2007). We presented more thorough coverage of this are in Section 1.1. Also refer to Section 3.4 for more discussion on text-driven forecasting works including the ones with tweet data.

One of the important trends in computational social science is the use of Twitter data as the surrogate for natural behavioral observations to allow data-driven inquiry. (Eisenstein et al., 2011)

[35] The number is as of September 2012. The amount of data is increasing. Library of Congress has been adding the historical data to publicly accessible archive over time in reverse chronological order.

examines the questions of demographic associations with linguistic variation; (Kooti et al., 2012) examined the formation of social norms. Another popular line of research is the use of Twitter data as social sensor for event prediction and detection, such as the emergence of flu epidemics (Paul et al., 2010), epicenter of earthquakes (Sakaki et al., 2010) or rise and fall of public opinion (O'Connor et al., 2010a). Some of the most active tasks in this are related to politics; there has been quite a few works on voter sentiment detection and the election result prediction based on Twitter data (Tumasjan et al., 2010; Gayo-Avello, 2012b; Jungherr et al., 2012; O'Connor et al., 2010a). In (Gayo-Avello, 2012a), authors give a comprehensive overview on this topics.

On the subject of the U.S. Congress and MC's tweet messages, there are relatively few computational approaches. Among them, a notable work is done by (Golbeck et al., 2010), in which the authors conducted a comprehensive analysis of the use of Twitter by MC. Also noteworthy is the work by (White and Counts, 2012), which incorporated Twitter data in a spatial model of political ideology and examined various theories in the political tweets (or expressed ideology) and legislative actions.

To our knowledge, there is no work which connects the campaign contribution profiles to the Congressional tweets, or conducts prediction tasks on the subject.

3.6 Summary and Contribution

In this chapter we presented predictive systems which address two important questions concerning the United States Congress. One system predicts whether a bill is going to survive through the congressional committee system based on the bill metadata and the textual contents. The other predicts campaign contribution profiles for MCs based on their public tweets.

For both, we designed our stochastic models utilizing the approaches well used for NLP applications. These techniques however are relatively new for the the purpose of real world prediction. We empirically show that our models are competitive for the target tasks in strictly predictive evaluation settings.

Model interpretability is an important issue in computational social science applications. For the bill prediction tasks, we closely engaged the political science insights into our models through the feature engineering process. Because of this construction, the resulting models, their prediction performance and estimated feature weights, were able to make testimonies on the extrinsic values of the underlying theories. We further engaged expert knowledge during the post-hoc evaluation, discovering the intriguing trends in bill survival, and at the same time demonstrating our models' potential for exploratory discovery.

For the campaign contribution prediction tasks, a straightforward generative approach was applied to the congressional tweet message for the purpose of prediction and topic discovery. Because each contributing interest group is explicitly tied to a unique latent topic, we were able to learn the group-specific language model, which was used for the prediction task. We showed that the discovered topics have high face validity, and agree well with our understanding of campaign contribution and political speech. The model provides clearer display of correlation between the political language and financial incentives, which otherwise is hard to tease out from the large number of unannotated microblog messages.

To summarize, our contributions from this section are the delivery of the following:

1. Effective and interpretable probabilistic models of text-driven prediction concerning some of the most important questions in U.S. politics. The techniques developed here could serve for analytic or assistive application particularly useful for those who wish to effectively monitor legislative politics.

2. Novel case studies on the text-driven inquiry into the political systems utilizing the large-scale corpus analysis. Exploitation of the textual resource, which are increasingly more available due to electronic archiving and social media, is a rising yet under-explored area in the quantitative political science.

3. Successful applications of the probabilistic techniques often used in NLP applications to novel tasks in the computational social science domain, demonstrating that text-driven prediction as a viable inquisitive option in the emerging discipline.

Chapter 4

Conclusion and Future Work

In previous chapters we presented a set of new text-driven prediction tasks in the domain of American politics. In developing our systems we had two criteria in mind; the model is to perform the target prediction tasks well, and also give human interpretable results at the end. We chose the probabilistic modeling approach to better serve these goals. For all tasks, we first postulated the stochastic relationships between the texts and the prediction targets, which we viewed as the actions caused, or "actuated," by the texts. We explored a variety of hypotheses on these relationships through model structure design or feature engineering. Through empirical evaluation we demonstrated the soundness of our proposed models on both criteria.

Prediction of real world events, fueled by text analysis, is starting to be a familiar subject in the contemporary NLP research community. Progress in this area is of high concern for multiple disciplines. We believe two recent occurrences are particularly relevant. One is the advent of social media, and the consequential challenges and opportunities in the exploitation of large scale user-generated contents, a substantial share of which are text. The other is the increasing awareness among quantitative social science researchers of the need for data-driven NLP. An especially interesting development is seen amongst quantitative political scientists, where an emerging emphasis on text-driven inquiry is often dubbed as *text-as-data*.

Social media (and collaborative media such as blogs) daily produce a vast quantity of user generated texts. How to turn this raw resource into intelligence is an active area of inquiry for both academic and industry researchers across many sectors. Text-driven response prediction can be useful in several ways in this context. It could be useful as a core algorithm for user assistive technologies such as personalized recommendation or filtering systems, and also for trend monitoring or profiling of user communities. It could also be useful for corpus exploration and discovery. (We have discussed some of these and other potential applications and relevant research in previous chapters.)

Contemporaneous with the rise of user generated texts, political scientists started to pay closer attention to corpus based text analysis. Although quantitative analysis is a cornerstone of empirical political science, statistical text analysis has never been in its main stay. Text-driven prediction is one way to unite social science questions and text analysis into a computational framework. Model intelligibility, our second criterion for success, is especially relevant in this context.

The specific tasks we chose for this dissertation are highly interesting from both of the above perspectives. These tasks are approachable by familiar techniques in NLP, yet each offers new challenges, requiring innovation beyond perfunctory applications of existing techniques. For predictions in the blogosphere, the challenge was how to make the best use of user *comments* — noisy, seemingly unreadable run of reader generated texts. We chose a latent topic model approach. The technique has been shown competent in both prediction and corpus exploration in other domains, yet there were no previous works dealing with user comments. Our challenge was therefore to extend the basic designs to capture reactive relationships between political bloggers and their community.

The prediction tasks from the third chapter are perhaps more relevant to computational social science trends, and in particular, text-as-data initiatives in political science. For the bill survival prediction task, our preliminary survey had convinced us that, in order to be competitive in a realistic prediction setting, the model must be able to incorporate an arbitrary number of metadata alongside bill texts. For this reason we chose a feature-based discriminative model over the generative approach. The challenge here was the development of sensible features. We engaged political science collaborators throughout the feature engineering process, as well as during the post-hoc analysis of the learned model. For the campaign contribution prediction task, we again considered topic modeling, since we anticipated the greater interest in corpus discovery and exploration in this subject. Congress persons' microblog messages were examined before, but never before for their relation to campaign contributions; political donation is an established issue, but it has never been studied alongside natural, spontaneously occurring texts such as tweets. We successfully induced the contribution specific language model, and applied them for the prediction task. The main challenge here was again how to extend the basic design to incorporate the real world response (campaign contribution data) to the generative story over the texts (tweets).

The work presented in this dissertation is by no means complete, but we believe that we have delivered a significant kernel of knowledge serviceable in the emerging area of text-driven prediction and computational social science. We hope the researchers and engineers in these fields find them useful as well.

Future Works

Our empirical evaluations demonstrated that our proposed models are fundamentally sound, intelligible, and competitive in the target tasks. They also clarified some shortcomings of the model, and what would be the next steps in this line of research. We note some of our thoughts in future works in this section.

The latent topic models we developed in Chapter 2 simplified the user comments as an aggregated count of comments, or unordered set of user id and unigram words. Needless to say, there are a number of different attributes in comments as well as in main post we chose not to include at this time. Extending our models to include these attributes is one interesting direction in future. For example, time of day likely affects the popularity of blog the posts. We also suspect that the authorship of the post, when the blog site includes multiple writers, likely influences both the comment popularity and the commenter identities. It is possible that including these evidences improves the model performance in our prediction tasks. It may also permit a more nuanced understanding of language in the blogosphere and in political discourse more generally. Of course improved performance might also be obtained with more topics, richer priors over topic distributions, or the models which permit weaker independence assumptions among users, blog posts, or blog sites. Several types of topic model offer suitable architecture for such improvements (Blei and Lafferty, 2006; Wang et al., 2012; Blei and Lafferty, 2007). We also observed that our model does not perform uniformly well across blog sites. For example, in commenter prediction task, ignoring comment contents entirely improved the performance for some site, but not for others. Models which capture these differences among the sites during the learning process would perhaps be another candidate for future work. Such models would permit blog-specific properties to be considered during the prediction, so that, for example, the comment words can be exploited when they are helpful but assumed independent when they are not.

As we mentioned, in the bill prediction task we left out several promising features due to the cost of feature engineering. Some of these features, such as sub-committee chairmanship, are worth revisiting in future work. Another possible direction is reconsidering impact score and the course of model inspection. Interpretation of discriminative models is often difficult, and our impact score helped us with the model examination, but is not without shortcomings. A more thorough examination of this issues and alternatives methods would make a useful future study.

An important tangent of future research is to contemplate how these text-driven predictions could turn into *real world* applications. There are variety of possible engineering issues, but perhaps two of the most apparent ones are the issues of training data size and model calibration over time.

Although blogosphere or twitter as a whole is large, once pared down to the subgroups of interest (such as a specific blog site, or tweets from small subgroups such as members of the current Congress) the size of the suitable data at our disposal can be quite modest. This could lead to a data sparsity problem. Note that a data sparsity problem is not only a function of data size but also of the complexity of the model, or rather the number of model parameters. Recall that in our blog user prediction experiments (Section 2.5.3), our topic models did poorly on DK and RWN. In these sites, the best overall performing model was Naïve Bayes, a classifier which is derived from a much simpler generative story. In both sites, we strongly suspect that the cause of the sluggish performance is the data sparsity problem. DK, with its whipping 16,849 users, had the most number of parameters among the five sites.[1] The number of users of RWN is much smaller than DK, however it is the least prolific site among our corpus, and therefore has the least amount of training data. Application development based on our model should consider first and foremost a careful performance benchmarking with respect to the model complexity (parameter size) and the amount of training data.

Another important engineering consideration in our models is how to calibrate them over time. Note that, unlike rule-base models, our models utilize corpus based machine learning techniques and therefore can be retrained with new data periodically if necessary. For our campaign con-

[1] The parameter size of our topic model is the function of vocabulary size, topic size, and user size. Note that the size of vocabulary does not change from site to site as drastically as the size of users.

tribution predictor, model retraining is obviously necessary as a new contribution cycle begins. Although our other models do not have as drastic or immediate constraints as the congressional election, they will nevertheless become obsolete over time. Recall that, in the exploratory analysis of the user comments (Section 2.6.3) we noted an increasing trend in the comment volume in MY over time. This trend can be due to an external reason (perhaps the approaching of the general election) or internal (perhaps the blog author is becoming more popular). In either case, the model needs to be adjusted to the new trends to make an accurate prediction. Other reasons the model might require new training include the turnover of new users (new blog readers or newly appointed congressman), or the change in the discussion topics (emergence of new topics, or obsoleted agendas). How often a model should be retrained depends on the various engineering factors and goal of the end applications. Again, careful benchmarking of the performance along with such factors as the training time or data curation cost would be critical.[2]

Comparing to discriminative features, word distributions from the topic model are easier to understand by human. Nonetheless, lack of higher order linguistic notions in the unigram word based topic model is often a source of criticism. Several works in machine learning and statistics have attempted to remedy this shortcoming (Griffiths et al., 2005; Boyd-Graber and Blei, 2010; Li and McCallum, 2005). Considering that intelligibility is an important aspect in computational social science, advancement in this area, and empirically examining how these models help with the informativeness or interpretability, is an interesting research direction. A related subject is the question of how to evaluate *topic quality* of any given model. It is not always easy to assess the quality of learned topics (or quality of learned feature weights for that matter). Several researchers have suggested empirical evaluation methodologies for topic quality (Boyd-Graber et al., 2009), although the area is yet to see strong consensus on what it means to be a "reasonable" evaluation of topic quality. In our work, extrinsic evaluation tasks (prediction tasks), and expert helps (in the case of bill prediction task), to some extent helped us with the assessment of the models. However, as we observed in the campaign contribution prediction task, prediction performance of the model does not always relate to the topic quality. Although the fundamental questions involved in this issue perhaps go much beyond the scale of the current dissertation, we note that this line of research is one of the most important directions concerning ours and similar works in computational social science.

[2] Note that the training speed is largely a function of the data size (and model complexity), therefore the same model might consume much longer training time if the data is different.

Bibliography

L. Adamic and N. Glance. The political blogosphere and the 2004 U.S. election: Divided they blog. In *Proc. of the 2nd Annual Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2005.

E. Scott Adler and John Wilkerson. The scope and urgency of legislation: Reconsidering bill success in the house of representatives. In *Proc. of annual meetings of the American Political Science Association*, Washington, DC, 2005.

E. Scott Adler and John Wilkerson. *Congress and the Politics of Problem Solving*. Cambridge University Press, London, 2012.

Nitin Agarwal, Huan Liu, Lei Tang, and Philip S. Yu. Identifying the influential bloggers in a community. In *Proc. of WSDM*, 2008.

Amr Ahmed and Eric P. Xing. Staying informed: supervised and semi-supervised multi-view topical analysis of ideological perspective. In *Proc. of EMNLP*, 2010.

Galen Andrew and Jianfeng Gao. Scalable training of l 1-regularized log-linear models. In *Proceedings of the 24th international conference on Machine learning*, pages 33–40. ACM, 2007.

David Austen-Smith. Interest groups, campaign contributions, and probabilistic voting. *Public choice*, 54(2):123–139, 1987.

J. Bafumi, A. Gelman, D. Park, and N. Kaplan. Practical issues in implementing and understanding bayesian ideal point estimation. *Political Analysis*, pages 134–153, 2005.

Ramnath Balasubramanyan, William W. Cohen, Doug Pierce, and David P. Redlawsk. What pushes their buttons? predicting comment polarity from the content of political blog posts. In *Proc. of Workshop on Language in Social Media (LSM 2011)*, 2011.

Ramnath Balasubramanyan, William W. Cohen, Douglas Pierce, and David P. Redlawsk. Modeling polarizing topics: When do different political communities respond differently to the same news? In *Proc. of ICWSM*, 2012.

Mikhail Bautin, Lohit Vijayarenu, and Steven Skiena. International sentiment analysis for news and blogs. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2008.

Kenneth Benoit, Michael Laver, and Slava Mikhaylov. Treating words as data with error: Uncertainty in text statements of policy positions. *American Journal of Political Science*, 53(2): 495–513, 2009.

Edward Benson, Aria Haghighi, and Regina Barzilay. Event discovery in social media feeds. In *Proc. of ACL*, 2011.

Adam L Berger, Vincent J Della Pietra, and Stephen A Della Pietra. A maximum entropy approach to natural language processing. *Computational linguistics*, 22(1):39–71, 1996.

D. Blei and M. Jordan. Modeling annotated data. In *Proc. of SIGIR*, 2003.

D. Blei and J. Lafferty. Topic models. In A. Srivastava and M. Sahami, editors, *Text Mining: Theory and Applications*. Taylor and Francis, 2009.

D. Blei and J. McAuliffe. Supervised topic models. In *Advances in Neural Information Processing Systems 20*, 2008.

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

David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM, 2006.

David M Blei and John D Lafferty. A correlated topic model of science. *The Annals of Applied Statistics*, pages 17–35, 2007.

Jonathan Boyd-Graber, Jordan Chang, Sean Gerrish, Chong Wang, and David Blei. Reading tea leaves: How humans interpret topic models. In *NIPS*, 2009.

Jordan Boyd-Graber and David M Blei. Syntactic topic models. *arXiv preprint arXiv:1002.4665*, 2010.

S. R. K. Branavan, H. Chen, J. Eisenstein, and R. Barzilay. Learning document-level semantic properties from free-text annotations. In *Proc. of ACL-08: HLT*, 2008.

Paul Burstein, Shawn Bauldry, and Paul Froese. Bill sponsorship and congressional support for policy proposals, from introduction to enactment or disappearance. *Political Research Quarterly*, pages 295–302, 2005.

George Casella and Roger L. Berger. *Statistical Inference*. Duxbury Press, 2nd edition, June 2001.

George Casella and Christian Robert. *Monte Carlo Statistical Methods*. Springer, 2nd edition, 2004.

Paula Chesley, Bruce Vincent, Li Xu, and Rohini Srihari. Using verbs and adjectives to automatically classify blog sentiment. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW)*, pages 27–29, 2006.

Joshua Clinton, Simon Jackman, and Doug Rivers. The statistical analysis of roll-call data. *American Political Science Review*, pages 355–370, 2004.

Lauren Cohen and Christopher Malloy. Friends in high places, 2010.

D. Cohn and T. Hofmann. The missing link—a probabilistic model of document content and hypertext connectivity. In *Neural Information Processing Systems 13*, 2001.

Gary W. Cox and Keith T. Poole. On measuring partisanship in roll-call voting: The u.s. house of representatives, 1877-1999. *American Journal of Political Science*, pages 477–489, 2002.

Pradipto Das, Rohini Srihari, and Smruthi Mukund. Discovering voter preferences in blogs using mixtures of topic models. In *AND '09: Proc. of The Third Workshop on Analytics for Noisy Unstructured Text Data*, pages 85–92, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-496-6. doi: http://doi.acm.org/10.1145/1568296.1568311.

Homero Gil de Zúñiga. Blogs, journalism, and political participation. *Journalism and citizenship: New agendas in communication*, page 108, 2009.

Arthur T Denzau and Michael C Munger. Legislators and interest groups: How unorganized interests get represented. *The American Political Science Review*, pages 89–106, 1986.

Laura Dietz, Steffen Bickel, and Tobias Scheffer. Unsupervised prediction of citation influences. In *In Proceedings of the 24th International Conference on Machine Learning*, pages 233–240, 2007.

Peter Sheridan Dodds and Christopher M. Danforth. Measuring the happiness of large-scale written expression: Songs, blogs, and presidents. *JOURNAL OF HAPPINESS STUDIES*, 11 (4):441–456, 2008.

M. Dredze, H. M. Wallach, D. Puller, and F. Pereira. Generating summary keywords for emails using topics. In *Proc. of the 13th International Conference on Intelligent User Interfaces*, 2008.

Jacob Eisenstein, Noah A. Smith, and Eric P. Xing. Discovering sociolinguistic associations with structured sparsity. In *Proc. of ACL*, 2011.

Jaocb Eisenstein. What to do about bad language on the internet. In *Proc. of NAACL*, 2013.

E. Erosheva, S. Fienberg, and J. Lafferty. Mixed membership models of scientific publications. *Proc. of the National Academy of Sciences*, pages 5220–5227, April 2004.

Lawrence C. Evans. Participation and policy making in senate committees. *Political Science Quarterly*, 1991.

William P Eveland and Ivan Dylko. Reading political blogs during the 2004 election campaign: Correlates and political consequences. *Blogging, citizenship, and the future of media*, pages 105–126, 2007.

Yi Fang, Luo Si, Naveen Somasundaram, and Zhengtao Yu. Mining contrastive opinions on political texts using cross-perspective topic model. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 63–72. ACM, 2012.

Katja Filippova and Keith B. Hall. Improved video categorization from text metadata and user comments. In *Proc of SIGIR*, 2011.

Jerome H. Friedman, Trevor Hastie, and Rob Tibshirani. *The Elements of Statistical Learning*. Springer, New York, NY, 2009.

Jerome H. Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

Daniel Gayo-Avello. I wanted to predict elections with twitter and all I got was this lousy paper: A balanced survey on election prediction using Twitter data, 2012a. arXiv 1204.6441.

Daniel Gayo-Avello. No, you cannot predict elections with twitter. *IEEE Internet Computing*, 16(6):91–94, 2012b.

Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian data analysis*. Chapman & Hall/CRC, 2004.

Matthew Gentzkow and Jesse M Shapiro. What drives media slant? evidence from us daily newspapers. *Econometrica*, 78(1):35–71, 2010.

Sean Gerrish and David Blei. A language-based approach to measuring scholarly impact. In *Proc. of ICML*, 2010.

Sean Gerrish and David Blei. Predicting legislative roll calls from text. In *Proc. of ICML*, 2011.

Sean M. Gerrish and David M. Blei. How they vote: Issue-adjusted models of legislative behavior. In *Proc. of NIPS*, 2012.

Eric Gilbert, Tony Bergstrom, and Karrie Karahalios. Blogs are echo chambers: Blogs are echo chambers. In *Proc. of HICSS*, 2009.

Namrata Godbole, Manjunath Srinivasaiah, and Steven Skiena. Large-scale sentiment analysis for news and blogs. In *Proc. of ICWSM*, 2007.

Jennifer Golbeck, Justin Grimes, and Anthony Rogers. Twitter use by the U.S. congress. *Journal of the American Society for Information Science and Technology*, 61(8), 2010.

T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proc. of the National Academy of Sciences*, 101 Suppl. 1:5228–5235, April 2004.

Thomas L Griffiths, Mark Steyvers, David M Blei, and Joshua B Tenenbaum. Integrating topics and syntax. *Advances in neural information processing systems*, 17:537–544, 2005.

Justin Grimmer. A Bayesian hierarchical topic model for political texts: Measuring expressed agendas in Senate press releases. *Political Analysis*, 18(1), 2010.

Justin Grimmer. A Bayesian hierarchical topic model for political texts: Measuring expressed agendas in Senate press releases. *Political Analysis*, Forthcoming.

Justin Grimmer and Brandon Stewart. Text as data: The promise and pitfalls of automatic content analysis methods for political documents, 2012. `http://www.stanford.edu/~jgrimmer/tad2.pdf`.

Justin Grimmer and Brandon M Stewart. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, 2013.

D. Gruhl, R. Guha, David Liben-nowell, and A. Tomkins. Information diffusion through blogspace. In *In WWW 04*, pages 491–501. ACM Press, 2004.

Richard Hall. Participation and purpose in committee decision-making. *American Political Science Review*, pages 105–128, 1998.

Gregor Heinrich. Parameter estimation for text analysis. Technical report, University of Leipzig, 2008. URL `http://www.arbylon.net/publications/text-est.pdf`.

Simon Jackman. Multidimensional analysis of roll call data via bayesian simulation. In *Proc. of NAACL*, pages 227–241, 1991.

Charles W. Johnson. *How Our Laws Are Made*. U.S. GOVERNMENT PRINTING OFFICE, Washington, DC, 2003. U.S. House of Representatives Document 108D93.

Mahesh Joshi, Dipanjan Das, Kevin Gimpel, and Noah A. Smith. Movie reviews and revenues: An experiment in text regression. In *Proc. of NAACL*, 2010.

Andreas Jungherr, Pascal Jürgens, and Harald Schoen. Why the pirate party won the german election of 2009 or the trouble with predictions: A response to Tumasjan et al. *Social Science Computer Review*, 30(2):229–234, 2012.

David Karpf. Understanding blogspace. *Journal of Information Technology and Politics*, 5(4): 369–385, 2008.

M Kendall. *A New Measure of Rank Correlation*, volume 30. 1938.

A. Kittur, B. Suh, and E. Chi. What's in wikipedia? mapping topics and conflict using collaboratively annotated category links. In *Proc. of CHI*, 2009.

Beata Beigman Klebanov, Daniel Diermeier, and Eyal Beigman. Lexical cohesion analysis of political speech. *Political Analysis*, 16(4):447–463, 2008.

Minsam Ko, HW Kim, MY Yi, Junehwa Song, and Ying Liu. Moviecommenter: Aspect-based collaborative filtering by utilizing user comments. In *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*, pages 362–371, 2011.

Shimon Kogan, Dimitry Levin, Bryan R. Routledge, Jacob S. Sagi, and Noah A. Smith. Predicting risk from financial reports with regression. In *Proc. of NAACL-HLT*, 2009.

Farshad Kooti, Haeryun Yang, Meeyoung Cha, P. Krishna Gummadi, and Winter A. Mason. The emergence of conventions in online social networks. In *ICWSM*, 2012.

Glenn Krutz. Issues and institutions: Winnowing in the u.s. congress. *American Journal of Political Science*, 49:313–26, 2005.

Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI Symposium on Computational Approaches to Analysing Weblogs (AAAI-CAAW)*, pages 100–107, 2006.

Michael Laver and John Garry. Estimating policy position from political texts. *American Journal of Political Science*, 44(3), 2000.

Michael Laver, Kenneth Benoit, and John Garry. Extracting policy positions from political texts using words as data. *American Political Science Review*, 97:311–331, 2003.

Eric Lawrencea, John Sidesa, and Henry Farrell. Self-segregation or deliberation? blog readership, participation, and polarization in american politics. *Perspectives on Politics*, 4(8): 141–157, 2010.

Kevin Lerman, Ari Gilder, Mark Dredze, and Fernando Pereira. Reading the markets: Forecasting public opinion of political candidates by news analysis. In *Proc. of COLING*, Manchester, UK, 2008.

Kristina Lerman and Rumi Ghosh. Information contagion: An empirical study of the spread of news on Digg and Twitter social networks. In *ICWSM*, 2010.

Jure Leskovec, Andreas Krause, Carlos Guestrin, Christos Faloutsos, Jeanne VanBriesen, and Natalie Glance. Cost-effective outbreak detection in networks. In *Proc. of KDD*, 2007a.

Jure Leskovec, Mary Mcglohon, Christos Faloutsos, Natalie Glance, and Matthew Hurst. Cascading behavior in large blog graphs. In *In SDM*, 2007b.

Wei Li and Andrew McCallum. Semi-supervised sequence modeling with syntactic topic models. In *Proceedings of the National Conference on Artificial Intelligence*, volume 20, page 813. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

W.-H. Lin, E. Xing, and A. Hauptmann. A joint topic and perspective model for ideological discourse. In *Proc. of 2008 ECML and Principles and Practice of Knowledge Discovery in Databases*, 2008.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

R. Malouf and T. Mullen. Graph-based user classification for informal online political discourse. In *Proc. of the 1st Workshop on Information Credibility on the Web*, 2007.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008. ISBN 0521865719, 9780521865715.

Andrew D. Martin and Kevin M. Quinn. Dynamic ideal point estimation via Markov chain monte carlo for the u.s. supreme court, 1953-1999. *Political Analysis*, pages 134–153, 2002.

A. McCallum. Multi-label text classification with a mixture model trained by EM. In *AAAI Workshop on Text Learning*, 1999.

Nolan M McCarty and Keith T Poole. An empirical spatial model of congressional campaigns. *Political Analysis*, 7(1):1–30, 1998.

P McCullagh and J.A. Nelder. *Generalized Linear Models*. Chapman & Hall, 1989.

D. Mimno and A. McCallum. Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. In *Proc. of UAI*, 2008.

Thomas Minka. Estimating a Dirichlet distribution, 2000. `http://bit.ly/XTEJFu`.

Gilad Mishne and Natalie Glance. Leave a reply: An analysis of weblog comments. In *Proc. of Workshop on the Weblogging Ecosystem*, 2006.

Burt Monroe, Michael Colaresi, and Kevin M. Quinn. Fightin' words: Lexical feature selection and evaluation for identifying the content of political confrict. *Political Analysis*, pages 372–403, 2008.

Arjun Mukherjee and Bing Liu. Modeling review comments. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 320–329. Association for Computational Linguistics, 2012.

T. Mullen and R. Malouf. A preliminary investigation into sentiment analysis of informal political discourse. In *Proc. of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.

Kevin P Murphy. *Machine Learning: A Probabilistic Persepective*. MIT Press, Cambridge, MA, 2012.

R. Nallapati and W. Cohen. Link-PLSA-LDA: A new unsupervised model for topics and influence of blogs. In *Proc. of the 2nd International Conference on Weblogs and Social Media*, 2008.

Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39:103–134, 1999.

Brendan O'Connor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proc. of ICWSM*, 2010a.

Brendan O'Connor, Jacob Eisenstein, Eric P. Xing, and Noah A. Smith. Discovering demographic language variation. In *Proc. of NIPS Workshop on Machine Learning for Social Computing*, 2010b.

Brendan O'Connor, David Bamman, and Noah A. Smith. Computational text analysis for social science: Model complexity and assumptions. In *Proc. of the NIPS Workshop on Comptuational Social Science and the Wisdom of Crowds*, 2011.

Iadh Ounis, Maarten de Rijke, Craig Macdonald, Gilad Mishne, and Ian Soboroff. Overview of the TREC-2006 Blog Track. In *Proceedings of the 15th Text REtrieval Conference (TREC 2006)*, 2006.

Bo Pang and Lillian Lee. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. of ACL*, 2004.

Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2: 1–135, January 2008.

Souneil Park, Minsam Ko, Jungwoo Kim, Ying Liu, and Junehwa Song. The politics of comments: predicting political orientation of news stories with commenters' sentiment patterns. In *Proceedings of the ACM 2011 conference on Computer supported cooperative work*, 2011.

Michael J. Paul, ChengXiang Zhai, and Roxana Girju. Summarizing contrastive viewpoints in opinionated text. In *Proc. of EMNLP*, 2010.

Keith Poole and Howard Rosenthal. Spatial model for legislative roll call analysis. *American Journal of Political Science*, pages 357–384, 1985.

Keith Poole and Howard Rosenthal. Patterns of congressional voting. *American Journal of Political Science*, pages 118–178, 1991.

Keith T Poole, Thomas Romer, and Howard Rosenthal. The revealed preferences of political action committees. *The American Economic Review*, 77(2):298–302, 1987.

Martin Potthast, Benno Stein, Fabian Loose, and Steffen Becker. Information retrieval in the commentsphere. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(4):68, 2012.

David Price. *Who Makes the Laws? Creativety and Power in Senate Committees*. Transaction Publishers, 1972.

Kevin M. Quinn. Bayesian factor analysis for mixed ordinal and continuous responses. *Political Analysis*, pages 338–353, 2004.

Kevin M. Quinn, Burt L. Monroe, Michael Colaresi, Michael H. Crespin, and Dragomir R. Radev. An automated method of topic-coding legislative speech over time with application to the 105th–108th U.S. Senate, 2006. Midwest Political Science Association Meeting.

Kevin M. Quinn, Burt Monroe, Michael Colaresi, Michael Crespin, and Drago Radev. How to analyze political attention with minimal assumptions and costs. *American Journal of Political*, pages 209–228, 2010.

Alan Ritter, Colin Cherry, and Bill Dolan. Unsupervised modeling of twitter conversations. In *Proc. of NAACL*, 2010.

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

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *WWW*, 2010.

Noah A. Smith. *Linguistic Structure Prediction*. Synthesis Lectures on Human Language Technologies. Morgan and Claypool, May 2011.

M. Steyvers and T. Griffiths. Probabilistic topic models. In T. Landauer, D. McNamara, S. Dennis, and W. Kintsch, editors, *Handbook of Latent Semantic Analysis*. Lawrence Erlbaum, 2007.

M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. L. Griffiths. Probabilistic author-topic models for information discovery. In *Proc. of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

Sakaki Takeshi, Okazaki Makoto, and Matsuo Yutaka. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proc. of WWW*, 2010.

Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proc. of EMNLP*, 2006.

I. Titov and R. McDonald. A joint model of text and aspect ratings for sentiment summarization. In *Proc. of ACL-08: HLT*, 2008.

Andranik Tumasjan, Timm Oliver Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Predicting elections with Twitter: What 140 characters reveal about political sentiment. In *ICWSM*, 2010.

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

Kevin Wallsten. Political blogs: Transmission belts, soapboxes, mobilizers, or conversation starters? *Journal of Information Technology and Politics*, 4(3):19–40, 2008.

Chong Wang, David Blei, and David Heckerman. Continuous time dynamic topic models. *arXiv preprint arXiv:1206.3298*, 2012.

John Myles White and Scott Counts. Improving spatial models of political ideology by incorporating social network data. In *Workshop on Information in Networks*, 2012.

Tae Yano and Noah A. Smith. What's worthy of comment? content and comment volume in political blogs with topic models. In *Proc. of ICWSM*, 2010.

Tae Yano, William W. Cohen, and Noah A. Smith. Predicting response to political blog posts with topic models. In *Proc. of NAACL-HLT*, 2009.

Tae Yano, Philip Resnik, and Noah A. Smith. Shedding (a thousand points of) light on biased language. In *In Workshop on Creating Speech and Language Data With Mechanical Turk, NAACL-HLT 2010*, 2010.

Tae Yano, Noah A Smith, and John D Wilkerson. Textual predictors of bill survival in congressional committees. In *NAACL*, 2012.

Tae Yano, Dani Yogatama, and Noah A Smith. A penny for your tweets: Campaign contributions and capitol hill microblogs. In *ICWSM*, 2013.

Sarita Yardi and Danah Boyd. Tweeting from the town square: Measuring geographic local networks. In *Proc. of ICWSM*, 2010.

Dani Yogatama, Michael Heilman, Brendan O'Connor, Chris Dyer, Bryan R. Routledge, and Noah A. Smith. Predicting a scientific community's response to an article. In *Proc. of EMNLP*, 2011.

H. Zhang, B. Qiu, C. L. Giles, H. C. Foley, and J. Yen. An LDA-based community structure discovery approach for large-scale social networks. In *Proc. of the IEEE International Conference on Intelligence and Security Informatics*, 2007a.

Yan Zhang, A.J. Friend, Amanda L. Traud, Mason A. Portes, James H. Fowler, and Peter J. Mucha. Community structure in congressional cosponsorship networks. *Physica A*, 387(7), 2007b.

Jun Zhu, Ahmed Amr, and Eric P. Xing. MedLDA: maximum margin supervised topic models for regression and classification. In *Proc. of ICML*, 2009.